A Corpus Driven Computational Intelligence Framework for Deception Detection in Financial Text

Thesis submitted in accordance with the requirements of
the University of Stirling for the degree of Doctor of Philosophy

Saliha Minhas

Division of Computing Science and Mathematics
School of Natural Sciences
University of Stirling
Scotland, UK

November 2016

ABSTRACT

Financial fraud rampages onwards seemingly uncontained. The annual cost of fraud in the UK is estimated to be as high as £193bn a year [1] . From a data science perspective and hitherto less explored this thesis demonstrates how the use of linguistic features to drive data mining algorithms can aid in unravelling fraud. To this end, the spotlight is turned on Financial Statement Fraud (FSF), known to be the costliest type of fraud [2]. A new corpus of 6.3 million words is composed of102 annual reports/10-K (narrative sections) from firms formally indicted for FSF juxtaposed with 306 non-fraud firms of similar size and industrial grouping. Differently from other similar studies, this thesis uniquely takes a wide angled view and extracts a range of features of different categories from the corpus. These linguistic correlates of deception are uncovered using a variety of techniques and tools. Corpus linguistics methodology is applied to extract keywords and to examine linguistic structure. N-grams are extracted to draw out collocations. Readability measurement in financial text is advanced through the extraction of new indices that probe the text at a deeper level. Cognitive and perceptual processes are also picked out. Tone, intention and liquidity are gauged using customised word lists. Linguistic ratios are derived from grammatical constructs and word categories. An attempt is also made to determine '*what'* was said as opposed to '*how*'. Further a new module is developed to condense synonyms into concepts. Lastly frequency counts from keywords unearthed from a previous content analysis study on financial narrative are also used. These features are then used to drive machine learning based classification and clustering algorithms to determine if they aid in discriminating a fraud from a non-fraud firm. The results derived from the battery of models built typically exceed classification accuracy of 70%. The above process is amalgamated into a framework. The process outlined, driven by empirical data demonstrates in a practical way how linguistic analysis could aid in fraud detection and also constitutes a unique contribution made to deception detection studies.

PREAMBLE

Four observations have forged to produce this research. Rising fraud, rising textual data, the potential of computational techniques to uncover deception and the dearth of analysis using textual data in the financial reporting domain. Succinctly, the central premise of this thesis is that an intent to deceive filters through to the language deployed and this can be detected using techniques delineated in this thesis. Language in financial reporting is put under the spotlight. Transparent and effective financial communication in this area is pivotal in maintaining stakeholder trust and confidence in the stewardship of firms that drive economic growth. Catastrophic financial events such as financial statement fraud can threaten the fabric of financial markets rendering the raising of capital to fuel investment a much more arduous task. As the Economist put it [3]: "*If accounting scandals no longer dominate headlines as they did when Enron and WorldCom imploded in 2001-02, that is not because they have vanished but because they have become routine"*. The failure of conventional tools used to alert to impending disasters such as the 2008 financial crises has been repeatedly raised. Quantitative models such as the standard bankruptcy prediction models by Altman, Modified Jones Accruals model, Beneish model have played a dominant role in model building for predictive analytics [4]. However results from such models are inconsistent and produce high levels of error. A more systematic and tangential approach using financial narratives could strengthen the predictive analysis landscape in financial reporting. The sentiment expressed by, Jean-Claude Trichet governor of the European Central Bank in 2010 resonated with many analyst: "*macro models failed to predict the crisis and seemed incapable of explaining what was happening to the economy in a convincing manner. As a policy-maker during the crisis, I found the available models of limited help. In fact, I would go further, in the face of the crisis, we felt abandoned by conventional tools*" [4].

In order to contribute to the search for alternatives that can be additional aids to predicting catastrophic financial events such as financial statement fraud (FSF) a corpus is built. This corpus of 6.3 million words long is constructed from narrative sections of annual reports/10-K of firms formally indicted for financial statement fraud, juxtaposed with non-fraudulent firms from the same industry in the same time period and of similar size. The underlying objective for this premise is to show that those who

have the audacity to misstate the numbers would also lace the text with dishonesty. A corpus is the central armoury used in this thesis to demonstrate that this can be the case. Despite criticisms from adherents of rationalist approach to language a corpus remains the only source of linguistic utterances, the naturally occurring data that is observable. It is a powerful tool from the point of view of the scientific method as it is open to objective verification of results. This cannot be said of thought processes, the reliance on introspection the tenets of belief of the rationalist school of thought [5, 6].

Further, despite research on annual reports that points to paucity of value relevant information, its status as the most important statement of firm progress and intentions by its stewards remain steadfast. Recent attempts by the UK government [7] to revamp the reports to aid transparency (by enforcing a new structure to draw out relevant information) is in resonance with this fact. The quantitative sections (the balance sheet, profit and loss, cash flow statements) have been heavily analysed for anomalies such as fraud and bankruptcy using mostly ratio analysis. In contrast the textual sections have been less intensively scrutinised, largely because they are harder to analyse. As Ingersoll et al. [8] maintain: "*text comes in all shapes and meanings and trips up even the smartest people on a regular basis*". However this thesis seeks to show that using a corpus upon which a range of computational techniques are applied, text can be tamed to prize out underlying patterns in language use that aid in discriminating a fraud from a non-fraud firm.

The computational techniques deployed are varied to strengthen validity of findings. The ultimate aim is to arrive at a feature set that captures aspects of language use in the reports that could aid in the discrimination task. Tools from within the resurgent discipline of corpus linguistics are used to pick up distinguishing linguistic patterns such as keywords and collocations. Both customised for the financial domain and the most up to date general dictionaries are used to extract word types that may be significant. Obfuscation in text is a key linguistic ploy used by those intent on deception to reduce readability is also put under the spotlight. Coh-Metrix, a state of the art tool is executed over the corpus to pick up a range of readability indices in the text to determine if fraud reports contain less readable text than their non-fraud counterparts. Thematic content is also examined using topic modelling techniques. A new algorithm is written using Part of Speech (POS) tagging and WordNet to pull out key concepts from the corpus. All these techniques provide features that are representative of the document, the annual reports/10-K. Once these features have been extracted they are

then put through feature selection routines to reduce dimensionality of matrices and irrelevant features. The reduced feature sets are then passed to the classification and clustering algorithms that determine if using the features given, a fraud report can be distinguished from a non-fraud report. The results are revealing. In the depicted ways used to extract linguistic features, the classifiers and the k-means clustering algorithm that ultimately use these features indicate a difference in the language used between fraud and non-fraud firms. The corpus was examined from a variety of lens to demonstrate that rigorous analysis of the text was undertaken. Consequently, given the results the framework proposed in this thesis could be added to an auditor's toolkit to alert to potential anomalies in a firms financial reporting.

TABLE OF CONTENTS

APPENDICES

LIST OF FIGURES

LIST OF TABLES

ACKNOWLEDGEMENTS

I have really liked coming to the university. The majestic hills, the cloud reflecting lake, gliding swans and birds, capped by the glorious Wallace monument is wonderful to behold. I have also enjoyed helping out in the labs in the department over a couple of years. Through this, I got to know a few of my fellow PhD students and staff. They are all very pleasant and friendly. I am glad I got to know them a little. Many thanks to the department for offering this opportunity. I learnt a lot from the students and the material they were working on. I should add also that I have liked attending seminars and courses organised by the Stirling Graduate School. They were engaging and informative and an opportunity to meet other research students.

I would like to extend my thanks to my supervisor, Prof Amir Hussain for his oversight and efforts in this PhD endeavour. I would also like to thank Sam and Graham for their help on various issues. In particularly, Sam set up remote access to machines at the university for me. He ironed out all the issues surrounding this which made a big difference to my productivity. Similarly, many thanks to Nia Dowell at the University of Memphis who helped out with data processing requests using tools that the university have developed. Again this made a big difference to my productivity.

Furthermore, thanks also to Prof Steven Young at Lancaster University, who swiftly answered all emails and provided the requested help and advice on various issues. Many thanks also to Andrew Thurston at the University of Glasgow and Savi for organising lab sessions that I worked on, in a time optimal manner, helped me considerably. It freed up more of my time to work on the thesis. Thank you also to Gabriela for reading a final version of the thesis and giving feedback. I would also like to thank Dr Vander Viana at the School of Education for some tips on corpus analysis and taking an interest in my work. Lastly thanks to IT staff at university libraries, Glasgow and Stirling for giving training and advice on referencing up the thesis.

However, without the support of my family this PhD would have been dead in the water. My husband who funded this PhD, without question and took good care of the kids. My sister and brother law for unrepayable childcare and moral support. My nieces and nephews for fun and laughter. My three children who fire me up every day to keep trying and who have patiently put up with me staring at the screen for hours. One person that stands tall in my life is my mum (even though she is petite in size!). She has always been there for me, no matter what. This thesis is dedicated to my mum.

Finally (at last, truly finally), I thank God, that this day has come and that the will to finish this thesis didn't depart, that it overcame doubt and despair. All praise be to God and as Malcolm X once said, only the mistakes have been mine.

DECLARATION

I understand the nature of plagiarism, and I am aware of the University's policy on this.
I certify that this dissertation reports original work by me during my University project.
I confirm that this thesis has not been previously submitted for the award of a degree by this or any other university.

_____ _____

Signature Date

GLOSSARY OF ABBREVIATIONS AND DEFINITIONS

AI: Artificial Intelligence

AR Annual Report

DT: Decision Trees

f: fraud

FSF: Financial Statement Fraud

IG: Information Gain

IFRS: International Financial Reporting Standards

LIWC: Linguistic Inquiry and Word Count

kNN: k-Nearest Neighbors

LSA: Latent Semantic Analysis

LDA: Latent Drichlet Allocation

LBC: Linguistic Based Cues

nf: non-fraud

MDS: Multidimensional scaling (equivalent to MDA)

MDA: Multidimensional analysis

NLP: Natural Language Processing

NLTK: Natural Language Processing Toolkit

OCR: Optical Character Recognition

PCA: Principal component analysis

POS: Part of Speech

RF: Random Forest

SEC: Securities and Exchange Commission

SGB: Stochastic Gradient Boosting

SFL: Systemic-Functional Linguistics ()

SVM: Support Vector Machine

TDM: Term Document Matrix

tm: text miner

VSM: Vector Space Models

alere flammam

Chapter One

INTRODUCTION

"We are men, and hold together, only by our word… Lying is an ugly vice… Since mutual understanding is brought about solely by way of words, he who breaks his word betrays human society. It is the only instrument by means of which our wills and thoughts communicate, it is the interpreter of our soul. If it fails us, we have no more hold on each other, no more knowledge of each other. If it deceives us, it breaks up all our relations and dissolves all the bonds of our society"

Montaigne c.1572

1.1 The Challenge Posed by Language

It uniquely raises the human stature above all of creation and is woven into the fabric of existence. Miraculous is its many forms and varieties and how this perhaps shapes the psyche of the users. Although it is different outwardly it has a commonality, an all embracing pervasiveness. All varieties are used to give meaning to thought, convey self-awareness and consciousness to others. Yet how did it originate? How is it acquired? And the question tackled in this thesis, how can it be tamed for computation? All fundamental questions still subject of fierce debate. This mystifying and remarkable ability is language.

The complete comprehension of language, marked by sophisticated word play such as metaphor, sarcasm, euphemism, innuendo will be the ultimate triumph for Artificial Intelligence (AI). This however remains a far and distant goal. The few "*intelligent*" systems that exist struggle to comprehend even fragments of natural language, throw any sophisticated word play at them and they quickly crumble. Put plainly: "*No AI system today can learn, understand, and use language as quickly and accurately as a 3-year old child*" [9]. Why is language so elusive, so difficult to tame for computation? To begin from within, there is evidence to suggest that language was developed for thought. Pinker [10] argues that: *"People do not think in English or Chinese or Apache; they think in a language of thought which is more alike that the spoken equivalent. This language of thought probably looks a bit like all these languages, presumably it has symbols for concepts, and arrangements of symbols that correspond to who did what to whom. Knowing a language then is knowing how to translate mentalese (language of thought) into strings of words and vice versa*".

This view that words represent concepts is a challenge for Artificial Intelligence. Computers have no appreciation of concepts such as dog, cat, knife, profit, loss, it has no knowledge of their attributes, characteristics or relations [11]. Therefore, a key aspect of understanding language means, among other things: "*knowing what concepts a word or phrase stands for and knowing how to link those concepts together in a meaningful way*" (Microsoft Research quoted in [12]). Chapter two will discuss steps that have been taken to build in background knowledge and common sense for computational purposes.

Language can be incredibly imprecise, yet surprisingly accurate [13]. As indicated it is full of idiosyncrasies and idioms. Yet despite this humans can: "*see through the gaps, the inconsistencies and contradictions, the irregularity, and the lack of clarity and still understand each other with a great deal of accuracy*" [13]. Language is also marked by: "*vagueness, ambiguity, and context sensitivity*" [14]. Ambiguity, can result from both a lack of world knowledge on what a concept is and also from the grammatical use of words. Words and sentences can have multiple meanings and a concept can be expressed in a multitude of ways. The challenge for computers in understanding natural language is how to resolve ambiguity when interpreting a sentence. For example, similar sentences can have different meanings [15]:

"The CEO was fired up about his new role"
"The CEO was fired from his new role"

Seemingly different sentences can have the same meaning:

"*IBM's PC division was acquired by Lenovo"*
"Lenovo bought the PC division of IBM"

Humans can figure out the intended meanings of these sentences by inferring from context or by drawing on personal knowledge and understanding. Whereas, computers do not benefit from the: "*subtleties of human experience and learning, which is why determining intended meaning is such a difficult task*" [15].

Another fundamental fact about language that keeps it elusive from computation is:

"virtually every sentence that a person utters or understands is a brand-new combination of words, appearing for the first time in the history of the universe. Therefore a language cannot be a repertoire of responses; the brain must contain a

recipe or program that can build an unlimited set of sentences out of a finite list of words" [10].

Therefore, language is not only ambiguous but immensely variable and changeable. This view espouses the belief that humans are genetically endowed with language just as: "*spiders spin webs…because they have spider brain"* [10], humans learn to talk because it is hard-wired into our brains [16]. This belief is generally known as universal grammar and is the main doctrine of the rationalist school of thought. Attempts at: "*direct mimicry of nature algorithms*" [17] have been made in order to build intelligent systems by hand-coding into them a lot of starting knowledge and reasoning mechanisms. Projects such as Cyc [18] that attempt to encode common sense knowledge into an ontology is an attempt in this direction. However, issues surrounding the scalability and complexity has prevented its expected wide scale adoption and advance [19, 20].

The alternative empiricist view reinforces the rationalist belief that there is a basic machinery in the brain that aids language learning, the mind is not a blank slate. However, it propagates the view that by observing language in use much can be learnt about its composition. Typically, a large body of text, a corpus in a domain of interest is put under the spotlight. This is then used to draw out linguistic structures of interest by applying pattern recognition and machine learning models. This empiricist corpus based approach resolves ambiguity by detecting context through the use of words or knowing a word: "*by the company it keeps*" [21]. Therefore a document collection on cricket would contain words such as run, wicket, catch, century as opposed to a document collection on astronomy with a very different vocabulary [22]. Such distinctions are attained through frequency counts and application of statistical techniques.

The computational techniques adopted within this thesis falls within this empirical school of thought. Whilst acknowledging that language has the elusive nature described above, it can be grasped for computational purposes, if enough evidence of its use is gathered from a narrow field of interest. This "*naturally occurring data*" [6] is a corpus. This is subject to computational techniques that are used to tame the text for prediction and analytical purposes.

1.2 The Problem Domain - Financial Reporting

The imperative to find ways and means to further the computational grasp of language has gained momentum with the rise of big data. It is estimated that 80% of this data is unstructured text [23]. This includes data such as facebook, twitter, blogs, academic research, business reports and many other sources. Such data is informational rich and can reveal customer needs, competitor actions, emerging trends and other pieces of information necessary to make critical business decision. However, the scale of the data renders it impossible for humans to extract valuable information from it. From a computational purposes, the existence of such volumes of data leans more favourable to the empiricist corpus based approach where knowledge can be distilled using statistical techniques.

Given the difficulties to automated language understanding, success at developing systems that process text have been limited to narrow, defined domains. These domains include spell checkers, google translate, email spam filters, question answering amongst others. This restriction in breadth and often depth enables better management of the ambiguity and variability in linguistic structures that are present in language, thus enabling development of applications that are fit for purpose.

This thesis focuses in on a particular genre – financial text. The definition for genre demonstrates that it fulfils the criteria for it to be a distinct body of text, which is narrow and defined: *"A genre comprises a class of communicative events, the members of which share some set of communicative purposes. These purposes are recognised by the expert members of the parent discourse community, and thereby constitute the rationale for the genre. This rationale shapes the schematic structure of the discourse and influences and constrains choice of content and style"* [24]. Given this definition, financial text would have idiosyncratic linguistic structure understood by its readership. This specialised linguistic style with defined content that is constrained would be more conducive to computation.

Therefore, given the above argument a corpus would be an appropriate approach to use tame text in the financial domain. Significantly, in this area there is a dire need to seek out alternative ways to perform predictive analytics using text. An example of this is the near collapse of capitalist economies bought forth by the 2008 financial crises.

This prompted an investigation into weaknesses that could have caused such a large scale calamity. A few areas identified were:-

- Risk, defined as probability attached to an outcome and uncertainty, where outcome is not known [25]. Both were spectacularly underestimated and tackled using insufficient information.
- The fundamental underpinning that man is a rational agent and has rational expectations is flawed [26]

General Equilibrium theories and business cycle analysis that uses measures such as VaR (Value at Risk - risk of loss on a specific portfolio of assets) are misleading. Such prognosis emanate from the existence of agency and information asymmetry that permeate through all aspects of financial dealing [27]. In financial reporting, two competing explanations are put forth as to how it impinges upon this domain. The first being the efficient market hypothesis view that economic agents act in a rational, utility maximising manner. The assumption using agency theory is that managers are motivated by incentives to provide: "*information gain*" [28] as it enhances their reputation and compensation. Investors would then absorb all such data into their rational decision-making process. As a consequence of this rationality and drive to provide value-add information the incentives for biased reporting is reduced as users driven by utility maximisation are able to detect bias.

The other view rooted in behavioural finance theory is that information asymmetry and agency can result in impression management – where managers have the potential to: "*distort readers' perceptions of corporate achievements*" [29] by means of obfuscating failures and emphasizing successes [29]. This opportunistic behaviour by managers exploit information asymmetries by releasing biased information such as: "*cognitive constraints that render investors unable to undo reporting bias, resulting in short-term capital misallocations*" [30]. This underlying theory directs this research as managers can exploit information asymmetries which can result in bias to falsification in their financial reporting. Management that engage in impression management are not being untruthful but often introduce bias in their narratives to deflect blame for poor performance [31]. Therefore, if the avenue for bias as afforded by agency and information asymmetry is available then the door is open for those who take the leap further into outright falsification. This gives rise to '*opportunity*' [2] one of the three factors in the fraud triangle as depicted by Rezzae [2]. These factors when present in

a firm increases the likelihood of financial statement fraud. The other factor being pressure or incentives for example to meet analyst expectations or high debt. The third is an attitude or rationalisation that justifies the misconduct by the perpetrators.

To date much of financial modelling and forecasting relies on data that rest on belief that humans beings are rational agents. No account is taken of the: "*plurality of human emotions, the connection between ethics and economics*" [32]. This view is corroborated by Adam Smith [33] who expounded the role of motivations that influence human action and behaviour in economic exchange. His well quoted phrase sums up well the greed and self-interest that escaped quantitative analysis in the 2008 crash: *"It is not from the benevolence of the butcher, the brewer, or the baker that we expect our dinner, but from their regard to their own interest. We address ourselves, not to their humanity but to their self-love"* [33]. Therefore, there is dire need for ways and means that would detect human motivations such as greed and avarice that are unleashed through opportunity afforded by information asymmetry and agency. It is such leanings that went undetected in predictive models that failed to alert to warnings signs that could have forewarned of the 2008 financial crises.

The backbone of this thesis is that those given to leanings such as greed, avarice, deception leave traces of their culpability in the language that they use. Financial markets have been rocked by cases where such motivations have stripped investors of millions and damaged trust and transparency which is pivotal to the smooth running of financial markets. Although cases such as Enron, Worldcom, Tyco are high profile disasters, they are symptomatic of an entrenched and widespread problem – financial statement fraud (FSF) or '*cooking the books'*. This type of fraud causes the biggest loss: "*a median loss of $1 million per case*" [34]. In forensic accounting, detection has been dominated by quantitative analysis (mostly ratio analysis) using data from financial statements garnered from profit and loss, balance sheets and cash flow statements.

This leads into the problem domain put into the spotlight: detection of financial statement fraud from narrative sections of financial reports. If "*cooking the books*" results in misstated numbers then there must be misleading, falsified narratives. It is an answer to this question that is sought, does deception leave a linguistic signature that can be detected using the corpus based approach driven by computational techniques? Can it pick up differences, if any between the linguistic style of fraud and non-fraud firms? As indicated, it is not rationality that predominates in economic

agents but behavioural traits. Do such leanings as the intent to deceive leave an imprint in the language used? Can such an imprint be picked up using natural language based computational techniques?

1.3 Motivation and Aims

As indicated to counteract the shortcomings of the predictive analytical techniques based on numerical data that exist in the financial reporting domain and gain an appreciation on the language used in this area, a corpus is built. This corpus contains narrative sections of annual reports/10-K of companies formally indicted for fraud juxtaposed with companies of similar size, industry and time period where fraud has not been detected. For the stakeholder community the annual report remains the most significant document that companies are required by law to produce on their operations. Significant research has been done to develop models from the quantitative sections of annual reports for predictive model building purposes [35, 36]. In contrast to the quantitative sections of the annual report the narrative sections are much longer and would contain more details on company operations. In fact, as Goel et al. [37] argue that only a tiny fraction of corporate information is quantitative in nature.

Thus far given the arguments outlined. It can be deduced that:-

1. With the rise of big data, a corpus based approach using machine learning techniques to detect textual patterns in a domain of interest is an appropriate approach to take.
2. The computational challenges faced in understanding language can be mitigated to some extent by dealing only with text of a certain genre.
3. There is a dire need in the financial domain for technologies that shift the focus away from the numerical analysis. Financial Statement Fraud is a specific problem identified to demonstrate the benefits of applying machine learning based classifiers to aid deception detection using linguistic features.

Therefore, in order to demonstrate that language can be harnessed in the financial reporting domain to aid in the differentiation of a fraud from a non-fraud firm, a corpus of narratives extracted from annual reports/10-K is constructed. Further, details on corpus composition is given in chapter 3. From this corpus features are extracted that

are representative of the documents. These features are chosen based on past research in deception detection which indicates that they have the potential to discriminate between liars and truth-tellers. Once the features are extracted, they are checked for their usefulness using a number of feature selection techniques, delineated in chapter 5. Then classification which attempts to perform the differentiation between fraud and non-fraud reports based on features selected using a number of supervised machine learning techniques is attempted. An unsupervised technique, k-means is also deployed over the corpus. Full details of classifiers is expounded in chapter 6.

The whole process of deception detection from the corpus of financial text will be mapped out into a framework to demonstrate that this whole process could be streamlined.

1.5 Original Contributions

- In the arena of high stakes deception there is a: "*scarcity of ground truth verification for data collected from real world sources*" [38]. To address such a short supply of "*ground truth*" a new corpus of 6.3 million words is constructed that contains narratives of firms known to have committed financial statement fraud, juxtaposed with narratives from similar non-fraud firms.

- This is the first study that systematically in a wide ranging manner examines this corpus of narratives from fraud/non-fraud firms for differences in the use of language. It does this in the following ways:-
  - Application of methods from Corpus linguistics to determine keywords and collocations. No other study has used in a formal way a corpus linguistic methodology to study financial narratives from fraud/non-fraud firms.
  - Extraction of n-grams, this would enable pick up of multi-word expressions that can reveal patterns of language use. No other study has comprehensively looked at all three n-grams (unigrams, bigram, trigrams) as used in a body of financial text to detect deception.
  - Extraction of readability measures that examine the text at a deeper level than the simplistic formulas currently in use. This is the first study that

investigates readability measures extracted from the corpus using the Coh-Metrix tool in a bid to gauge if obfuscation is more marked in fraud reports.

- An attempt is made to decipher '*what*' is said in the corpus rather than '*how*' using topic modelling techniques. Topics are extracted and each document is weighted by topic. This is again the first time where this technique has been applied to a corpus, such as the one under study.
- This is the first study that executes Linguistic Inquiry Word Count (LIWC) 2015 for deception detection in financial reports. This new version contains modern, revised internal dictionaries that are linked to words that are representative of the linguistic categories that it seeks to measure in the text.
- Use of customised word lists to pick up language categories related to tone, intention and risk. This is the first time that a comprehensive set of word lists specific to the financial reporting domain have been applied to a corpus of the nature under study.
- Condensing words to concepts using WordNet and POS tagging. A new module is developed using Python 3.5. The reports (documents) are represented by concepts. This is the first time that such a program was executed over a corpus of the nature described.
- Linguistic based cues (LBC) deemed to capture deception in text is executed over the corpus. The extraction of LBC are derived from tools that are more robust and current than previous similar work. Further the corpus constructed in this study is much more extensive than used by previous research.

- Three separate category of feature selection routines are applied over the corpus in a bid to determine the most discriminatory features. This is the first study that in a comprehensive manner uses these 3 routines to exhaustively search for the features that are most optimal ahead of the classification task.

- For visualisation and exploratory data analysis purposes, Multi-Dimensional Scaling (MDS) is applied to the features selected above to provide an overview and insight into how the reports represented by features extracted and selected

are distinct. Again this technique as applied to the corpus under study is the first such endeavour.

- A cohort of classification algorithms based on supervised and unsupervised machine learning algorithms are executed over the features extracted to determine success in discriminating a fraud from a non-fraud firm.

- The above process is captured in a framework that is new to the domain of deception detection in financial reports.

1.6 Structure of Thesis

Chapter 2 delineates the competing views that impinge on the study of language and computation. It then turns to examine further the problem domain under study financial fraud and in particular financial statement fraud (FSF). Competing theoretical views in this area are mapped out. The theoretical strands behind known linguistic imprints of deception are outlined. A key such construct, readability is fully examined. The chapter closes with a comprehensive literature review (the past six years) into FSF detection using intelligent techniques.

Chapter 3 maps out the framework developed in this thesis for deception detection. It then examines the corpus constructed using techniques from the methodological discipline of Corpus Linguistics. Differences found between fraud and non-fraud reports are put through significance testing for verification. Keywords are extracted using a number of customised word lists.

Chapter 4 rolls out the varied ways in which the reports will be represented through a select number of features. The chapter builds up from the standard vector space model and then enumerates the novel approaches used to extract features of interest with respect to deception in text. The tools used to enact the extraction are explained and the resultant matrices for the 408 reports (the rows) with the features forming the columns are laid out.

Chapter 5 uses dimensionality reduction and feature selection techniques to reduce the dimensions of the matrices and remove spurious relations to enable the downstream classifiers to run more efficiently. Multidimensional scaling (MDS) is

deployed for visualisation purposes to highlight the distances between vectors produced from fraud and non-fraud reports.

Chapter 6 introduces the machine learning task. It describes the classification process and some of the pitfalls inherent in the process and counter measures. It also introduces a clustering technique k-means in a bid to show how the fraud and non-fraud reports could also be separated using an alternative machine learning approach. All results obtained are detailed in Appendix N to W.

Chapter 7 closes with a review and emphasizes contribution made by thesis. It also delineates the limitation of this research. Further ideas are mentioned on how to determine '*quality*' of financial text and how to capture compositionality of the meaning at a sentence level in a further bid to identify text mired with deception.

1.7 Publications

Minhas S, Poria S, Hussain A, Hussainey K, "A review of artificial intelligence and biologically inspired computational approaches to solving issues in narrative financial disclosure" BICS'13 Proceedings of the 6th international conference on Advances in Brain Inspired Cognitive Systems Pages 317-327, 2013.

Minhas S and Hussain A, "From Spin to Swindle: Identifying Falsification in Financial Text", Cognitive Computation, 8 (4), pp. 729-745, 2016.

Minhas S and Hussain A, "Linguistic Correlates of Deception in Financial Text - A Corpus Linguistics Based Approach", Stirling International Journal of Postgraduate Research, 1.3, 2016

Chapter Two

STATE OF THE ART AND LITERATURE REVIEW

Say what you believe to be true (Maxim of Quality), do not say more than needed (Maxim of Quantity), stay on the topic (Maxim of Relevance), and do not be vague (Maxim of Manner)

Grice's Four Maxims, 1975

2.1 Introduction

As indicated in the preamble, this thesis is pivoted on observations of rising fraud, rising textual data and the potential of techniques from the toolkit of computational linguistics to detect patterns in text that are indicative of anomaly in a domain of interest. In order to gauge insight and understanding, each of these factors will be examined. Some prominent views on language use will be outlined and two main perspectives that hold currency will be expanded. This is needed to orientate this study into the correct arena (the empirical school of thought) and highlight why this choice was made. Thereafter a full exposition of the extent and nature of deception, financial fraud and with an emphasis on financial statement fraud will be undertaken. This is done to highlight the predisposing conditions that lead to such misconduct. Further by examining the research conducted in this area an appreciation would be garnered as to how lies and intention to deceive would be manifested in text. Such understanding leads to developing models that are cognizant of linguistic imprints of deception in text. Limitations with regards to measuring readability measures often used in the literature to gauge the extent of obfuscation by management will be highlighted and an alternative tool based on more extensive measures will be mapped out. Obfuscation is a ploy used by those engaged in bias to deception to detract from the truth. Literature reviews on FSF detection using data mining techniques will be rolled out to provide a picture on the state of the art. This would further aid in the correct placement of this work within the FSF detection domain and would emphasize the contribution made by this thesis.

A few terms abound in the literature that attempt to capture language processing by computational means. A few of the terms will now be defined briefly. Natural Language

Processing (NLP) and Computational Linguistics are almost synonymous and used interchangeably in the literature. Essentially they both deal with techniques that deal with learning, understanding and production of human language content, for example applications include amongst many others: language parsers, mining social media, and sentiment analysis [39]. Similarly, data mining and text mining are often used in a similar context. However, the former deals with the process of discovering novel, interesting patterns in databases that can aid decision making. Typically machine learning techniques are applied to extract and identify these patterns [40]. Text mining is used to denote any system that analyzes large quantities of natural language text and detects lexical or linguistic usage patterns in an attempt to extract useful information [41]. Most researchers agree that the main difference is that in text mining the patterns are extracted from unstructured text as opposed to data mining where the data is structured such as databases. Irrespective of terms used before machine learning techniques can be applied to detect patterns the input data has to be in numerical format. In this thesis the terms used predominantly are data mining and machine learning to detect patterns that enable models to make predictions on new data.

2.2 Language Processing: Background and Potential

An aim of linguistic science is to understand the structure of language. Over the centuries many scholars have looked at this phenomenon and pondered over its mechanics. A few salient ideas are now delineated to provide insight into the issues that need to be grappled with to tame language for computational purposes.

The structural school of thought focuses on how elements of the language such as words convey meaning. Each word relays a concept in the mind of the listener/reader that is strongly rooted in time and place, in other words strongly influenced by societal norms and practices [42].

Systemic-Functional Linguistics (SFL) is a linguistic theory centred on the notion of language function (what language does and how it does it). The main proponent of this theory, Halliday [43] argues: "*language allows people to exchange meanings*". This exchange must be studied within its context as it has distinctive characteristics that separate it from any other exchange in a different context. Systemic linguists chart

their analyses by mapping the choices language users can make in a given setting [44].

Linguistics has been heavily impacted by Noam Chomsky and his theory of generative grammar. Chomsky did away with the prevalent behavioural conditional-response theory of the 1950's to language which postulated that language was learnt. Chomsky [45] argued that language was innate, specifically the rules of grammar (descriptive rules that describe how people actually speak) were hard wired into the brain. Evidence for this comes from children, subjected to "*poverty of the stimulus*". This view states that given no formal instruction on language learning children are able with amazing rapidity to master the rules of grammar and construct well-formed sentences. Chomsky sought to find this universal grammar which he claimed would apply to all languages as he argues they are more similar than different. If a grammar was able to do this then it would have "*descriptive adequacy*", grammar should also have "*explanatory adequacy*" simple enough to reflect the small set of inborn principles that allow humans to acquire and use language. Chomsky's work is a history of tension between these two constraints.

The whole generative grammar program spun off many syntactical modelling techniques such as phrase structure grammar, context free grammar, lexical functional grammar. They all take the sentence as the construct upon which rules are built. Carnie [46] argues sentences are generated by a sub conscious set of procedures, these procedures are part of our cognitive abilities. The goal of syntactic theory is to model these procedures using formal grammatical rules (different from the stylistic grammar rules that is learnt at school).

Cognitive approaches to understanding language overlap with Chomskian views in a similar bid to understand how exactly the language faculty works. Pinker [10] views overlap with those of the structural school of thought who argue that words are symbols that stand for concepts in our minds (a mental dictionary) what symbols are used is irrelevant as long as they are used consistently. There is also he believes a set of rules that combine the words to convey relationships among concepts (a mental grammar). There is a language of thought, or "*mentalese*"—and words are used to cloth these thoughts whenever we need to communicate them to a listener. Furthermore, he adds that language is ambiguous and does not convey all that we wish to convey and much of the information with regards to common sense is implicit. Thoughts in our head are interlaced with reams of information that cannot all be

expressed and only a fraction of the message is couched into words and must count on the listener as to fill in the rest. Significantly, the cognitive view is that the arrangement of symbols represents concepts and propositions in the brain in which ideas including the meaning of words and sentences are couched. Chomsky is also well aligned with the view that at core all languages are invariant and only the externality what comes out the mouth is variable. However Chomsky's central belief that rules were integral to our mental use of syntax has waned, whereas cognitive scientists still believe that they are still integral to "*discovering the mental reality underlying actual behaviour*" [45].

All of the above approaches keep within a tradition that attempts to understand natural language by devising rules that separate well-formed and ill-formed sentences [47]. These ideas fall within the rationalist school of thought. A rationalists theory is a theory: "*based on artificial behavioural data and conscious introspective judgements*" [5]. The aim is to develop a theory of language that: "*not only emulates the external effects of human language processing, but actively seeks to make the claim that it represents how the processing is actually undertaken*" [5].

Manning and Schutze [48], Manning et al [49], Hirschberg and Manning [39] argue that the over reliance on rules to grasp the linguistic structures used in communication has proven inadequate. They contend that "*all grammars leak*" that it is not possible to have hard and fast rules, given the flexibility with which people deploy language to convey meaning. There is no definitive way to separate a well formed sentence from an ill formed sentence. However they agree that even given this loophole there is a structure to sentences that obey to a large degree syntactic rules which seek to pick out grammatical constructs such as noun phrases/determiners, adjectives etc.

This world view on linguistics has given rise to probabilistic models or the "*return of empiricism*" [6, 17]. An empiricist approach to language is dominated by the observation of naturally occurring data, such as a corpus. As McEnery and Wilson [5] argue it can be used to: "*to determine whether sentence x is a valid sentence of language y by looking in a corpus of the language in question and gathering evidence for the grammaticality, or otherwise, of the sentence*". Probability models have been used in part-of-speech tagging, parsing and ambiguity resolution and semantics. An empiricist approach to NLP suggests that: "*we can learn the complicated and extensive structure of language by specifying an appropriate general language model*

and then inducing the values of the parameters by applying statistical, pattern recognition and machine learning methods to a large amount of language use" [48].
In order to examine "*the complicated and extensive structure of language*" [48] and to induce parameters, a surrogate for situating language in a real world context is used. This surrogate is called a corpus. This is a large and structured set of text [50].
The rationalist like Chomsky attempt to describe operations of the language faculty that resides in the mind (the I-language, also known as competence, the tacit internalised knowledge of language) for which data such as text (the E-language, also known as competence the external evidence of language competence) provide only an indirect clue to the operations of the mind. Empiricists are primarily interested in the use of language, the E-language [51]. They argue that they can get: "*good real world performance*" by assigning probabilities to linguistic events so that they can say which sentences are '*usual*' and '*unusual*' [48]. Chomsky argues it is the knowledge of language that we are trying to grasp therefore linguistic competence and not performance should be modelled. Chomsky's view is: "*rather than try to account for language observationally, one should try to account for language introspectively*" [5].
A few points that Manning and Schutze [48], Hirschberg and Manning [39] expound are worthy of note. In order to study what things people say in a domain of interest they argue that building a corpus is the natural route to follow. This enables an understanding of patterns of use (syntactic structure of a language) and a greater appreciation of the topics under discussion. This depth of understanding that can be prized from these statistical approaches is in sharp contrast to what is sought by Chomsky, grammaticality. The concept of grammaticality judges simply whether or not a sentence is structurally well formed and does not capture the fact that many sentences are of interest, irrespective of grammaticality. Manning and Schutze [48] argue that the difficulties in giving grammaticality judgements: "*to complex and convoluted sentences show the implausibility of extending a binary distinction between grammatical and ungrammatical to all areas of language use*". As indicated under the structural school of thought, language use is subject to change over time: "*the frequency of the use of a word in various contexts gradually changes so that it departs from the typical profile of use of words in the category to which it formerly belongs and comes to resemble words of another category*" [48]. This can best be detected by the use of hard core evidence such as a corpus. This view is reinforced by an upsurge in comparative linguistic studies using corpora [51].

Manning and Schutze [48] also note that we assimilate information from the external world using our sensory input and we use probability in an unconscious way to determine likelihood of the consequences of our actions. These cognitive processes are probabilistic processes that can handle uncertainty. If human cognition is probabilistic, then language processing must be similar. Manning and Schutze [48] contend that processing of words and forming an idea of the overall meaning of a sentence is no different. They also add that the construction of syntactic parsers to answer questions such as: "*who did what to whom*" is strongly reliant on probability theory to handle the ambiguity that is riddled in natural language. A statistical approach using frequency information learns lexical and structural preferences from corpora. It aids in the pickup of collocational knowledge that can enhance understanding of semantic relationships [52].

According to Hirschberg and Manning [39] this statistical (corpus) based approach to NLP has been a notable success, given the rise of big data. They cite POS (part of speech) taggers and sentiment classifiers as examples. They argue that high-performance tools based on a corpus of text and probability theory: *"that identify syntactic and semantic information as well as information about discourse context are now available*" [39]. Such tools are therefore invaluable for applications that need to extract information from natural language.

Hirschberg and Manning [39] identify a few other notable successes for applications built on NLP technology and herald this as a step forward for the Artificial Intelligence community. They cite Machine Translation (MT) as a substantial contribution in which computers have aided in human-human communication. They argue that this success was possible through a corpus of parallel text (corpus contains text for example English and French that are direct translations of each other). This data enables the collection of statistics of word translation and word sequences that were built into probabilistic models of machine translation. Further improvements in this area are ongoing using deep learning based sequence models based on neural networks the goal is to: "*build deeper meaning representations of language to enable a new level of semantic MT*" [39]. These neural networks capture subtle semantic similarities in MT and have produced state of the art results [53].

Spoken dialogue systems (SDS) have also benefitted from advances in NLP. SDS requires automatic speech recognition and identification of what a human says. Currently this process is enabled using Markov decision process. This attempts to:

"*identify an optimal system policy by maintaining a probability distribution over possible SDS states and updating this distribution as the system observes additional dialogue behaviour*" [54].

Hirschberg and Manning [39] also mention machine reading as another AI goal where strides have been made. This is an idea that: "*machines could become intelligent and could usefully integrate and summarise information for humans*" [39]. This task has been approached by building large structured knowledge bases or ontologies. This aim was that this knowledge base would power a reasoning mechanism to derive further new facts. A machine reading system would extract basic facts from text for example a relation. This is often performed by looking for patterns and training machine learning classifiers to identify such patterns. The goal in this area is to extract relations, events, facts and to be able to understand the relations between events. A combination of ontology and machine learning has been applied in this area to forge ahead.

Social media has permeated deeply into human interactions and has unleashed natural language text, like never before into the public domain. Insights and opinions can be garnered on products and services, demographics, language use, social interaction amongst many other real world applications [55]. For example, machine learning techniques have been used to recognize deception in fake reviews [56] and food related illnesses [57]. The best results are derived from applications that use corpora and statistical/machine learning based procedures [39].

IBM Watson is also a system of noteworthy success in its ability to understand natural language. A question-answering system that analyses unstructured data and understands complex questions [58]. Although it uses a range of techniques for its purpose a key ingredient in its success is the use of a knowledge corpus and machine learning techniques: *"Of paramount importance to the operation of Watson is a knowledge corpus. This corpus consists of all kinds of unstructured knowledge, such as text books, guidelines, how-to manuals, FAQs, benefit plans, and news. Watson ingests the corpus, going through the entire body of content to get it into a form that is easier to work with*" [13]. In the development and test stages it was held that machine learning techniques provided the results that indicated that Watson could win the jeopardy (question-answering) game [58].

Success using ontologies to fortify AI systems has been muted. An ontology captures the concepts and relationships between these concepts and the terms used to refer to

these concepts. A vital human like quality that is used for computational purposes, common sense is missing from all computing systems [11]. In order to equip applications with common sense to perform human-like reasoning in a scalable, wide ranging manner the Cyc ontology is being built. This maps out and represents everyday common sense knowledge. CYC is often mentioned as a success of the knowledge-based approach to AI. However, as Davis and Marcus [59] argue that the project in 15 years has not accomplished much. It is not clear what common sense has been captured and applications that use it are few in number. The authors list understandability, learnability, portability, reliability, compliance with standards, and interface to other systems as problematic. They add that CYC has had comparatively little impact on AI research. Such ontologies that are often used by rule based systems are not scalable and the rules can get quite convoluted [60].

Modern statistical/machine learning models that attempt to understand linguistic structures abide by the algorithmic modelling culture as described by Breiman [61]. This holds that nature's black box cannot be described by a simple model. Complex algorithmic approaches, such as support vector machines (SVM), decision trees (DT) are used to estimate the function that maps from input to output variables. Norvig [62] explains that using this approach there is no expectation that the form of the function that emerges from this complex algorithm reflects the true underlying nature. Norvig [62] argues that Chomsky finds this objectionable as these models make no claim to represent the generative process used by nature. In other words: "*algorithmic modelling describes what does happen, but it doesn't answer the question of why*" [62]. Breiman [61] argues that there should not be a heavy focus on trying to model the true underlying form of nature's function from inputs to outputs. As Norvig [62] puts it Breiman [61] argues that it is sufficient that a function: "*accounts for the observed data well, and generalizes to new, previously unseen data well, but may be expressed in a complex mathematical form that may bear no relation to the "true" function's form (if such a true function even exists*)". Chomsky takes the opposite approach: he prefers to keep a simple, elegant model, and gives up on the idea that the model will represent the data well. Instead, he declares that what he calls performance data, what people actually do, is off limits to linguistics; what really matters is competence, what he imagines that they should do [62].

2.3 Theoretical Underpinnings in the Financial Reporting Domain

In perfect market conditions sellers would give accurate price signals and the buyer would carry out diligent screening (risk assessment) in the investments they make [63]. This would satisfy the tenets of portfolio theory and result in the realization of the free market mantra of utility maximization. This efficiency is unattainable, a key reason often cited in the literature is information asymmetry and the agency problem [27]. Agency theory deals with the relationship where one party (the principal) delegates work to another (agent) who performs that work. According to Pepper and Gore [64] there are 2 issues that arise (a) the goals of the principal and agent conflict (b) it is difficult for the principal to verify what the agent is doing. Information asymmetry is where one participant in an economic exchange knows more than the other. This results in the '*lemons problem*' where buyers fearful of being sold '*lemons*' – poor quality products and services, stick to an average price [65]. This exchange could result in goods and services below or above what they are worth. The view that holds sway is that unresolved information asymmetries have adverse repercussion: "*potentially lead to a breakdown in the functioning of the capital market*" [66]. The main vehicle with which these twin issues are tackled is through financial reporting. This necessitates management to communicate firm performance and governance to outside stakeholders. As Beyer et al [67] argue the demand for accounting information by outsiders arises for two reasons:-

- Management have more information about the expected profitability of firms' current and future investments than outsiders. This information asymmetry makes it difficult for outside capital providers to assess the profitability of the firm's investment opportunities. This problem is exacerbated because insiders (both managers and owner-managers) have incentives to exaggerate their firms' projected profitability. Capital providers also cannot assess firms' profitability, they will under-price firms with high profitability and over-price firms with low profitability, potentially leading to market failure.
- The separation of ownership and control results in capital providers not having full decision making rights. Investors who have confidence in a firms financial reports would then require lower rates of return.

Merkl-Davies and Brennan [30] argue that from a financial reporting perspective, information asymmetry gives management leeway to engage in impression management. This is an attempt: "*to control and manipulate the impression conveyed to users of accounting information*" [68]. This opportunity for impression management is increasing as the narrative section of financial statements are growing in importance and length [30]. An illustrative excerpt from Enron, a firm known to have committed FSF is shown below.

Extract from Enron's Letter to Shareholders, Annual Report 2000 (emphasis added by Merkl-Davies and Brennan [30])

"Enron's performance in 2000 was a ***success by any measure****, as we continued to* ***outdistance the competition*** *and* ***solidify our leadership*** *in each of our major businesses. In our largest business, wholesale services, we experienced an* ***enormous increase*** *of 59 percent in physical energy deliveries. Our retail energy business achieved its* ***highest level ever*** *of total contract value. Our newest business, broadband services,* ***significantly accelerated*** *transaction activity, and our oldest business, the interstate pipelines, registered* ***increased earnings****. The company's net income reached a* ***record*** *$1.3 billion in 2000"*

As Merkl-Davies and Brennan [30] indicate the highlighted phrases are all added as impression management tactics to portray the firm in a positive light and the claims stated sound grandiose.

How issues such as agency, information asymmetry, impression management impact financial reporting are explained from two perspectives. One is the Efficient Market Hypothesis view. This states that all market participants have rational expectations about future returns thus the market is able to assess reporting bias [69]. Agency theory focuses in on the relationship between managers and investors which is characterized by contractual obligations and utility maximisation. Both managers and shareholders are regarded as rational, self-interested decision-makers [70]. Under this view biased reporting such as impression management would lead to higher cost of capital and reduced share price performance. The incentives for biased reporting is reduced as users driven by utility maximisation, are able to detect bias, regarded as "*cheap talk*" [71]. As managers' compensation is linked to stock price performance, managers have no economic incentives (based on cost-benefit analysis) to engage in impression management. Instead managers provide discretionary narrative information to overcome information asymmetries to lower the cost of capital, enhance

share performance and thus increasing managerial compensation. Under this school of thought impression management does not exist, instead managers release value relevant incremental information. Managers are assumed to have economic incentives to engage in unbiased reporting as it enhances their reputation and compensation [70]. The alternative view from the behavioral finance perspective is that management disclosure is opportunistic and driven by self-interest. Poor firm performance gives rise to conflicts of interest between managers and shareholders. This prompts managers to manipulate outsiders' perceptions of financial performance and prospects. This opportunistic managerial behaviour results in concealment and attribution. The latter can be achieved in two ways by obfuscating failures and emphasizing successes. Research that investigates positive bias presumes that "*sections of the [annual] reports are allegedly managed so as to present management in as favorable a light as possible*" [30].

Attribution is a: "*tendency of people to attribute successes to their own abilities but failures to external factors*" [31]. In a financial reporting context this manifests as managers attributing positive organizational outcomes to internal factors and negative organizational outcomes to external factors [31].

The linguistic cues to deception (see Table 2.1) also indicate the deceivers use obfuscation to reduce readability and for distancing themselves from their narratives.

Figure 2.1 extracted from Merkl-Davies and Brennan [70] shows the possible ways attempts at impression management filter into financial reporting. This bias ties in with linguistic cues shown in Table 2.1. According to Merkl-Davies and Brennan [70] Figure A.1 in Appendix A shows the type of information affected by impression management (verbal/numerical) and the types of manipulation (presentation/disclosure of information) and the type of impression management strategies examined in prior accounting research. The two dominant interpretation on factors that impinge on financial reporting, based on the description above is shown in Figure 2.1.

On the other side how do users respond? Psychological research [72] shows that economic actors suffer from cognitive biases that results in bounded rather than pure rationality [73]. Bounded rationality takes into account that economic actors make decisions based on incomplete information, by exploring a limited number of alternatives, and by attaching only approximate values on outcomes [74]. Decision-making in the real world is not determined by: "*some consistent overall goal and the properties of the external world, but rather by the inner environment*" [73] of people's

Deceptive Linguistic Cues	The effect in text	Authors	Theory/Method
Word quantity	Could be higher or lower in deceptive text. Generally, higher quantities of verbs, nouns, modifiers and group references.	Zhou [123]	Interpersonal Deception Theory
Pronoun use	First person singular pronouns less frequent, greater use of third person pronouns. This is known as distancing strategies (reducing ownership of a statement).	Newman et al [127] Zhou [123]	Interpersonal Deception Theory
Emotion words	Slightly more negativity, greater emotional expressiveness.	Newman et al [127]	Leakage Theory
Markers of cognitive complexity	Fewer exclusive terms (eg but, except), negations (eg no, never) and causation words (eg because, effect) and motion verbs - all require a deceiver to be more specific and precise. Repetitive phrasing and less diverse language is more marked in the language of liars. Also, more mention of cognitive operations such as thinking, admitting, hoping.	Newman et al [127] Hancock et al [132]	Reality Monitoring
Modal verbs	Verbs such as would, should, and could lower the level of commitment to facts.	Hancock et al [132]	Interpersonal Deception Theory
Verbal non-immediacy	"Any indication through lexical choices, syntax and phraseology of separation, non-identity, attenuation of directness, or change in the intensity of interaction between the communicator and his referents." Results in the use of more informal, non-immediate language.	Zhou [123]	Interpersonal Deception Theory
Uncertainty	"Impenetrable sentence structures (syntactic ambiguity) or use of evasive and ambiguous language that introduces uncertainty (semantic ambiguity). Modifiers, modal verbs (e.g., should, could), and generalizing or "allness" terms (eg "everybody") increases uncertainty."	Zhou [123]	Interpersonal Deception Theory
Half-truths and equivocations	Increased inclusion of adjectives and adverbs that qualify the meaning in statements. Sentences less cohesive and coherent thereby reducing readability.	McNamara et al [50] Bloomfield [137]	Management Obfuscation Hypothesis
Passive voice	Increase in use, another distancing strategy - switch subject/object around.	Duran et al [141]	Interpersonal Deception Theory
Relevance manipulations	Irrelevant details.	Duran et al [141] Bloomfield [137]	Management Obfuscation Hypothesis
Sense based words	Increase use of words such as see, touch, listen.	Hancock et al [132]	Reality Monitoring

Table 2.1: Linguistic cues to deception.

minds, both their memory contents and their processes'. This results in satisfactory, rather than optimal outcomes.

Kahneman [75] uncovered two modes of thought. There is system one – which is fast, automatic, associative, intuitive and there is system two which is slow and deliberate. Both systems are subject to behavioral biases but Kahneman warns of system one, which operates on WYSIATI ("*what you see is all there is*") is something that many individuals fall prey to. It is subject to a whole suite of irrational biases [75]. For example:-

- Prospect theory: How the message is relayed, influences the way it is processed, known as "*framing effects*" [75]. Investors move away from framing options that relay loss, they will take more risks to avoid loss.
- Functional Fixation Hypothesis: Unsophisticated investors are assumed to be incapable of "*unscrambling the true cash flow implications of accounting data*" [76].
- Schrand and Walther [77] and Frederickson and Miller [78] confirm such

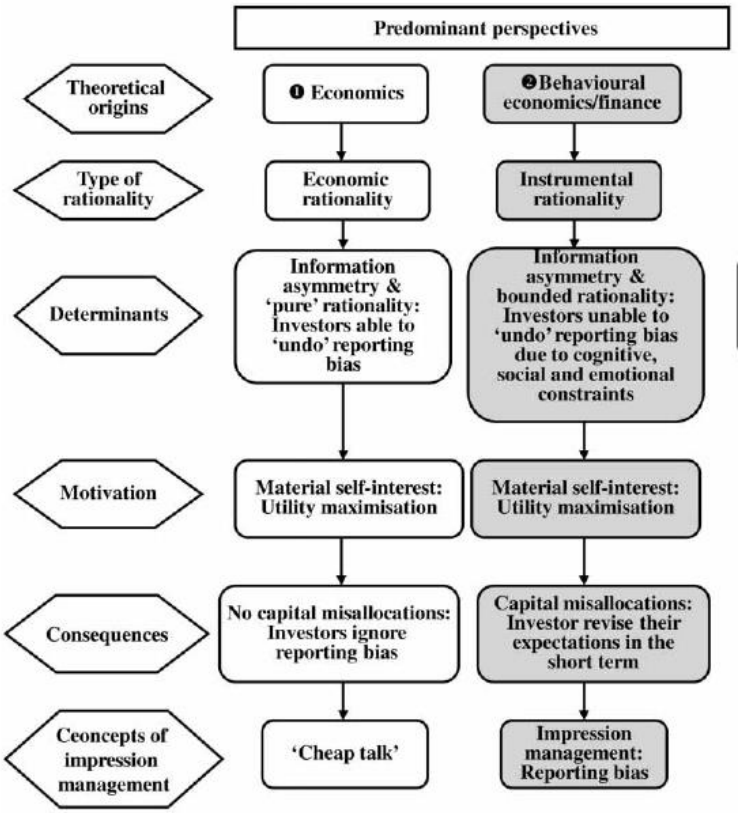


Figure 2.1: Theoretical underpinnings in financial reporting [70].

susceptibility due to information processing limitations.

- Incomplete Revelation Hypothesis: "*The easier information is to extract the more it is impounded into share prices*" Merkl-Davies et al. [30], Li [79], Bloomfield [80] therefore argue that managers would make financial narratives difficult to read to mask poor performance. It achieves the aim of confusing the reader.
- Belief-adjustment model [81]: Using this model, Baird and Zellin [82] show that users' perceptions of firm performance and prospects are more influenced by information presented first (primacy effect). Based on this model other influencing factors are the complexity of the information, the length of the information set (short vs. long), the consistency or inconsistency of the information components.

Therefore, it is a truth not so self-evident that humans are unaware of how they are being manipulated. Biased reporting can trigger modes of thought in investors (system one/system two) that can lead them badly astray. This is exacerbated as information abundance piles on, whereas human information processing abilities remain constant. Some salient research that has examined financial narratives to gauge managerial motivations and company performance is mapped out on Table A.1 in Appendix A.

Content analysis is the main textual analysis approach used within research shown in Table A.1. It involves: "*draw*[*ing*] *inferences from data by systematically identifying characteristics within the data"* [83]. Within content analysis, two approaches are typically taken: form orientated analysis, which involves routine counting of words or concrete references and meaning orientated analysis, which focuses on analysis of the underlying themes in the texts under investigation [84]. Form orientated can be quantitative, use of proxies is common to enable statistical analysis or it can be qualitative, searching for occurrence of predefined content categories within texts [85]. The main problem with these approaches is that it requires human interaction and judgement and is thus subject to reliability issues (such as inter coder agreement on terms), it is error prone and expensive. Researchers have also used both general purpose dictionaries [86, 87] and custom financial dictionaries [88-90] to decipher content.

As outlined by Slattery [92] these studies emanate from the intuitive recognition of a link between the textual report content and corporate performance. This is clear from the findings in research outlined in Table A.1, Appendix A. The narratives are correlated with variables, some of which are proxied for example Kothari et al [86] used the cost of capital, stock return volatility and analyst forecast dispersion to proxy for firm risk. In the majority of cases it is found that the narratives have information content of a predictive nature (see Table A.1, Appendix A) or that it sheds light on management actions or explains industry specific disclosure practices. In all the text analysis of narratives conducted the researchers were keen to pick up forward looking information and tone as it relays messages that can be significant for stock markets and industry analysts. This was primarily done using keyword searches, which can be improved upon by concept based opinion mining tools [11].

2.4 Financial Fraud and Financial Statement Fraud (FSF)

Financial fraud encompasses the ever growing ways and means that criminals through treachery and lies extract funds from an unsuspecting investing public for personal gain. It can be defined: "*as the intentional use of illegal methods or practices for the purposes of obtaining financial gain*" [93]. Figure 2.2, the top half depicts the main categories of financial fraud. Cost of fraud to major economies climb high, into

hundreds of billions. In 2012, Association of Certified Fraud Examiners (ACFE) reported that the U.S. organizations lose almost 5 percent of their revenue due to fraud [94]. Whilst in the UK research indicates that insurers pay out 1.6 billion pounds due to fraudulent claims [1]. A recent report [1] puts the annual cost of fraud to the UK economy at £193 billion a year – equating to more than £6,000 lost per second every day. Such costs are then mitigated by business through higher costs on products and services borne by the unsuspecting consumers unconnected to these financial scams. Given this broad picture on the scale of the problem, it is imperative that anti-fraud measures are researched and rolled out to combat this criminality.

This study focuses in on Financial Statement Fraud (FSF) that comes under the category of corporate fraud (from Figure 2.2). The US Security and Exchange Commission (SEC) state that financial statements should "*provide a comprehensive overview of the company's business and financial condition and include audited financial statements*" [95]. They are the output of an accounting cycle and provide a representation of a company's financial position and periodic performance. Financial statements are a legitimate part of good management and provide important information for stakeholders. Fraudulent financial statements are intentional and illegal acts that result in misleading financial statements or misleading financial disclosure [35, 97, 98]. Financial Statement Fraud (FSF) or "*book cooking*" is a: "*deliberate misrepresentation of financial statement data for the purpose of misleading the reader and creating a false impression of an organization's financial strength*" [2]. According to a study conducted by Beasley et al [99] the two most common techniques used to fraudulently misstate the financial statements involved improper revenue recognition and asset overstatements.

The majority of frauds (61%) involved revenue recognition, while 51% involved overstated assets primarily by overvaluing existing assets or capitalizing expenses. The understatement of expenses and liabilities was much less frequent (31%). Misappropriation of assets occurred in 14% of the fraud cases. These statistics have been consistent over time [34].

FSF is the costliest type of financial fraud as it causes the biggest loss: "*a median loss of $1 million per case*" [34]. Albrecht et al. [104] notes that: "*financial statement fraud causes a decrease in market value of stock of approximately 500 to 1,000 times the amount of money*" [100]. Mohamed et al. [100] cite a case in which a $7 million fraud caused a drop in stock value of about $2 billion. In their 2014 publication, the ACFE

[101] reported that financial statement fraud now occurs in 9% of the cases they studied, and that this figure has progressively increased from 4.8% in 2010 to 7.6% in 2012. Using worldwide ACFE figures [101], the annual cost of financial statement fraud is estimated to be more than $1.2 trillion (US) worldwide.with more than $377 billion in the US [101]. The resultant loss of trust in capital markets and "*confidence in the quality, reliability and transparency of financial information*" [2] is significant. It jeopardises the integrity and objectivity of the auditing profession. It has disastrous implications for jobs, savings and investments. All can be wiped out. The financial industry's meltdown in 2008 is a perfect example of what catastrophe follows when investors lose trust and confidence.

High profile accounting scandals were FSF was involved, such as Enron, WorldCom, Tyco, and Satyam, have cost market participants several billions of dollars, and eroded confidence in published financial statements. It can therefore be deduced that FSF is still ever present and steps that have been taken to stall its advance have not been too successful.

One of these steps was the Sarbanes-Oxley Act enacted in 2002. The Act mandated significant reforms to public companies' governance structures and the oversight of public company accounting firms. Many of its requirements were intended to raise the standard of corporate governance and mitigate the risk of fraudulent financial reporting. Other organisations such as the Public Company Accounting Oversight Board (PCAOB), Association of Certified Fraud Examiners (ACFE) were set up to minimize the occurrence of fraud through training professionals to detect and prevent fraud [2]. Additionally, to improve the audit processes associated with the detection of FSF, the American Institute of Certified Public Accountants' (AICPA) Auditing Standards Board (ASB) released Statement on Auditing Standard (SAS) No. 99 in 2002. Under SAS 99, auditors are required to take a more proactive approach to detecting FSF through improved and expanded audit procedures. In 2014, AICPA redrafted a clarified Statement of Auditing Standards AU-C 240, "*Consideration of Fraud in a Financial Statement Audit*". The goal of AU-C 240 is to increase the effectiveness of auditors in detecting fraud through the assessment of firms' fraud risk factors based on Cressey's (1953) fraud risk theory – discussed below [102]. However, fraud in financial statements/reports can be very difficult to detect. It is uncommon for external auditors to find material misstatements or omissions [103]. They are required to continually question and assess the audit evidence to maintain professional

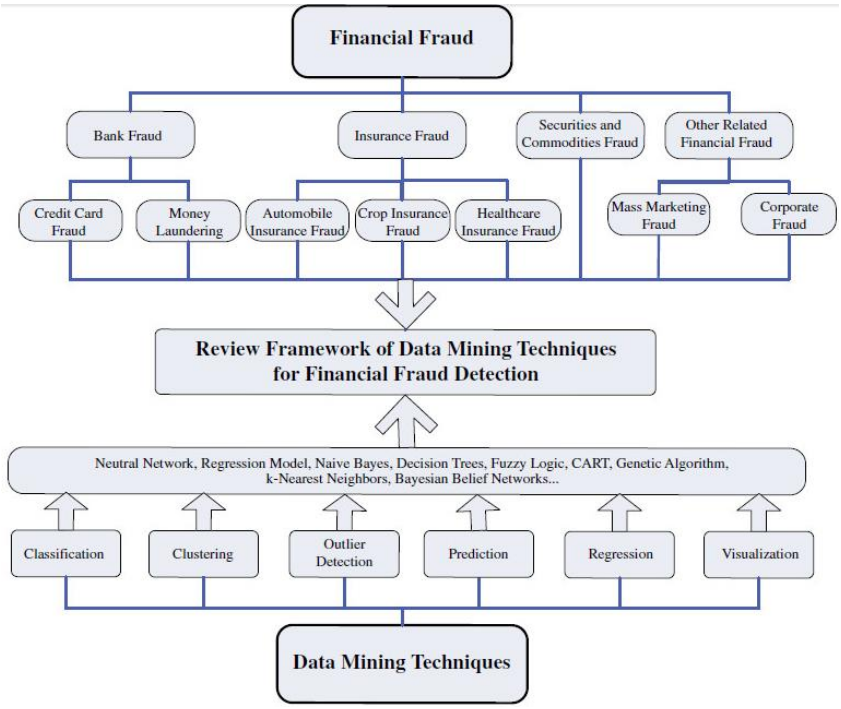


Figure 2.2: The categories of financial fraud and '*intelligent*' detection techniques [96]

scepticism.

Before reviewing approaches that have been utilised for FSF detection, greater insight into this process would be garnered if it could be understood what causes managers to misstate their financial statements.

2.5 The Push to FSF

From a classical, theoretical standpoint, behavioural aspects of fraud have been explained using classical agency theory [104]. The perspective from this theory is that management are motivated by self-interest and self-preservation. Therefore, fraud is committed to further these goals. Hence, to reduce the lure of misdemeanour such as FSF incentives and rewards should be devised that align management behaviour with shareholder goals. However, Albrecht et al. [104] indicate that recent research has attempted to view FSF and other forms of corruption from a more humanistic standpoint. They all shed a light on why top management would be propelled to commit FSF.

However, a theory that is still credited to have good explanatory pull on financial fraud and FSF is known as the fraud triangle (see Figure 2.3). Cressey [105] suggests that

3 attributes must be present for the fraud to occur.

1. *Pressure to commit fraud:* This is most acute when financial goals are known to be unattainable yet pressure to meet analyst expectations, career advancement, compensation are all linked. Research indicates top three reasons for fraud: personal gain, achieving short term financial goals (internal targets or external analyst expectations) and hiding bad news from the capital markets [99]. The research also indicates that desire to recoup or avoid losses is much more of a motivating factor for management fraud than personal gain [106]. Recent research by Schuchter and Levi [107] and Huang et al. [106] indicate that this attribute in the fraud triangle is the most salient and most powerful in pushing management to fraud. From a questionnaire study undertaken by Huang et al. [106] found that the propellants that lead to fraud centred on the following topics: "*poor performance*", "*the need for external financing*", "*financial distress*", "*insufficient board oversight*", and "*competition or market saturation*". The authors assert that such insights would enable auditors and managers to critically evaluate their business processes and would aid in fraud detection.

2. *Opportunity for Fraud:* Without this factor, fraud would be stalled even when pressure is extreme. This has 2 aspects: the inherent susceptibility of the company's accounting to manipulation and the conditions within the company that may allow a fraud to occur. Poor internal controls give great leeway for management to manipulate transactions such as improper use of journal entries, misuse of management discretion over bad debts and expenses, misaccounting of unusual one time but significant transactions [2]. Albrecht et al. [104] adds that opportunity is largely about perceiving that there is a method for perpetuating fraud that is undetectable. Therefore, he argues that auditing procedures should strive to reduce the perception of opportunity by implementing systems and controls.

3. *Rationalization of Fraud*: Under this tenet the pressure to commit fraud results in a mind-set that justifies it. Albrecht et al. [104] argue that people are basically honest and suffer considerable, significant, cognitive dissonance and negative affect at the prospect of committing fraud. However, as the pressure increases, individuals construct rationalization for fraudulent actions. They often conclude it is the only course of action to save jobs or simply to keep the company afloat and self-assurances like "*everyone is doing it*" pushes them to a criminal act. Though the Center for Audit Quality [108] leave it as an open question: Why do people commit fraud? Is it a function of circumstances? Or is it a fundamental character flaw?

Classical fraud theory (the fraud triangle) indicates that fraud is most likely to take place when all three elements are perceived by the potential perpetrator [104]. However, the three factors work together interactively so that if more of one factor is present, less of the other factors need to exist for fraud to occur [104]. Roden et al [102] empirically test the risk factors in the fraud triangle by developing variables that serve as proxy measures for opportunity, pressure and rationalization. They extract characteristics from 103 firms with fraud violations with a matched sample of 103 control firms. They find significant explanatory variables representing all 3 sides of the fraud triangle. Their overall observation is that fraud violations are more common when the board of directors has fewer women, longer tenure, more insiders and CEO is also the chairperson. Fraud is also more likely when the managers and directors are compensated with stock options and when there has been a recent auditor change. A questionnaire study by Ani [109] where 70 companies listed on the Nigerian stock exchange in 2007 revealed that there is a relationship between financial reporting fraud and company size, weak audit committees, internal control and auditor's independence. The study established a positive relationship between these variables. The larger the size of a company and the ownership structure, the more difficult it is for the management to engage in acts that could lead to fraudulent financial reporting. Lokanan [110] cautions against the unquestioning uptake of tenets of the fraud triangle into use as a vehicle for deterrence, as done by the Association of Certified Fraud Examiner's (ACFE). This view he argues presents a restricted version of fraud. Using case based evidence he asserts that fraud is a multifaceted phenomenon whose contextual features may not fit into a particular framework. Therefore the reliability of the fraud triangle should not be held as sacrosanct.

Variation on the fraud triangle also exist, a prominent one is given by Rezzae and Riley [2]. They use a model consisting of conditions, corporate culture and choice in explaining the pressures, opportunities and rationalisations for FSF. They argue that in particular FSF will occur if the benefits to the fraudster outweigh the associated costs (the probability of being caught). Ripe conditions for FSF would be: economic pressure, a downturn in organisational performance and economic recession. A corporate culture with poor corporate governance is again ripe for FSF. Ultimately Rezzae and Riley [2] argue that management make a knowing decision to commit FSF and in some cases it is committed as a strategic tool motivated by aggressiveness, lack of moral principles or misguided creativity/innovation. The authors argue that a

combination of these 3C's result in FSF, as shown in Figure 2.4. Gepp and Kumar [111] provide further enhancements to the fraud triangle by proposing a new framework. They modify the opportunity factor and call it Exploitable Opportunity. The authors define this as opportunity, given that the people concerned have the capability of committing the fraud. They add that it also incorporates the concept of being capable of concealment and the perception of being capable. The authors argue that it would also benefit future research to study how many senior managers have the necessary capabilities to commit financial statement fraud (given the opportunity). This is added to the new framework.

A broader definition of the pressure factor that includes incentives and MICEis included in the new fraud triangle. The authors define the acronym MICE as Money, Ideology, Coercion and Ego or Entitlement. Gepp and Kumar [111] argue that more money and boosting ego are the common drivers and were present in high profile cases such as Tyco, Enron and WorldCom [112]. An example of Coercion is a mid-level accountant in WorldCom being ordered to make false accounting entries [112]. A less-frequent motivation is ideology. To illustrate this aim, authors give an example of HealthSouth. This was a large public company in the US that was able to falsify its financial statements for eleven years without discovery. The senior management considered that falsification of financial statements helped them provide life-saving equipment to hospitals. The rationalisation factor is renamed integrity/attitude rationalisation factor to embrace the importance of personal integrity and attitude in fraud cases.

Gebb and Kumar [111] add a new factor to the fraud triangle called suspicious information category. They argue that it would also be possible to detect fraud by finding unusual/suspicious patterns in data that occur as a result of fraud, separate from the precursor conditions. They cite an example that it can be suspicious if a company is growing at a fast rate financially, but non-financial variables are remaining constant such as stable number of employees. The new triangle proposed by the authors is shown in Figure 2.5. An important point that the authors make on the new triangle that differs from the original is that only one factor in the framework needs to be present for there to be a concern that fraud has occurred. The authors cite evidence by Dorminey et al [112] to support this as it is argued that fraudsters only require an opportunity to commit fraud.

The significant output from Gebb and Kumar's [111] research is that they provide

concrete variables that fall under each of the categories shown in their new fraud triangle. For example, under exploitable opportunities variables related to the board of directors include size, composition and share-holdings, Auditor Big 4 (yes/no), stock exchange listing of company. The pressure/incentive (I) factor includes variables that measure sales growth compared to industry average, cash sales, sales relative to total assets, return on equity. The Integrity/Attitude/Rationalisation (R) factor includes variables that measure the use of operating leases as a proxy for managers who are more focused on short-term window dressing that might more easily rationalise committing fraud [113]. The Suspicious Information (S) factor includes variables that measure for example either the change in assets or in sales compared with the change in the number of employees. The authors list a number of other variables both quantitative and non-quantitative that fall under the categories in the new fraud triangle. These variables can be used to empirically collect evidence on firms to develop predictive models that can alert to anomalies in a firms financial reporting.

2.6 Financial Fraud and FSF Detection

From the above outline it can be gathered that financial fraud and FSF is surging ahead uncontained. The questions that need to be addressed are: For FSF how best can investors, auditors, financial analysts, and regulators detect misstatements?
How can the opportunity outlined above be reduced?
FSF is difficult to detect [103, 111]. Peng [114] cite a company Healthsouth (elaborated above), whilst others such as Sunbeam, Tyco, Enron and WorldCom were also able to continue this misconduct for a prolonged period of time without detection [111].
As Humpherys et al. [103], Gebb and Kumar [111], West et al. [93] argue there is a need for better decision aids to help detect financial statement fraud because research has shown human beings have only a slightly better than random chance ability at detecting deception [115]. Further, as put by Humpherys et al. [103] most external auditors do not have a lot of experience in fraud detection and their impartiality has been questioned in high profile fraud cases [99]. Therefore, there is a pressing need for decision aids that are unbiased and rigorous and that can aid in fraud detection.
Humpherys et al. [103] argue that the difficulty of detecting fraud is further exacerbated by the fact that financial statements can be misleading even if they are in conformity

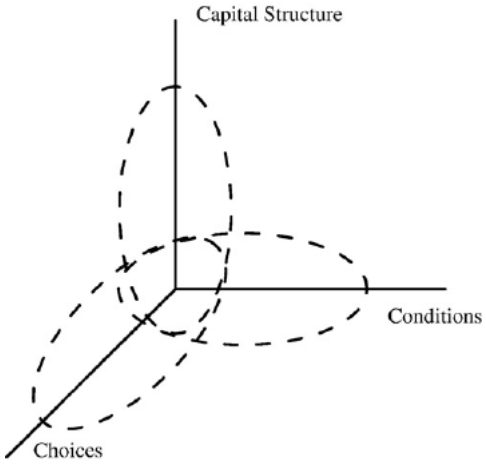


Figure 2.3: The 3 C's model [2]

with GAAP (Generally Accepted Accounting Principles). The rules outlined by for example US GAAP cannot cover every conceivable situation. It is possible for the companies to be creative in financial measurements as well as in the narrative disclosures. Auditors should examine both the quantitative and qualitative sections to perform a compete check on the financial statements. Humpherys et al. [103] argue that the textual narratives in financial reports would not contain explicit indicators of fraud. Instead deception would be camouflaged: "*using rich syntactic as well as semantic arsenal*" [103]. They state that by understanding the nature of deception and how it manifests itself in text and then applying statistical analysis upon the variables identified could prove to be pivotal in uncovering FSF.

Fraud detection models can be used as decision aids to assist in detecting financial statement fraud. Some examples from the academic literature are shown in Appendix A (Table A.2 and Table A.3). Typically, these models commonly assign categories (fraud or non-fraud) by analysing information such as publicly available financial and accounting ratios derived from data in financial statements. These models can act as early indicators of potential anomaly with regard to firms financial reporting [113]. As Gepp and Kumar [111] argue that fraud detection models can be used as a first step to quickly highlight the cases with the highest likelihood of fraud. Regulators and auditors can then use these results to prioritize and more efficiently allocate human specialists to investigate individual cases. Thus accurate fraud detection models can reduce the costs and increase the effectiveness of detecting financial statement fraud by facilitating more directly targeted investigations.

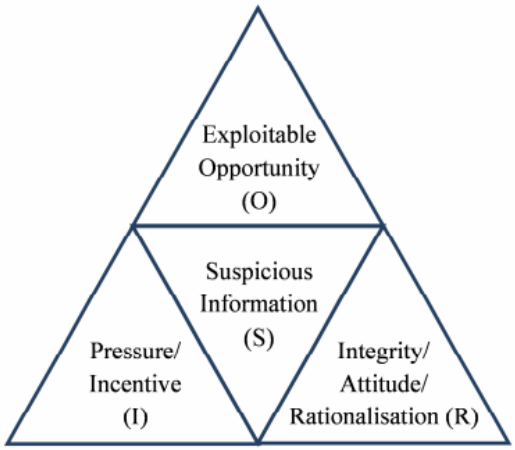


Figure 2.4: A new updated fraud triangle [111]

Ngai et al. [96] map out the different types of financial fraud and categorise data mining approaches and the algorithms used to enact these approaches as shown in Figure 2.2. The data mining based approaches and the main types of financial fraud are also shown in Figure 2.5. As indicated this thesis will concentrate on FSF detection and will use classification and clustering to determine success of these approaches in separating out narratives of fraud from a non-fraud firms.

It has been shown that companies that do not have fraud control mechanisms have losses of approximately 45% median larger than the companies with fraud controls [94]. This could persuade those predisposed to fraud to not to commit such misconduct because of the increased likelihood of detection and punishment.

It is a pressing matter that regulatory framework that oversee financial exchange is strengthened. Weakness in this area can even result in mass disorder and death, as was the case in Albania, 1997. More recently, China is similarly facing mass unrest due to rampant financial fraud due to poor regulation [116]. The 2000 and 2008 financial crises that had a world-wide impact could also have been contained through regulation that dampened expectation through providing more unbiased economic advice [117]. A central component of this process, auditing could be strengthened by performing linguistic analysis using data mining techniques that are elaborated upon in this thesis.

2.7 The Linguistic Correlates of Deception

It is universally acknowledged that the fabricated narrative differs from truthful narrative at all levels. Bachenko and Fitzpatrick [118] point out a few differences cited by previous research: "*narrative structure and length, text coherence, factual and sensory detail, filled pauses, syntactic structure choice, verbal immediacy, negative expressions, tentative constructions, referential expressions, and particular phrasings have all been shown to differentiate truthful from deceptive statements in text*". Deception refers to messages: "*knowingly transmitted by a sender to foster a false belief or conclusion by the receiver*" [119]. It is a deliberate attempt to mislead. Financial misreporting is a particular type of deception intended to deceive a company's stakeholders. However, deception detection is not an easy task. Most laypeople are poor detectors [115]. Human ability to detect deception is only slightly better than chance: typical accuracy rates are in the 55–58% range [115]. Trained researchers, professional lie-catchers are only slightly better [115]. McCarthy et al. [120] point to past research that indicates that the reasons for poor detection is that humans come equipped with a truth-bias, where all statements are initially assumed to be true. The authors further add that training people involves developing insight into "*leakage cues*" which involves examining body language closely and linguistic cues. McCarthy et al. [120] confirm that after examining past research even with training, human performance is still too inconsistent for real world applicability. Enhanced detection methods are needed [115, 121].

A possible way ahead is to use automated deception detection tools that incorporate findings from deception research harnessed using natural language processing technology. Assessing: "*risk is a non-intuitive, humanly-biased, cognitively difficult task*" [122]. Therefore: "*tools that augment human deception detection thereby increasing detection accuracy would prove to be quite valuable*" [122].

2.7.1 The Theories of Deception

To understand the nature of the language used in fraudulent financial narratives, it is necessary to have an appreciation of the theories and practices that underlie

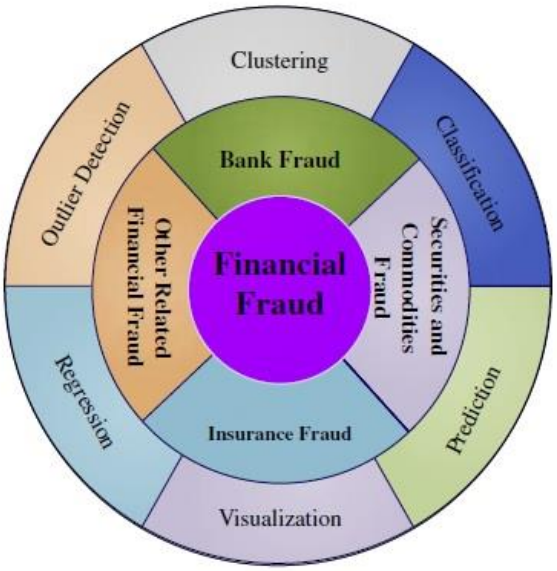


Figure 2.5: Applications of data mining to Financial Fraud Detection [96]

deception and its detection. Some of the prominent ones are elaborated below [103, 123, 124]

Interpersonal Deception Theory

As the name suggest this theory was proposed by Buller and Burgoon [125] to examine deception in an interpersonal context. It sheds light on the strategies and tactics that deceivers may employ to evade detection [123, 124]. From research that uses this theory [103, 123, 124] the following insights have been attached to it:-

- quality (truthfulness) manipulations

This can surface as half-truths and equivocations operationalised by the use of adjectives and adverbs, to satisfy a central tenet of deception, to qualify the meaning in statements. Within the narrative section of a fraud report, Humpherys [103] argues that managers would seek to minimise the number of definitive statements made in order to reduce incriminating evidence.

- quantity (completeness) manipulations

Reticence to divulge a true detailed account of affairs on behalf of deceivers could result in fewer words and sentences in text. This could also manifest through sentences that are incomplete from a syntactic/semantic perspective. The language used may be simpler as it could be devoid of description. An imagined event is less concrete and as it is conjured up may not be repeated accurately. Therefore,

potentially narratives in a falsified report would be less specific with less vocabulary and content than a truthful report.

- clarity (vagueness and uncertainty) manipulations

Text riddled with deception may be less clear as a consequence of dense sentence structure, possibly marked by contradiction. A key ploy used by deceivers is to introduce doubt and uncertainty through the use of evasion and ambiguity. Hedging and uncertainty refers to words, phrases, and constructions that introduce: '*ambiguity, abstruseness, or vagueness*' in a statement [118, 124]. Words such as '*may be*', '*it*', '*this*', *'that*' introduce ambiguity and uncertainty in text. Further modal verbs such as '*could*', '*might*', '*approximately*', '*depend*', '*variable*' are also as Burgoon et al. [124] put it '*semantically empty*'. Generalizing terms such as '*everybody*' also introduce uncertainty and reduces personal responsibility [123]. Additionally, in a financial context, there are additional words that represent uncertainty such as, '*indefinite*', '*speculate*' [126].

As Burgoon et al. [124] argue a strategy of injecting hedging and uncertainty can be used as a mechanism to lessen culpability when securities litigation is brought against management of fraud firms.

- relevance manipulations

Another key ploy for deceivers is to detract from the main issues, a real possibility in a corporate scenario for example to deflect from chronic issues related to profitability and liquidity, managers may introduce irrelevance.

- depersonalism (disassociation) manipulations

Deceivers are known to use distancing tactics. Language is used to remove direct referencing and thereby diffuse responsibility. Non-immediate language such as lack of pronouns, especially first person pronouns, and use of passive voice, use of verb tense, modifiers in a statement reduce a sender's ownership of a statement and/or remove the author from the action being described. Other linguistic features such as use of more second person pronouns may imply dependence on others and lack of personal responsibility. Pronouns are touted as the top 10 linguistic features that distinguish between truths and lies [124]. Use of the first-person singular pronouns are held to signal a likely truthful communicator [127].

- image-and relationship-protecting behavior

This relates to maintaining a credible and trustworthy front. This would result in language that was low on negative emotions and low on any indication of uncertainty.

Firms engaged in fraud will therefore more likely engage in image management and put more of a positive spin on their operations. If necessary, non-fraud firms are more likely to relay negative news.

The theory (IDT) acknowledges the "*superordinate role*" of the context and relationship within which the interaction occurs. An example given by Burns and Moffit [128] is that the situational factors of a 911 homicide call will influence how deceptive exchanges play out, and consequently the hypotheses regarding these exchanges. IDT proposes that: "*the behavior of the deceiver will vary systematically with the spontaneity of the interaction (i.e., lying to a 911 operator requires more dexterity than lying in a written letter) and the immediacy of the context*" [128].

According to Fuller et al.[119] the dimensions of IDT most relevant to text based deception are those related to information management: "*veracity, completeness, directness/relevance, clarity and personalisation*" [119].

Criteria-based content analysis (CBCA)

The backbone to this theory is that a statement derived from memory of an actual experience differs in content and quality from a statement based on invention or fantasy [129]. Therefore truthful narratives contain more details and more references to time, space and feelings, more contextual details than deceptive messages [103, 130]. As described by Burgoon et al. [124] the length of an utterance are indicative of how forthcoming a speaker is. Specificity concerns the amount of detail present in an utterance. Length and specificity can be used to mark out truth from falsehood. Burgoon et al. [124] argue that deceivers could display more reticence (saying less and concealing incriminating information) or the opposite loquacity depending on context. Their findings indicate that in general that deceivers lean towards being loquacious. The authors assert that research indicates that prepared statements of formal language are more likely to include bigger words and sentences with qualifying and clarifying phrases and clauses, regardless of veracity.

Reality Monitoring

Similar to CBCA, it postulates that based on perceptual processes, experience-based memories would contain more sensory, contextual, and affective information than from internally generated falsehoods or imaginations [103, 130]. Nonexperience-based memories would contain more indicators of cognitive operations such as thoughts and

reasoning [131] this enables deceivers to "*spin a yarn*". Therefore, markers of imagined events would be that its description would be more elaborate and irrelevant. Recall of actually experienced events would include external information such as spatial, temporal, sensory, and semantic details [130]. Therefore, an increase in the number of cognitive operations used in a statement could throw into doubt the veracity of that statement [103]. According to Hancock et al. [132] liars may be particularly wary of using distinction markers that delimit what is in their story and what is not. They give the following examples: exclusive words (e.g., '*but*', '*except*', '*without*' and '*exclude*') and negations (e.g., '*no*', '*never*', '*not*') require: "*a deceiver to be more specific and precise, which may increase the likelihood that a deceiver will be caught in a contradiction*" [132].

Scientific Content Analysis (SCAN)

Similar to CBCA, this technique attempts to differentiate between truthful and false statements. The linguistic markers that it tries to identify: lack of description, recount ability, missing links, absence of: connectives, first person singular, past tense verbs raise suspicion on the truthfulness of the narrative. Liars are deemed to experience greater cognitive load. As argued by Hauch et al. [133] when constructing a lie, a convincing scenario has to be communicated. The authors argue that due to the demands for cognitive resources, a lie may not: "*include the complexities and richness of information that characterize reports of real experiences. In contrast, telling a story about a true event relies on retrieval of experienced events. Although this typically involves reconstruction and may at times even take increased effort, recall of episodic memories and supporting details is generally rather automatic*" [133].

Verbal immediacy (VI)

The general construct of immediacy-nonimmediacy refers to verbal and nonverbal behaviours that create a psychological sense of closeness or distance [123, 134]. Verbal nonimmediacy thus entail: "*any indication through lexical choices, syntax and phraseology of separation, non-identity, attenuation of directness, or change in the intensity of interaction between the communicator and his referents*" [123]. An example given by Mehrabian [134] is that while "*you and I selected*" may be equivalent to "*we selected*" in meaning, the former is considered more non-immediate than the latter. One of the most distinguishing linguistic features of deceivers, as expounded in the

literature is that they use fewer self-reference words and reference other people more frequently. First person singular pronouns such as (‘*I*’, ’*me*’, ’*myself*’) signal that the communicator takes ownership of a sentence [124, 127].

According to Zhou [123] VI aids in detecting avoidance strategies by deceivers. It is indicated by non-immediacy categories such as: “*spatial and temporal terms*, *passive voice, presence of modifiers, and other expressions such as volitional words, politeness, and automatic phrasing*” [123].

Furthermore, a passive rather than active voice, and use of past rather than present tense verbs, can lower immediacy as a mechanism to distance oneself from a deceptive act [134]. Burgoon et al. [124] point out that originally under VI view, future tense word usage was also thought to distance oneself from the present. However, in a financial context future oriented language is used to refer to financial projection. Significantly as Burgoon et al. [124] observed safe harbor provisions in securities legislation provide protection for management with respect to forward looking projections, which may provide an additional vehicle for avoiding ownership. Burgoon et al. [124] argue that future-oriented statements when combined with other assertive linguistic forms could convey confidence and correlate with certainty markers. The relationship between future verb tense aligned with certainty and immediacy measures should be examined to determine truth or deceit.

Four Factor Theory

This theory delineates four processes described below that underlie deceivers’ behaviors [128].

Control - how deceivers control or suppress their behavior to try to conceal their deception. For example, in a financial reporting setting, managers will manage the linguistic features of interaction with stakeholders in order to appear as truthful as possible and not to induce suspicion.

Arousal - refers to various autonomic arousal responses of the deceiver’s central nervous system that coincide with the deceptive behavior or story.

Felt emotion - registers emotions that deceivers experience, for example guilt, anxiety, and/or “*duping delight*” entails satisfaction in successfully executing the deception. An example given by Burns and Moffit [128] given the negative feelings associated with guilt, deceivers try to disassociate themselves from their crime by referring to others rather than to the self through a greater use of third-person pronouns. According to

Ekman [135] when people lie they may experience feelings of guilt and fear. This may elicit verbal and nonverbal cues to deception [131]. Typically, it is thought that those engaged in deception display (both written and verbal forms) greater negative emotion. Burgoon et al. [124] find that deceivers tend to use more emotional expressiveness (both negative and positive verbal tokens of emotion, such as happy and sad) compared to truth tellers.

Anxiety - increase in this factor could impair the level of control that deceivers execute to conceal their deception. Deception results in an increased mental burden and more cognitive processing required to perpetuate/maintain a lie and to retain credibility.

Management Obfuscation Hypothesis (MOH)

A key stratagem used by management of firms that have committed fraud is obfuscation. This is defined as: "*a narrative writing technique that obscures the intended message, or confuses, distracts or perplexes readers, leaving them bewildered or muddled*" [136]. Reduced readability in text would blunt the underlying ill health of firms that often marks fraud firms. According to Bloomfield's [137] *"Incomplete Revelation Hypothesis*" (IRH) which asserts that information that is costly to extract from public data is less completely revealed in market prices. Therefore, knowing this MOH states that management would manipulate transparency by reducing clarity of the written narrative disclosure. MOH derives from agency and signalling theory. As Rutherford [138] argues: "*Agency theory holds that, in an environment in which their remuneration and wealth is linked to the financial performance of the companies that employ them, managements have economic incentives to disclose messages conveying good performance more clearly than those conveying poor performance. Signalling theory holds that, in an environment of information asymmetry, companies whose performance is superior to that of the market as a whole will seek ways of signalling the superiority of that performance, such as disclosing it with greater clarity. Hence, 'we would expect ... that good financial performance will be associated with a clear and readable ... narrative, and an obscure narrative with bad financial performance*". A number of studies have look at examining obfuscation in text using readability measures such as Gunning Fog [79, 124, 139]. These measures are based on variables such as number of words and syntactical complexity (quantified often by sentence length and average number of syllables) employed, hypothesising that obfuscation involves composing text that is more

syntactically complex [138]. Readability research is summarised in section 2.7.2. The implication is that longer sentences and longer words are indicative of complexity and possibly fraud.

In sum the above insights garnered from theory and practice into the nature of deception have offered concrete cues to identify. Specifically the main linguistic based cues that can be extrapolated from the above exposition on deception has been mapped onto Table 2.1.

2.7.2 Readability

Those that hold office in public life and use public funds to offer goods and services should be held accountable is uncontentious. The stakeholders (tax-payer, shareholders, investors and others): "*have a right to know*" [140]. Full disclosure on operations and use of finances that leads to transparency has been hailed as a means to achieve accountability. This can reverse distrust in government, improve program efficiency and help fight public corruption. In some cases, enhanced transparency calls have been heeded as more digital data grows**.**

However, as Baraibar-Diez et al. [140] point out that expanded transparency initiatives are not accompanied by information that is comprehensible. As Baraibar-Diez et al. [140] indicate that poor comprehensibility of financial narratives is due to poor readability. This is defined as: "*an inquiry into what properties of texts help or hinder communication*" [142]. Within a financial reporting domain, it has been defined as: "*an ability of individual investor and analysts to assimilate valuation relevant information from a financial disclosur*e" [143].

Bailin and Grafstein [142] argue that readability can best be understood through three basic concepts related to textual comprehension:-

- Linking units of information, this refers to the ability of the reader to connect units of information on the word, sentence and discourse level.
- Ambiguity in text leaves its traces through poor interpretability where word, sentence or a discourse can have multiple meanings.
- Contextual Knowledge refers to any knowledge that the reader uses to make inference from a segment of the text.

Carstens et al. [144] tested the readability of financial narratives on government websites to using the Flesch reading ease formula. This assesses the grade-level reading skills required for users to understand written material and is a standard readability formula (Appendix B, Table B.1 and Eq. 4.4). It thereby gives an indication of the how narratives in English are difficult to understand. All other commonly used readability tests use similar variables to arrive at scores for text clarity [50]. Typically, this involves variables that relate to the frequency of the word in the language and the length of the sentence. Given the result Appendix B, Table B.1 is typically used to rank the text on readability.

Carstens et al. [144] found that increased measures to increase transparency does not result in increased comprehensibility of the text as measured by Flesch score. Other studies have also found that even where regulation has stipulated for greater transparency, poor readability or "*technical opacity*" of text has still resulted in investors being no wiser on the financial implication of their investments [145]. This study indicates the continuing hold of readability measures to ascertain text complexity. These measures have been criticised for being limited and incomplete.

McNamara et al. [50] argue that such unidimensional metrics provide only a reasonable first approximation of scaling text on difficulty. They put forth an alternative approach which adopts a multilevel theoretical framework for language and discourse processing. Central to this framework is the concepts of cohesion and coherence in text. '*Cohesion*' refers to the connectedness of concepts present in the text. It helps to generate order by tying: "*together the clauses and sentences in text at a semantic level and thus helps the reader better understand the ideas in the text*" [50]. Whereas '*coherence*' refers to the: "*connectedness of mental representations that readers are likely to construct from the text*" [50].

The multilevel theoretical framework they propose would contain 6 levels relating to: word, syntax, the explicit textbase, the referential situation model, the discourse genre and rhetorical structure (the type of discourse and its composition) and the pragmatic communication level (between reader and writer) [50]. These ideas were encapsulated into a tool named Coh-Metrix. McNamara et al. [50] argue this is a much more robust tool to use to measure readability. This tool outputs 110 indices that rigorously probe the text for readability. The main categories that contain these indices are delineated below and are expounded in [50]:

- Descriptive Indices

These give general metrics on the corpus or the set of text under study. This includes measures such as number of paragraphs, number of sentences, number of words etc. The indices that fall under this category are shown in Table B.2, Appendix B.

- Text Easability Principal Component Scores

These components provide a more complete picture of text ease (and difficulty) that emerge from the linguistic characteristics of texts. The indices that fall under this category are shown in Table B.3, Appendix B.

- Referential Cohesion

This refers to overlap in content words between local sentences or co-reference. Co-reference is a linguistic cue that can aid readers in making connections between propositions, clauses, and sentences in their textbase understanding [146, 147]. Coh-Metrix measures for referential cohesion vary along two dimensions. First, the indices vary from local to more global. Local cohesion is measured by assessing the overlap between consecutive, adjacent sentences, whereas global cohesion is assessed by measuring the overlap between all of the sentences in a paragraph or text [50]. Table B.4 in Appendix B details the indices used to measure referential cohesion.

- Latent Semantic Analysis

Latent Semantic Analysis [148] provides measures of semantic overlap between sentences or between paragraphs. Coh-Metrix 3.0 provides eight LSA indices. Each of these measures varies from 0 (low cohesion) to 1 (high cohesion) of text [50]. Table B.5 in Appendix B details the indices used to measure latent semantic analysis.

- Lexical Diversity

Lexical diversity refers to the variety of unique words (types) that occur in a text in relation to the total number of words (tokens). When the number of word types is equal to the total number of words (tokens) then all of the words are different. In that case, lexical diversity is at a maximum, and the text is likely to be either very low in cohesion or very short. A high number of different words in a text indicates that new words need to be integrated into the discourse context. By contrast, lexical diversity is lower (and cohesion is higher) when more words are used multiple times across the text [50]. Table B.6 in Appendix B details the indices used to measure lexical diversity.

- Connectives

Connectives play an important role in the creation of cohesive links between ideas and clauses and provide clues about text organization [149]. Coh-Metrix provides an incidence score (occurrence per 1000 words) for all connectives as well as different types of connectives. Table B.7 in Appendix B details the indices used to measure connectives.

- Situation Model

The expression situation model has been used by researchers in discourse processing and cognitive science to refer to the level of mental representation for a text that involves much more than the explicit words [150]. Some researchers have described the situational model in terms of the features that are present in the comprehender's mental representation when a given context is activated [151]. Table E.7 in Appendix B details the indices used to measure connectives. Table B.8 in Appendix B details the indices used to measure the situation model.

- Syntactic Complexity

Theories of syntax assign words to part-of-speech categories (e.g., nouns, verbs, adjectives, conjunctions) group words into phrases or constituents (noun-phrases, verb-phrases, prepositional-phrases, clauses), and construct syntactic tree structures for sentences. Some sentences are short and have a simple syntax that follow an actor-action-object syntactic pattern and have few if any embedded clauses they follow an active rather than passive voice. Some sentences have complex, embedded syntax that potentially places heavier demands on working memory. The syntax in text tends to be easier to process when there are shorter sentences, few words before the main verb of the main clause, and few words per noun phrase [50]. Table B.9 in Appendix B details the indices used to measure syntactic complexity.

- Syntactic Pattern Density

Syntactic complexity is also informed by the density of particular syntactic patterns, word types, and phrase types. Coh-Metrix provides information on the incidence of noun phrases (DRNP, verb phrases (DRVP), adverbial phrases (DRAP), and prepositions (DRPP). The relative density of each of these can be expected to affect processing difficulty of text, particularly with respect to other features in a text. If a text has a higher noun and verb phrase incidence, it is more likely to be informationally

dense with complex syntax [50]. Table B.10 in Appendix B details the indices used to measure syntactic pattern density.

- Word Information

Word information refers to the idea that each word is assigned a syntactic part-of-speech category thus, syntactic categories are segregated into content words (for example nouns, verbs, adjectives, adverbs) and function words (for example prepositions, determiners, pronouns). Many words can be assigned to multiple syntactic categories. For example (as given by McNamara et al. [50]) the word '*bank*' can be a noun ("*river bank*"), a verb ("*don't bank on it*"), or an adjective ("*bank shot*"). Coh-Metrix assigns only one part-of-speech category to each word on the basis of its syntactic context. In addition, Coh-Metrix computes word frequency scores and psychological ratings [50]. Table B.11 in Appendix B details the indices used to measure word information.

- Readability

The traditional method of assessing texts on difficulty consists of various readability formulas. More than 40 readability formulas have been developed over the years. The most common formulas are the Flesch Reading Ease Score and the Flesch Kincaid Grade Level. Table B.12 in Appendix B details the indices used to measure word information.

These indices are derived as described above and by McNamara et al. [50]. Coh-Metrix production of these indices cannot be changed in any way by the user. The user can choose not to use any number of the indices produced but the manner of their derivation is fixed and unalterable.

Coh-Metrix tool was built using a number of tools and techniques used within NLP applications. Significant aspects behind the science and technology that contributed to the development of Coh-Metrix are elaborated below:-

- Lexicons are heavily used. These are dictionaries of words that list qualitative features/quantitative values for each word. For example, WordNet is a large lexical database of English. Nouns, verbs, adjectives and adverbs are grouped into sets of cognitive synonyms (synsets) each expressing a distinct concept. The MRC Psycholinguistic database [152] has: "*human ratings of thousands of words on familiarity, imagery, concreteness and meaningfulness*" [50]. The CELEX Lexical

database is also used. It has estimates of how frequently English words are used in a very large corpus of documents.

- A syntactic parser [153]. This takes each sentence and outputs a syntactic tree structure. This enables a part of speech (POS) tag, derived from the Penn Treebank [154] to be assigned to every word in the text. This is used to derive a measure of syntactic ease or difficulty of text. McNamara et al. [50] argue that syntactic difficulty increases with structural ambiguity: "*with the degree with which sentences have embedded constituents and with the load on working memory. Working memory is taxed when there are noun phrases with many modifiers and when many words must be held in working memory before the reader receives the main verb of the main clause*".
- Statistical algorithms are used to quantitatively measure discourse components. Latent Semantic Analysis – LSA [148] discussed in Chapter 5 is such an algorithm that uses word and world knowledge from a large corpus. LSA similarity values are used in computations of text cohesion and coherence.
- Insights from researchers that work to analyse words, sentences and discourse are incorporated into Coh-Metrix mechanisms and measures.
- A central tenet through which text is examined in Coh-Metrix is through a mechanism known as Textbase. This captures the meaning of explicit information in the text. Van Dijik and Kintsch [155] distinguish between the explicit textbase level and a deeper level called the situation model that contains: "*more inferences and more global conceptualisations*" [50].

According to Van Dijik and Kintsch [155] propositions play a central role in conveying meaning in the textbase. Each proposition contains a textbase (eg main verb, adjective, connective) and one or more arguments (for example nouns, pronouns, embedded propositions) that have a thematic role such as agent, patient, object, time or location.

Below is an example extracted from McNamara et al. [50] that explains propositional meaning representation.

Sentence: When the committee met on Monday they discovered the society was bankrupt.

PROP 1: meet (AGENT = committee, TIME = Monday)

PROP 2: discover (PATIENT=committee, PROP 3)

PROP 3: bankrupt (OBJECT: society)

PROP 4: whwn (EVENT=PROP 1, EVENT=PROP 2)

Arguments within the parentheses have role models, predicates are outside the parentheses. These proposition, clauses and noun phrases are connected by principles of cohesion. Referential Cohesion occurs when a noun, pronoun or noun phrase that captures an argument refers to another constituent in the text. The use of connectives such as (‘*because*’, ‘*in order to’*, ‘*so that’*), transitional phrases (‘*on the other hand’*), adverbs (‘*therefore*’, ‘*afterwards*’) link propositions or clauses improves cohesion in text. Other types of connectives that Coh-Metrix takes account of include connectives that correspond to additive cohesion (‘*also*’, ‘*moreover*’, ‘*however*’, ‘*but*’) temporal cohesion (‘*after*’, ‘*before*’, ‘*until*’), causal/intentional cohesion (‘*because*’, ‘*so*’, ‘*in*’), logical operators (‘*or*’, ‘*and*’, ’*not*’). These are all cohesive links that influence the complexity of the text.

- Coh-Metrix also measures co-reference cohesion or referential cohesion. This occurs when a noun, pronoun or noun phrase refers to another constituent in the text. There is a gap when content words in a sentence do not connect to words in surrounding text or sentences. Coh-Metrix tracks five major types of lexical co-reference by computing overlap in nouns, pronouns, arguments, stems and content words (the indices for these are elaborated in Chapter 4).
- Measurement of lexical diversity is another salient feature of Coh-Metrix. This construct can impede comprehension as each new word introduces new information that needs to be encoded and integrated into the discourse content. This is measured primarily through the type token ratio (TTR, [156]). This is the number of unique words in a text (word types) divided by the overall number of words (tokens). This measure is sensitive to variations in text length because as the number of tokens increase, it increases the possibility of more words being unique, thus affecting the ratio. This can adversely affect comparative studies that look at language use in documents of different length. To counteract this tendency, Coh-Metrix also tracks other indices to standardise comparisons. For example, it uses a measure called Measure of Textual Lexical Diversity (MTLD) which introduces more randomness to the measurement of lexical diversity.

As indicated a deeper level of textbase is the situation model. This moves us from what is being said explicitly in the text to that what can be inferred and the conceptual

meaning relayed. Typically, this has been performed by adding in background world knowledge through ontologies and dictionaries. According to McNamara et al. [50]:*"such generic knowledge packages were thought to be activated during comprehension through pattern recognition processes and to guide comprehension by monitoring attention, generating inferences, formulating expectations and interpreting explicit text. AI researchers quickly learned that it was extremely difficult to program computers to comprehend text even when the systems were fortified with many different classes of world knowledge. Moreover it was tedious to annotate and store large volumes of world knowledge in formats needed to support computation".*

Being cognizant of the above issues, Coh-Metrix developers used LSA (McNamara et al. [50] as described in Chapter 5) to represent world knowledge. Essentially this method taps into word meanings by looking at context of a sentence/paragraph. If two words have similar context then they have similar meaning. LSA is used in Coh-Metrix to compute text coherence at the level of the situation model. LSA similarity scores are computed between adjacent sentences in the text, between all possible pairs of sentences in a paragraph and between adjacent paragraphs. Text difficulty is predicted to increase as a function of decreases in LSA similarity scores. Coh-Metrix using LSA is also able to detect when new information is added to the text. An LSA based metric compares the LSA vector of each incoming sentence to the existing vector of the preceding text. The exact statistical method is called a span [50].

The default corpus used in Coh-Metrix to determine the statistical representation of words is the Touchstone Applied Science Associates (TASA) corpus of academic books. It is a corpus of more than 11 million words and covers a broad range of topics. The situation model in Coh-Metrix is also examined for causality and intentionality, time and space perspective. According to McNamara et al. [50] a break in cohesion or coherence occurs when there is a discontinuity on one or more of these situation model dimensions. They argue that in such circumstances, it is important to have connectives, transitional phrases, adverbs or other signalling devices that convey to the reader that there is a discontinuity. The authors refers to these different forms of signalling as particles. They maintain that: "*cohesion is facilitated by particles that clarify and stitch together the actions, goals, events and states conveyed in the text*" [50].

Intentionality refers to the actions of animate agents as part of plans in pursuit of goals. Whereas causal dimension refers to mechanism in the material world that may or may

not be driven by goals of people. McNamara et al. [50] elaborate on how they pull out the goal orientated plan based situation model:

"identifying clauses in which (a) the noun in the syntactic subject position is human or animate(ie a causal agents) and the main verbs are diagnostic of goals and actions. The syntactic parser isolates the syntactic subject and then WordNet takes over. The subject noun needs to be human or animate according to WordNet, whereas the main verb needs to be in charge of other relevant categories according to WordNet. That is, the verbs are change verbs ("stretch), contact verbs ("smash"), create verbs (build), competition verbs("fight") and communicate verbs ("tell"). All three conditions have to be met in order to classify a clause as being an intentional action or goal. Once this intentional content is extracted from the text, we ask how much of this content is woven together cohesively by causal particles namely connectives (ie "in order to", "to", "so that", "by means of"). Intentional cohesion increases theoretically if the ratio of intentional particles to intentional content is higher. Intentional cohesion is predicted to be inversely related to text difficulty"

The above outline on the technology behind Coh-Metrix and its examination of the text clearly shows its superiority as a tool for assessing readability. This tool will be used to extract the indices (shown in Appendix B) from the corpus to determine if there is a difference in readability between fraud and non-fraud reports.

2.7.3 Automated Linguistic Cues to Deception

Automated text classification methods have been introduced into deception research. The aim is to develop aids to detect deception in text in an automated, parsimonious, free from subjectivity and in a comparative manner. A starting point as proposed by Zhou et al. [123] was to use the theories and methods outlined above and extract linguistic cues to deception. They categorised these cues into 8 constructs [123]. As put by Fuller et al. [119] these constructs can be used to classify text on: "veracity or mendacity". Fuller et al. [119] examined these constructs in order to fully determine the most appropriate constructs for use in studying deception in a high stakes domain. The revised constructs, almost identical to the original and their theoretical foundations are shown in Table 2.2 (extracted from Fuller et al. [119]).

The above constructs or variations thereof have been used to build machine learning based classification models to aid in discriminating liars from truthtellers [103, 123,

128] with good predictive ability and precision. From the descriptions given in Table 2.2, it can be inferred that these constructs are attempting to probe text to pick up the cues as outlined in deception research outlined. For example, as mentioned above, the psychological closeness/distance between a speaker and his or her message might be reflected in language. Liars would display more linguistic markers indicative of psychological detachment than truth-tellers [133]. Uncertainty words have been proposed as markers of psychological distance between a speaker and his or her account [118, 123, 133]. Non- immediacy construct in table 2.2 attempts to determine distancing through examining their use of pronouns and use of certainty words. Complexity attempts to determine comprehensibility of text by typically using readability measures such as the Gunning Fog index.

The linguistic cues to deception that are deemed to be the most prominent in deception research were highlighted by Bachenko et al. [118]:-

- Lack of commitment to a statement or declaration

  This is achieved linguistically through the use of hedges, verbs, nominals, qualified assertions, unexplained lapses of time, rationalization of an action.

  Preference for negative expressions in word choice, syntactic structure and semantics.
- Inconsistencies with respect to verb and noun forms such as verb tense changes, thematic role changes, noun phrase changes, pronoun changes.

Hancock et al. [132] confirm this finding to a large degree they find that previous research suggests that the linguistic cues associated with deception primarily include:

- word quantity
- pronoun use
- emotion words
- markers of cognitive complexity.

As argued by Hauch et al. [133] given the poor ability of individuals both the untrained/trained (in lie detection) computer systems developed to aid in deception detection can be developed/used to fill the gap. A computer system is less prone to the influence of cognitive biases and stereotypes, as previously mentioned. It could perform exhaustive/extensive checks in a rapid, neutral manner. However as Hauch et al. [120, 133] point out that for a computer to be able to detect deception, the linguistic characteristics to be analyzed must be revealing of deception. As they

Construct	Theoretical	Brief Description
Quantity	IDT, IMT	Length of message
Specificity	IDT, Reality Monitoring	Type of details in the message
Uncertainty	IDT, IMT	Relevance, directness, and certainty of message
Diversity	IDT	Variety or redundancy in language
Complexity	IDT, IMT	Message clarity
Non-immediacy	IDT	Psychologically distancing
Personalism	IDT	Amount of self-reference,
Affect	IDT, Reality Monitoring,	Emotional language present
Activation	IDT	Intensity or vividness of language
Cognitive information	Reality Monitoring	Increased or decreased cognitive information

Table 2.2: Linguistic cues that have been automated for deception detection [119].

indicate further research is needed to confirm which linguistic markers are indicative of deception and in what context. Such caution in drawing a definitive list is also put forth by McCarthy et al. [120] as they point out that successful detection is likely to be an "*elusive and fickle prey*".

In order to make an attempt to investigate which linguistic cues would elucidate differences between fraud and non-fraud reports the constructs similar to those shown in Table 2.2 are applied over the corpus. The constructs are based on Zhou et al. [123] original work. The ratios derived based on these constructs however are from those that were deigned to be relevant in the financial domain by Humpherys et al. [103]. The ratio derivation and the tools used are described in chapter 4.

2.7.4 Literature review of recent deception based studies

Burgoon et al. [124] attempted to amalgamate the findings from deception research, in terms of language markers and extract them from quarterly conference calls. They built a corpus of 1114 statements made by a CEO/CFO of one company formally indicted for fraud. These statements were manually annotated as prepared (ie as in a presentation) and unprepared as in (Q&A) responses. From this corpus they extracted constructs listed in Table 2.2. These constructs are key markers of deception, as already discussed. They were put through hypothesis testing in a: "*fully saturated three-way factorial multivariate or univariate analyses of variance with the three*

independent variables of speaker (male CEO/female CFO), preparation (prepared remarks/unscripted Q&A), and fraud (fraudulent restatement-related/non-fraud) and vocalic and linguistic composite measures as dependent variables" [124]. The results of analysis are shown in Appendix A, Table A.4. The authors emphasize that prepared remarks differed in substantial ways from unprepared ones. This factor must be taken into account before attributing any deception or misrepresentation to responses. Previous research into deception has also be divided into strategic and non-strategic deception. The latter which is more plentiful examines uncontrolled and unmonitored aspects of behaviour. This distinguishes inadvertent signals of deceit from more premeditated, deliberate and voluntary communication that is labelled as strategic behaviour.

Burgoon et al. [124] categorically state that fraud-related utterances differed systematically from non-fraud utterances. Their salient finding is that quantity and specificity measures revealed: "*consistent with recent evidence in the political arena by Braun et al. (2015) that fraud utterances were longer and more laden with details than non-fraud ones*" [124]. Also fraud-related utterances had more hedging and uncertainty language and more complexity throughout.

Significantly, results with respect to the Gunning-Fog index, which was deployed to measure comprehensibility, were not so definitive. The authors anticipated less comprehensible language would be used in the Q&A as an obfuscation technique. However they concede that in the light of recent criticisms of commonly applied readability indexes, Gunning-Fog may not be applicable to the financial domain. This is because many common business terms, like "*depreciation*" are multi-syllable, in turn generating high FOG scores [143]. Noting this possibility: "*the high FOG scores we observed in the presentation, regardless of fraud, plausibly reflects the firm simply discussing the verbatim accounting results*" [124]. They also find that in the Q&A, non-fraud utterances have lower FOG scores but fraud utterances had FOG scores of similar magnitude as the presentation. Together, this again suggests the possibility that when discussing fraud topics in the Q&A, management attempted to stay on script.

Van Swol and Braun [157] set up an experiment using 308 undergraduates. A participant (allocator) was given 6 dollars to divide between herself and another participant (receiver). Receivers were not told how much money allocators received. In 1/3 of interactions, the recipient was deceived either with a lie or deceptive omission.

Linguistic differences associated with deception (fewer first person pronouns) were found for lies and omission, but higher word count was only found for omission. The authors find no evidence of a relationship between negative emotion and linguistic factors related to emotion (negative emotion words, negations, pronouns). Coding of justifications found allocators used more justifications for their offers when recipient was suspicious. Liars used more justifications providing details about how they obtained the money.

The authors maintain that they add to research on speech and deception by distinguishing between bold-faced lies and deception by omission. A bold-faced lie represents a: "*deliberate falsification of reality, and omission involves being strategically vague and evasive and withholding information in a manner advantageous to the allocator*" [157]. They argue that previous research into deception by equivocation and omission is limited in comparison to research that examines bold faced lies. According to the authors omission and equivocation are more common in characterising deception and should be studied more extensively. They conclude their study by adding that liars should be asked to justify their actions and this results in increased accuracy in the detection of deception and truth. However Van Swol and Braun [157] state that: "*it is sensible to expect individuals engaged in fabrication to construct detailed accounts to increase plausibility and thereby give receivers the impression of completeness. By comparison, the vague and indirect nature of equivocal answers may necessarily result in impoverished detail*" [157]. The authors maintain that studies into deception show that as a rule that with mendacity comes loquacity. They agree with Kalbfleisch [158] that verbal cues of concreteness, clarity, and plausibility are good indicators to judge deception.

Fuller at al. [119] posed the question: "*What are the appropriate constructs for use in studying deception in text in a high-stakes domain?*" They used Zhou's Linguistic-Based Cues Framework [123] as a starting point and used a revised model based on the original to determine which worked better in a high stakes domain. They analyzed linguistic-based cues extracted from 367 written statements prepared by suspects and victims of crimes on military bases. Confirmatory factor analysis was used to evaluate the two models. The superior model retained seven constructs: quantity, specificity, affect, diversity, uncertainty, non-immediacy and activation.

Hauch et al. [133] collated 79 linguistic based deception cues used in over 44 studies that was based on automated detection. They posed 6 research question to determine

characteristics and behaviour of liars. They found that relative to truth-tellers, liars experienced greater cognitive load, expressed more negative emotions, distanced themselves more from events, expressed fewer sensory-perceptual words and referred less often to cognitive processes. Liars were not more uncertain than truth tellers. They stress though that these effects were: "*moderated by event type, involvement, emotional valence, intensity of interaction, motivation and other moderators*" [133].

Burns et al. [128] used 50 transcribed 911 call (25 truthful and 25 deceptive calls) and extracted the following linguistic cues/LIWC 2007 categories (see chapter 4 for description of tool): (Immediacy/first person plural, first person singular; Non-immediacy/3rd person singular, 3rd person plural; Control/ Assent, Negate; Felt Emotion/Negative emotion, Anxiety; Lack of felt emotion/ extreme swearing; Cognitive overload/Inhibition; Lack of cognitive overload/Numbers. The following classification algorithms were executed over the text: logistic model tree induction, naïve bayes, neural networks and random forests. The overall performance of the classification techniques was very strong and ranged from 70% to 84% for the cross-validation tests. The results yielded predictive models with much higher accuracy than that of unaided humans, which, as mentioned is around 54%. The results indicated that truthful callers display more negative emotion and anxiety than deceivers. They also referred to others in third person singular form and gave more details. Deceivers used third person plural at a higher rate perhaps to deflect blame. They also demonstrated more immediacy than truth tellers by using more first person singular and first person plural pronouns.

Bacheko et al. [118] put 275 propositions (164 verified as False and the remainder True) through a decision tree classifier (see Chapter 6 for description), the results show 75% classification accuracy. The features used were linguistic cues adapted from the literature, outlined above.

Using a contrastive corpus (80,000 words) made up of false and true statements McCarthy et al. [120] find through exploratory data analysis that there is a systematic differences between truthful and deceptive personal accounts. Results suggest that deceivers employ a distancing strategy that is often associated with deceptive linguistic behaviour. They find that deceivers struggle to adopt a truth perspective.

Hancock et al. [132] analysis of 242 transcripts revealed that liars produced more words, more sense-based words (for example seeing, touching), and used fewer self-

oriented but more other-oriented pronouns when lying than when telling the truth. In addition, motivated liars avoided causal terms when lying, whereas unmotivated liars tended to increase their use of negations.

Duran et al. [141] examined conversational transcripts in an attempt to differentiate between truth and deception. They used both LIWC (2001) and Coh-Metrix (both tools described in chapter 4). This is the only study apart from this one that uses Coh-Metrix for deception detection. The authors stress that given: "*the nebulous nature of deception, there is an impetus for researchers to clearly specify the context of the targeted deception, and to use convergent NLP approaches to evaluate the various types of linguistic features*" [141]. They add that using both tools offers a unique and complementary analysis that strengthen any investigation into the nature of deceptive language. The results of their findings are shown in Appendix A, Table A.5.

It can be seen from the table that *total word count*, *negation,* and *personal pronouns* had the same result for both LIWC and Coh-Metrix. According to Duran et al. [141] this convergence confirms that more words are used in deceptive conversations, that there are no differences in the use of negation and that deceptive senders use more third person pronouns. The authors assert that this confirms the relevance to deception detection of the quantity and immediacy constructs shown in Table 2.2. The authors also found using Coh-Metrix statistically significant differences for:

- The change in semantic memory retrieval (*accessibility*). Words used in deceptive discourse were more meaningful, as according to the authors they are more easily retrievable and consequently this reduces the cognitive processing burden of maintaining a lie.
- The change in grammatical phrasing (*complexity*). In Coh-metrix a complex sentence is defined as having more words before the main verb. Duran et al. [141] found that this type of sentences were more marked in deceptive conversations.
- The repetition of given information (*redundancy*). Semantic similarity of words compared and found that words similar in meaning are used more often in deceptive conversations.

Overall Duran et al. [141] results suggest that Coh-Metrix was largely able to reproduce LIWC results (eg in areas of *quantity* and *immediacy*) and provide more indicators than LIWC that are likely to be of discriminatory value (eg *accessibility*, *complexity*, and *redundancy*).

Fornaciari et al. [159] used a corpus of deception based on court transcripts containing both truthful and deceptive testimonies. The study units are not whole document but 1437 utterances. Each utterances is marked for clear truth or falsehood or other categories if unverifiable. This results in 537 (7908 tokens) true utterances and 333 false (5778 tokens) utterances. Seventy-two percent of the corpus was used to train the logistic regression classifiers and the rest used for testing purposes. The features extracted were from LIWC (2007). Results indicate over 60% classification accuracy but poor recall on false utterances. There is a tendency in the model to evaluate all utterances as true.

Larcker and Zakolyukina [160] analyze linguistic features present in Question and Answer (Q&A) narratives of CEOs and CFOs conference calls. A total of 29,663 such transcripts were examined. The underlying assumption is that CEOs and CFOs know whether financial statements have been manipulated, and their unrehearsed narratives provide cues that can be used to identify lying or deceitful behaviour. LIWC (2007) is used to extract features of interest. The authors then build a binomial logistic regression model for the likelihood of deception in quarterly financial statements. Classification accuracy exceeds 60%. The analysis indicates that deceptive executives make more references to general knowledge, fewer non-extreme positive emotions and fewer references to shareholders value and value creation. In addition, deceptive CEOs use significantly fewer self-references, more third person plural and impersonal pronouns, more extreme positive emotions, fewer extreme negative emotions, and fewer certainty and hesitation words. Overall, the results suggest that linguistic features of CEOs and CFOs in conference call narratives can be used to identify deceptive financial reporting.

The above research in deception studies strongly indicate the potential automated text classification methods have in detecting financial misreporting. These automated methods can be highly efficient and scalable and can be another armoury with which law enforcement officials can use to identify misconduct such as FSF.

2.8 FSF – A Literature Review

Research conducted using data mining techniques to detect FSF can be divided into two categories. Studies using non-linguistic data (mostly ratios and in some cases

non-financial numerical data is also included) as descriptive features for the fraud models predominate. Whereas, FSF using linguistic features are less numerous. Salient recent research in both areas is shown in Appendix A, Table A.2 and Table A.3 and is expanded below.

2.8.1 FSF using linguistic features – A Literature Review

A summary of recent FSF detection using linguistic features as input to predictive modelling techniques derived from publicly available data is presented in Appendix A, Table A.3. This identifies researcher with date of publication, data set used, the detection method employed and the results as given in the research paper.

Cecchini et al. [161] develop a methodology to analyze text to detect fraud and bankruptcy outcomes. They do this by creating a dictionary of terms (an ontology) from Management Discussion and Analysis Sections (MD&A) of 10-Ks. This can be used to discriminate between firms that encounter catastrophic financial events. From the results given, these dictionaries were able to discriminate fraudulent from non-fraudulent firms 75% of the time.

Chen [162] conducted a small study using a bag of words model on unigrams with TF-IDF scores. Clustering is then performed on these unigrams. Results are shown in Appendix A, Table A.3.

Dong et al. [163] combines a statistical language model (SLM) with latent semantic analysis (LSA). According to the authors, an SLM represents a probability distribution over a sequence of tokens (n-grams) that reflects how frequently they occur. The authors claim that this combination removes the need to extract linguistic cues and models better the dependency relationships between words in natural language. The LSA component enables better catchment of long-span information in text and extraction of semantic patterns that could discriminate between fraud and non-fraud narratives. Once the model is built using SLM/SLT classification accuracy is checked using 4 classifiers. Best classification results obtained using Neural Nets at 78%.

Dong et al. [164] based their feature extraction method on Systemic Functional Linguistics Theory [43]. The features they extract apart from 3 are from LIWC categories (Appendix A, Table A.3). The other 3 are: topic category (which they capture using latent dirichlet allocation – see chapter 4 for full explanation), fog index

score to measure readability and a TF-IDF weight of significant words. Using these features (no feature selection) over the corpus with SVM classifier they attained results shown in Appendix A, Table A.3.

Glancy and Yadav [165] designed a computational fraud detection model, named (CFDM). The model used the singular value decomposition – see chapter 5 for explanation to reduce term document matrix built on textual data. Document clustering is then attempted to determine if fraud and non-fraud firms separated well. The results show that only 3 reports were clustered incorrectly out of 69 documents.

Goel et al. [37] train a Support Vector Machine (SVM) based learning algorithm using surface language features deemed to be indicative of fraud such as the percentage of passive-voice sentences. Using an SVM classifier they achieved 89% classification accuracy.

Humphreys et al. [103] used Zhou et al. [123] linguistic-based cues formulated into ratios to differentiate between fraud and non-fraud firms. They find that the former use more: "*activation language, words, imagery, less lexical diversity*" [103]. These ratios form the feature set in a suite of classification algorithms. The model incorporating Naive Bayes and C4.5 achieved the highest classification accuracy.

Lee et al. [166] used only 4 LIWC (2001) variables to build a logistic regression model. However, performance metrics (Appendix A, Table A.3) reveal poor results. This can be attributed to the poor discriminative and predictive ability of the 4 features used.

Lee et al. [167] examine whether or not the content analysis approach is still an effective tool for detection of irregularities or fraud leading to financial statement restatements after the enacting the Sarbanes-Oxley act. They conduct t test and find that there is significant differences between fraud and non-fraud narratives using LIWC variables as features.

Purda and Skillcorn [168] first create a bag of words model (a term document matrix that contains words in the documents with their frequencies). They then use a decision tree-based approach called random forests to sort the words in rank order from most to least predictive. They use the top 200 predictive words to train an SVM model and then test on the 25% percent of the data not used in developing the model. Results are shown in Appendix A, Table A.3. The authors also compare the effectiveness of their method to alternative fraud detection approaches across different samples and find that it consistently performs well. They also find little evidence of effective fraud

prediction based on Loughran and Macdonald's financial word lists (negative, uncertain or litigious).

Throckmorton et al. [169] uniquely amalgamated features from categories used in previous research. They combined vocal, linguistic and financial cues. Financial cues were numeric features that related to firm performance (4 in total). The acoustic features were selected based on research by Zuckerman et al. [170] (7 in total). Finally, 7 linguistic features were extracted based on recommendation from research conducted by Vrij et al. [131]. These researchers find in accordance with findings from deception based studies that deceivers use less self-referential words. Therefore Throckmorton et al. [169] made a count of first person singular and plural nouns and impersonal pronouns and used them as features. Again in accordance with deception based research, Adams and Jarvis [171] find that deceivers are less likely to use positive emotion words and are more likely to use negative statements as markers of deception. Hence positive and negative emotion words were also measured and used as features. Proportion of tentative words and words that connote certainty were also included to capture linguistic features that denote lack of conviction.

Throckmorton et al. [169] attempted to find out if combining features across categories provided better fraud detection than was achieved by any of the feature categories alone. However, performance improvements were only observed if feature selection was used suggesting that it is important to discard non-informative features. Overall results are shown in Appendix A, Table A.3. To check that the imbalance in the data did not impact classifier training probability density functions of the AUC (Area under curve) were generated. This showed the result of training using 10-fold cross validation (see chapter 6 for definition of cross validation). The results from training matched test performance. The authors also found surprisingly that linguistic features had no predictive power. They admit themselves that this is contrary to previous research conducted they cite research by Larcker and Zakolyukina [160] as an example. This could be due to the mode of extraction. They explain that linguistic cues were extracted from 5 minute CEO Q&A excerpts with their median transcript containing 751 words. In contrast, Larcker and Zakolyukina [160] relied on CEO speeches in the entire conference call which resulted in a median transcript length of 2902 words. They recommend further research to elucidate upon the predictive power of linguistic features.

Wang and Wang [172] took 5 firms (3 fraud/2 non-fraud) and applied Hierarchical Clustering with successful separation of firms using words from these reports.

Zhou and Kapoor [173] propose a new method called Response Surface methodology (RSM) that extracts features based on Rezzae's [2] 3 C's model and uses data mining techniques for prediction purposes. This approach should pivot knowledge/feature extraction to suit the unique circumstances of the firm under question. The authors stress that fraudsters find ways to circumvent detection applications. Therefore, mechanisms need to be found that uses domain knowledge to strengthen data mining techniques. They further propose an active discovery module that evolves ahead of possible fraudsters. They call on future researchers to design this module based on garnering greater understanding on the nature of fraud.

2.8.2 FSF using non-linguistic features – A Literature Review

Some recent research work in FSF detection using non-linguistic features was also tabulated in Appendix A, Table A.2. Kanapickiene and Grundiene [174] found the following ratios differ significantly between fraud and non-fraud firms:-

- Profitability ratios **(**Return of sales)

In particular, the net profit to gross profit (NP/GP) ratio indicates fraud. It shows that sales, cost of sales or operating expenses, which are not typical of usual business are shown in financial statements.

- Profitability ratios **(**Return of Investment)

Gross profit to Total assets (GP/TA), the EBT to equity (EBT/Eq), the net profit to equity (ROE) ratios.

- Liquidity ratios

The inventories to current liabilities (INV/CL), the cash to total liabilities (CACH/TL), the cash to current liabilities (CACH/CL) ratios.

- Solvency ratios

All ratios of this group (except for the total liabilities to equity (TL/Eq) ratio) show statistically significant differences in fraud and non-fraud financial statements.

- Activity ratios

All ratios of this group (except for the inventories to sales (INV/SAL), the cost of sales to inventories (CS/INV) ratios, i.e., ratios defining inventory turns) show statistically

significant differences in fraud and non-fraud financial statements.

- Structure ratios (Total assets structure ratios)

All ratios of this group (except for the accounts receivable to total assets (REC/TA) ratio) show statistically significant differences between fraudulent and non-fraud financial statements.

- Structure ratios (Current assets structure ratios)

Two ratios of this group were investigated, the inventories to current assets (INV/CA) and the cash to current assets (CASH/CA) ratios. They show statistically significant differences between fraud and non-fraud financial statements. Interestingly, inventory structure is different both in total assets and in current assets of the company.

- Structure ratios (Property structure ratios)

Ratios defining the share of retained earnings in total assets or property are not a statistically different ratio. Meanwhile, the Current liabilities to Total liabilities (CL/TL) ratio differ significantly.

Alden et al. [175] use Genetic Algorithm (GA) and MARLEDA (an estimation of distribution) algorithm to train classifiers to detect patterns of FSF. The authors find that these two algorithms surpass traditional logistic regression models in the classification task. Results shown in Appendix A, Table A.2.

Chen et al. [367] build 3 models (SVM, C5 and Logistic Regression) using initially 29 variables that are reduced to 8 after feature selection. These include financial variables which relate to operating capabilities, profitability index, debt solvency ability index and financial structure. The non-financial variables include relevant variables of stock rights and scale of an enterprise's directors and supervisors. Their dataset comprise financial statements from 132 Taiwanese firms. The empirical result indicate that the SVM model performs the best in the type I error (specificity score) the C5.0 has the best performance in the type II error (sensitivity) and overall classification correct score (Appendix A, Table A.2).

Chen [95] used financial statements from 44 fraud firms and 132 non-fraud firms (Taiwan based). From these statements they extracted 30 variables (23 financial and 7 non-financial). Feature selection was performed using classification and regression trees (CART) and the Chi squared automatic interaction detector (CHAID). The former deigned: cash flow ratio, current assets ratio, sales growth rate, and natural logarithm of total liabilities as significant. The latter returned: debt ratio, cash flow ratio, quick

ratio, current assets ratio, return on assets before tax, interest and depreciation, operating expenses and operating expenses as important. The best results were returned by decision tree classifier using the (CART) selected variables (classification accuracy: 83%, Type 1 error 11%, Type 2 error 22 %). The next best result was obtained again using the CHAID selected features with a decision tree classifier (classification accuracy: 80%, Type 1 error 18%, Type 2 error 20%) .

Dechow et al. [113] document the most common types of misstatements and find that the overstatement of revenues, misstatement of expenses, and capitalizing costs are the most frequent types of misstatements. They investigate the characteristics of misstating firms on various dimensions, including accrual quality, financial performance, nonfinancial performance, off-balance-sheet activities, and market-related variables. They find that at the time of misstatements, accrual quality is low and both financial and nonfinancial measures of performance are deteriorating. They also find that financing activities and related off-balance-sheet activities are much more likely during misstatement periods. Results indicate that growth in cash sales is unusually high during misstatement years. They build scaled logistic regression models using both financial and non-financial data, results are shown in Appendix A, Table A.2.

Gill and Gupta [176] using 114 financial reports (29 fraud, 85 non-fraud) extracted financial ratios, reduced to 32 and tested for discriminatory power using 3 classifiers results shown in Appendix A, Table A.2.

Huang et al. [251] examine the spatial relationship of data that are classified into 2 categories (fraud and non-fraud). They use Growing Hierarchical Self-Organizing Map (GHSOM) to explore the topological relationship of the high-dimensional data that they have collected. These are unsupervised neural networks for clustering. Data is used (shown in Appendix A, Table A.2) to train a pair of GHSOMs and then the topological patterns are examined to derive a classification rule with respect to each subgroup for identifying the potential fraudulent samples. After confirming the existence of the spatial relationship, the proposed approach characterizes the underlying features of each subgroup. For each counterpart leaf node, all training samples and the statistical information from the samples of it is fraud type and non-fraud type leaf nodes are used to derive the non-fraud-central rule as well as the fraud-central rule, and the one with superior classification performance will be adopted. Using this approach the authors

find that fictitious revenues, capitalization of items that should be expensed and misappropriation of assets aid in correct classification of the statements.

Kim et al. [177] build 3 multi-class misstatement models using LR, SVM and Bayesian Networks (BayesNet) to detect and classify misstatements according to the presence of fraud intention. To deal with class imbalance they undertake cost sensitive learning using MetaCost (manipulation of the cost function associated with classifier, see chapter 6 for explanation). They used features (financial and non-financial) from previous research. They find that features such as short interest ratio and firm efficiency measures show discriminatory potential. Results shown are shown in Appendix A, Table A.2.

Li [178] took 55 financial reports of Chinese firms who had committed 2 types of FSF. He extracted 16 financial ratios per firm and applied K-means clustering algorithm to the data. The results (shown in Appendix A, Table A.2) indicate that clustering could be a promising approach to execute to aid in discriminating fraud from non-fraud firms.

Lin et al. (2015) build 3 fraud detection models using supervised learning on decision trees (CART), Logistic regression (LR) and Artificial Neural Networks (ANN). They build their models using features that relate to the fraud triangle. As described previously this theory posits that financial reporting fraud depends on three factors: incentives/pressures, opportunities, and attitudes/rationalization of financial statement fraud. The authors extract 32 features that relate to these categories. Specifically, 11 of 32 factors belonging to pressure/incentive dimension, the other 15 factors belonging to opportunity dimension and the last 6 of 32 factors are related to attitude/rationalization dimension.

This study also investigated the differences that came forth between the judgments from experts and empirical results of prediction model. In prediction model, two fraud factors are included in the top 10 of all prediction models. These fraud factors are: corporate credit risk Index and historical restate frequency which belong to ‘‘*The need for external financing*” category in the pressure/incentive dimension and *‘‘Historical restate frequency*” in the attitude/rationalization dimension respectively. This result is different from the judgments of the experts obtained in the study. The prediction model shows that three dimensions of the fraud triangle all play important roles but experts place importance only on pressure/incentive and opportunity dimension. According to Lin et al (2015) this gap warns that the auditors and users of financial statement should pay more attention to the attitude/rationalization dimension, especially when the firm

has a high frequency of financial restatements. The judgments of experts are most consistent with CART prediction model. Only two of the fraud factors (historical restate frequency, and CFO turnover frequency) of CART model are unmatched with experts' judgement.

Pai et al. [179] built an application as an aid to auditors to alert to possible anomalies that indicate fraud. Description and results are shown in Appendix A, Table A.2. Features found to aid in discriminating fraud from non-fraud records were: profitability features (net profit to total assets, earnings before interest and tax), leverage features (total debt to total assets), efficiency features (net income to fixed assets, inventory to sales) and corporate governance features (pledged shares of directors).

Perols [180] compares the performance of 6 machine learning models in detecting financial statement fraud. The results show, that logistic regression and support vector machines perform well relative to an artificial neural network, bagging, C4.5, and stacking. The results also reveal some diversity in predictors used across the classification algorithms. Out of 42 predictors examined only six are consistently selected and used by different classification algorithms: auditor turnover, total discretionary accruals, Big 4 auditor, accounts receivable, meeting or beating analyst forecasts, and employee productivity.

Ravisanker et al. [35] obtained 202 financial statements of firms that were listed in various Chinese stock exchanges, of which 101 were fraud and 101 were non-fraud. They extracted 35 financial data/ratios reduced to 18 after feature selection. These features related to the firm's ability to generate profit or profitability. The top features are associated with primary business income, and five are associated with either gross or net profit earned by the firm. According to the authors this indicates that: "*fraudulent firm usually tried to inflate the profit or the income figures in order to create an impressive financial statement*" [35].

Song et al. [181] (found) develop an ensemble classifier (decision trees, case-based reasoning, BPNN and SVM) to separate out fraud and non-fraud narratives. The features used are those risk factors identified on in SAS 99 (Statement on Auditing Standards). They relate to motivation, condition and attitude (similar variables that are included in the fraud triangle). The data used in this study were Chinese firms and in China according to the audit technology for detection of financial fraud risk (ATW No. 1) issued by the Chinese Public Accountant (CPA) committee, imperfect governance and internal environment must be seriously considered when assessing financial

statement fraud risk. So such factors are also included in the fraud model built. Results are shown in Appendix A, Table A.2.

Tarjo and Herawatib [182] build a Beneish M-Score models using logistic regression with numerical features (gross margin index, asset quality index, sales growth index, depreciation index, sales and general administration expenses index, total accrual, leverage index) and attained 77 % accuracy.

Whiting et al. [183] used financial statements from 228 firms (114 fraud and 114 non-fraud) from which they extracted 12 financial ratios and built machine learning models. The three best performing algorithms are shown in shown in Appendix A, Table A.2.

2.9 Discussion and Conclusion

From the views expounded on language by some scholars it can be deduced that it is possible to grasp language for computational purposes. The structural and cognitive school of thought agree that words represent concepts embedded in our minds. Advocates of Systemic-Functional Linguistics theory promote the study of language within the context of its use. This entails understanding the factors that impinge on language use within that context. It is possible to derive a set of rules that can represent the syntax - the construction of sentences used to convey meaning more fully. This is exemplified by the widespread use of parsers [39]. It was argued that use of a corpus in a domain of interest would provide empirical evidence on language use and the ground truth into the object of inquiry. Upon this corpus it was further argued that the use of statistical techniques to determine patterns would provide valuable insights. Chomsky's arguments against the use of these techniques were delineated but it was shown that his view into the study of language were from a different angle. The angle and approach taken here is that empirical methods have shown noteworthy success in the big data era and the use of a corpus provide a ground truth upon which deductions can be made and have proven to be correct.

The need for financial reporting is apparent and is the main armoury used to tackle agency and asymmetric information that can lead to impression management (bias) to outright deception. The view adopted in this thesis, in light of the argument is that the behavioural school of thought better reflects managerial motivations. The irrationality alluded to in this world view was long ago espoused by Keynes as "*animal*

spirits” which he defined as: "*a spontaneous urge to action rather than inaction, and not as the outcome of a weighted average of quantitative benefits multiplied by quantitative probabilities*" (Keynes, 1936). Such motivations have been widely endorsed by others [117]. Whereas, the concept of economic rationality has been discredited. It is not an adequate description of the behaviour of managers and investors in relation to the provision and dissemination of information in corporate narrative documents. The real world is characterised by: "*uncertainty and imperfect knowledge; ambiguous and heterogeneous expectations, abilities and preferences on the part of both management and all the groups which interact with the firm; competing and conflicting demands upon the firm; and dynamic and obscure relationships between strategies and outcomes*" [70] . It is further clear that given these “*animal spirits*” the information relayed to stakeholders may seek to confound the truth, to muddle the reader and possible hide misdemeanour such as fraud.

Clearly financial fraud and FSF is a massive problem that is escalating. As indicated the focus in this thesis is FSF. The predisposing factors that are ripe conditions for FSF were mapped out. Recent research has shown more concretely variables that relate to these predisposing factors as identified in the newer version of the fraud triangle. Generally, auditors have found it difficult to uncover FSF. Data mining techniques that have been used to detect FSF were shown. In sum such findings indicate that: insights garnered from fraud triangle based research and the data mining techniques shown do have potential in financial fraud detection and can be deployed by auditors who have difficulty in uncovering this misconduct unaided.

Deception studies provide further insight into how deception manifest in language use. It is clear from these studies that there are clues that aid in detection. Table 2.1 summarized many of these clues that have been found to be indicative of deception. Table 4.6 in chapter 4 shows how these cues could be operationalized to be extractable from a body of text. Of particular note was the Management Obfuscation Hypothesis which states that managers could have strong motivations to obscure the true poor performance of firms. This is often the case with fraud firms where underlying health of the firm is poor. The main way to detect reduced readability in text that currently hold sway are simplistic readability formulas. It was shown that these are inadequate on their own to properly gauge readability (they will be further discussed in chapter 4). New measures using the Coh-Metrix tool were put forth. The indices that are integral to this tool that probe the text further were delineated. These indices will

be used in the corpus under study to determine readability. In sum deception research clearly shows that where deception cues outlined have been used in deception based studies, they have been able to show up those that were engaged in lies and deceit.

Finally, research conducted into FSF using data mining techniques also clearly shows potential to aid auditors and law enforcement agents to detect this misconduct. Features used are often financial ratios and other details relating to firm and management structure. Such non-financial factors are crucial indicators to aid in fraud detection and should be included in fraud models. This research as mapped out in Appendix A (Table A.2 and A.3) primarily using the classification technique shows promising results, even in cases where there is an unbalanced data set. This indicates that the features that mark out a fraud firm could be potentially strong. However, it is easily observable from Table C.1 and table C.2 that FSF using linguistic data is less prodigious than research that uses ratios as features upon which fraud models are built. Given the increasing importance attached to narratives in financial reporting [7] this needs to be addressed. Further text is a better vehicle to hide deceit than numbers as language has an innate impreciseness therefore better suited for those given to deception. Research that used linguistic features showed similar good performance again the technique used was primarily classification. However as indicated by Zhou and Kapoor [173] further insights need to be constantly gathered on fraud tactics to update the techniques used for effective detection.

Additionally as pointed out by Throckmorton [169] models that use ratios, linguistic cues and non-verbal vocal cues have each demonstrated their potential for detecting financial fraud. He adds that cues from these 3 categories should be amalgamated and used as features for fraud models in a combined manner and could improve fraud detection outcomes.

As shown by the research covered in this chapter truth and deception are separable. Those engaged in the latter leave traces of this intent. How this intent is manifested was shown and how it can be extracted to determine features for fraud detection models was also demonstrated. Therefore, there is confident hope that such models can be adapted and enriched with further insights for better FSF detection. This will be the subject of the next chapters.

Chapter Three

FRAMEWORK DESCRIPTION AND CORPUS ANALYSIS

"Measure what is measurable, and make measurable what is not so"

Galileo c 1564

3.1 Framework Description

In light of the literature review, a coherent unifying approach that would alert to anomalies such as FSF in firms financial reporting from the narratives may prove to be beneficial. A new framework is proposed based on data mining that as the results from previous literature indicate can be a robust and thorough approach for uncovering this criminality.

A high level view of the framework is depicted in Figure 3.1. Each module will be fully explained using data as it moves through the framework. Any further processes within these modules will be elaborated. This framework requires a corpus to drive the downstream modules which automate the process of linguistic analysis and feature extraction. This in turn drives the machine learning based classifiers.

El-Haj et al. [184] present a similar general framework. They harvest reports by manual/data collection, then clean and parse text, removing tags, images, exhibits. They then analyse text though the use of word lists and text mining and machine learning to identify patterns. However, the framework below has been customised for fraud detection. Specifically the document representation stage and the number of ways presented for feature extraction are unique to the framework below. The approach outlined by Goel et al. [37], Humpherys et al. [103] Glancy and Yadav [165], Cecchini et al. [165] described in Chapter 2 follow a similar flow. Again the distinctive difference is the wide range of features chosen which determine success at the classification stage.

Before proceeding, it is instructive to re-state the research question addressed. Perpetuators of deception leave traces of their culpability in the narratives that they deploy. This has been verified through background and literature review as outlined

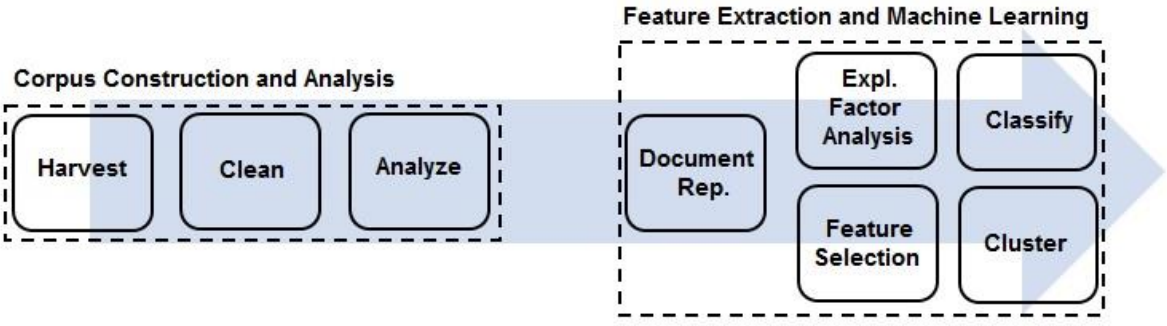


Figure 3.1: Proposed high level framework to decipher financial reports for deception detection.

in chapter two. In financial reporting, company narratives are used by the stakeholder community to gauge company operations and prospects. Given rising level of financial statement fraud, a renewed attempt is made to determine any anomalies that can alert to potential misconduct detectable from the narrative.

Hypothesis: Based on narratives of annual reports/10K alone, is it possible using the framework outlined to separate out a potential fraudulent firm from a non-fraudulent firm. This is based on evidence that suggests that the language deployed by truth-tellers and liars are distinct and can be distilled using natural language processing techniques.

As indicated the framework depicted above is used to tackle the research question. Pivotal to the framework is the corpus of 408 annual reports/10-K. This corpus will be the subject of this chapter. How the corpus was gathered and its composition and salient features will be mapped out.

3.2 Data Collection and Preliminary Analysis

At a microeconomic level, firms both large and small are the juggernauts of growth and raise national productivity at a macroeconomic level. Their operations should rightly be subjected to scrutiny. Transparency and openness on their operations are vital to ensure that the stewards act with integrity in managing these enterprises. Examples abound where corruption and nepotism, have creeped in and led to massive losses (Satyam, Enron, Royal Bank of Scotland). In a bid to boost transparency and detect fraud and to seek out alternative ways to the status quo, financial narratives are examined.

The question that arises is what financial narrative would be a suitable text to examine for uncovering financial statement fraud? The annual report (AR) and 10-K were chosen. The AR is a statutory requirement. For example in the UK, companies incorporated under the Companies Act must produce one. Likewise in the US under federal securities law, implemented by the Securities Exchange Commission (SEC) public companies have to submit AR on form 10-K. It is a comprehensive report on company's activities. The 10-K is similar to the AR but has a distinct structure which is used to relay details of company operations and performance. Despite questions over its '*informativeness*' [185, 186] its status as the most important statement of firm progress and intentions by its stewards remain steadfast. The UK government recently updated Companies Act (2006) to enforce measures that promote greater clarity in annual report narratives that deal with strategy, risk and uncertainty. Similarly in the US, Sarbanes–Oxley Act of 2002 sought to improve corporate disclosure for the investing public. These acts are acknowledgement of the significance of annual reports/10K to the stakeholder community. Furthermore, predominantly the AR is the narrative of choice for researchers investigating the effects of financial disclosure on variables as shown in Table A.1, A.2 and A.3 in Appendix A. Both 10-K and annual report were used as in some cases the fraud companies produced one and not the other. It was also noted that where companies had produced both an Annual Report and 10K they were cross referenced.

Annual report of companies based in the UK are published as PDF documents and follow a loose structure. This comprises sections such as chairman's statement, operating review, corporate governance, corporate social responsibility [184]. In the US annual reports on Form 10-K follow a defined structure comprising 4 parts and 15 schedules.

Harvesting these reports involved the following steps:-

1. Identification of fifty one companies formally indicted for Financial Statement Fraud (FSF). This is achieved, using archival records from the Serious Fraud Office (SFO) database and by examining Accounting and Auditing Enforcement Releases (AAERs) issued by the Securities Exchange Commission (SEC). These are financial reporting standard violations and provides information regarding enforcement actions (Humpherys 2011). AAER are checked to make sure the term 10-K was included in description and only those firms are selected that had violated Rule 10 of the Securities Exchange Act, 1934. This rule requires the intent to deceive, manipulate or defraud.

The original 10-Ks submitted were collected and not the restated 10-Ks. The original 10-Ks were selected because: "*a restatement of a financial statement is created to correct the previous financial statement for intentional/unintentional errors and accounting irregularities. Restatements represent an acknowledgment by the firm that prior financial statements were not in accordance with Generally Accepted Accounting Principles*" [37]. In the UK a regulatory body the Serious Fraud Office (SFO) has records of firms indicted for FSF. For UK based firms, press releases and SFO records are scrutinised to ensure that the crime committed is FSF enacted through the AR.

2. For each fraud company, the AR/10K for the fraud year and fraud year -1 is collected. This approach is aligned with the literature findings, which indicate that fraudulent accounting practises are typically undertaken 3.02 years before fraud is exposed [187]. This results in a 102 AR/10K from fraudulent companies.

3. For each fraud company a similar sized non-fraud company from the same industry and a known competitor is chosen. These prerequisites: "*minimize potential confounds because of differing economic conditions between the fraudulent and non-fraudulent companies or to eliminate differences across dissimilar industries*" [103]. AR/10-K's for the same years as the fraudulent firm is extracted. Again only the originals and not the restatements are chosen. Each firm is checked to make sure there were no AAERs to indicate non-conformance to GAAP regulations.

4. All numerical data and tables are removed to allow a focus in on the language used in these reports.

5. The reports are extracted from SEC EDGAR database, Companies House UK and Thomson ONE database. For the 10-Ks, sections on Item 1 Business, Item 1A Risk factors, Item 7 Management's Discussion and Analysis of Financial Condition and Results of Operations and Item 7A Quantitative and Qualitative Disclosures about Market Risks are extracted. For the AR sections on Chairman's statement, strategy, risk and uncertainty are extracted. Extracts from both reports are narratives that deal with similar areas of company operations.

The time period for AR/10-K span from 1989 to 2012, with a concentration of fraud cases from 2000-2002. In the US this precipitated the enactment of the Sarbanes–Oxley (SOX) legislation in 2002. Two scenarios for fraud detection will be set up. In the first case, a matched-pair data set, one fraud firm is matched to one non-fraud firm. Hence, given the above steps two fraud reports to two non-fraud reports. In the other peer set scenario, for each fraud firm was matched to 3 non-fraud firms, resulting

in a ratio of two fraud reports to six non-fraud reports. This data set up allows for a more realistic portrayal of the composition of fraud firms to non-fraud firms.

Cleaning the corpus involved the use of OCR software in cases where the AR was an image. The aim was to detect the text in the image. All formatting marks had to be removed and this was performed using in some cases Visual Basic Macros in Word, but often had to be undertaken manually. All images, figures, tags and exhibits were removed. Coh-Metrix is the tool (discussed in Chapter 4) that is used to extract readability measures from text. A stipulated prerequisite for its use was that the text had to look as if: "*the writer had just finished typing it, had it checked for typos and errors by a large group of copy editors, printed if off, and handed it over to the reader*" [50]. This was the standard that was attained in the corpus. All documents were then saved in .txt files to allow reading of file by text mining software.

Previous fraud detection models covered in Chapter 2 follow a similar approach. Goel et al. [37] were the only other authors that set up a peer set scenario. Goel et al. [37], Humpherys et al. [103], Glancy and Yadav [165], Cecchini et al. [161] all undertook a similar approach to data and sample selection (Table A.3, Appendix A maps out their data and findings). Differently in this study the firms selected are both US and UK based, both a matched-pair and a peer data set is constructed, more diverse and varied features are selected and results of classifiers compared to obtain optimal feature/classifier combination. This will be elaborated upon in Chapter 4 and 5.

3.3 The Corpus

The methodology of corpus linguistics will be executed over the collection reports that have been collated (the corpus). As indicated in chapter 2 introspection as the main source of data in linguistics has to some extent been rejected. Since the 1980's and with the rise of machine learning techniques the use of corpora have come into wide scale use in linguistics [6, 188]. As corpus is empirical data, it leans the study of language use towards a more objective enquiry.

As indicated in chapter 2, from the scientific perspective a corpus driven approach is a powerful methodology as it is a systematic approach to the analysis of language using such data as frequency counts. This renders it open to objective verification of results. This is not possible using introspection as proposed by Chomsky [5].

The rationale for using a corpus for this study was covered in chapter 1 and 2. The approach undertaken here is both corpus-based and corpus driven. The former as the corpus is used to test hypotheses and the latter as frequency lists are also used to drive the focus of the analysis [189]. The corpus enables a distillation of the salient linguistic features in the text, which is then used to test the hypotheses of a pattern that indicates a difference in use. According to Sinclair (1991) the raison d'etre of corpus based language study is to identify differences: "*the distinguishing features of one type of text only come to the forefront when contrasted to another type of text*" [190].

A central feature of deception detection research is to be able to recognise a lie. In the past, researchers [132] set up controlled experiments to aid in distinguishing a liar from a truth-teller. However, such studies are hampered by poor reproducibility of results, subjects have no personal loss or gain at stake, the motivation to lie is weak. Fitzpatrick and Bachenko [38] propose the: "*construction of standardized corpora that would provide a base for expanding deception studies, comparing different approaches and testing new methods*". They recommend using publicly available data as it is likely to be a rich source of ground truth evidence. A perfect example of this is the Enron e-mail corpus. This has been extensively interrogated and linguistic features put through algorithms to pick up patterns that could be indicative of fraud and workplace behavioural cues. This kind of empirical data would be very hard to attain in a laboratory setting. Therefore, the construction of this corpus is to set a standard by gathering authentic data (only text of companies known to have committed financial statement fraud, juxtaposed with similar non-fraud companies) as alternative more robust base upon which to build fraud models. This text was gathered from the most reliable of sources such as SEC-US and Companies House- UK.

McNamara et al. [50] outline the prerequisites for building a corpus:-

- Language must be a particular genre and be thematically related.
- Representative and balanced.

A corpus is said to be balanced: "*if the relative sizes of each of its subsections have been chosen with the aim of adequately representing the range of language that exists in the population of texts being sampled*" [6]. A representative corpus is one: "*sampled in such a way that it contains all the types of text, in the correct proportions, that are*

needed to make the contents of the corpus an accurate reflection of the variety of language that it samples" [6].

The AR/10K is financial text of a particular genre and fulfils the representative and balanced criteria. However, as McNamara et al. [50] point out it does not need to be a: "*perfect corpus we just need one that gets the ball rolling*". This perfect corpus would be time consuming and expensive to collect. The practical aspects of corpus compilation is under appreciated [188]. The results from corpus based studies should be: "*practical and suggestive rather than exhaustive and definitive*" [50]. McNamara et al. [50] also point out that a corpus has to be large enough to reflect the source. In this study the corpus contains representative documents from fraud firms that have been indicted for major financial statement fraud. It contains similarly representative documents from non-fraud firms.

AR/10K reports used from both the fraud and non-fraud firms to build up corpus are shown in Table C.1, Appendix C. 408 documents, 6356201 words make up the corpus. A hundred and two documents are from known fraud firms, most of them involved in high profile cases of FSF committed on a grand scale eg Enron, Worldcom, Tyco. Each are matched up with three similar firms eg Enron → Williams, Centrepoint Energy, American Electric Power; Worlcom → AT&T, Sprint Corporation, Verizon Corporation and so on This improves the representativeness of the corpus. Table 3.1 provides high level detail on the linguistic composition of the corpus.

The corpus follows the natural law observed in all languages and in all corpora: "*systematic frequency distribution such that there are few very high frequency words that account for most of the tokens in text (e.g. "a", "the", "I", etc.), and many low frequency words*" [191]. This simple pattern is often referred to as "*few giants and many dwarves*" [192]. This relationship obeys a power law known as Zipf's law. The rth most frequent word has a frequency f(r) that scales according to formula shown in Eq. 2.1, r is called the "*frequency rank*" of a word, and f(r) is its frequency in a corpus, with α ≈ 1 [193].

$$f(r)\alpha\frac{1}{r^{\alpha}} \qquad \text{(Eq. 2.1)}$$

Corpus Statistics	
Total Number of Reports	408
Number of Reports Fraudulent Firm	102
Word Tokens in Fraud Reports	1681800
Word Types in Fraud Reports	18164
Number of Reports from Non-fraudulent Firms	306
Total Word Tokens in all Non-fraud Reports	4674401
Total Word Types in all Non-fraud Reports	30127
Total Corpus Size	6356201

Table 3.1: Lexical statistics on corpus.

Figure 3.1 shows the distribution of tokens in the corpus and as can be seen it conforms to the above law.

The corpus will be used to investigate patterns associated with linguistic features and to gain insight into how these patterns differ within varieties [188]. Corpus linguistic methods are not well delineated [6]. However, a number of methods, are fundamental in this discipline to the study of language, listed below [188].

- *Use of frequency list*

These lists: "*record the number of times that each word occurs in the text. It can therefore provide interesting information about the words that appear (and do not appear) in a text*" [194]. The frequency information gives an indication of the vocabulary composition of the text. Sinclair (1991) noted that "*anyone studying a text is likely to need to know how often each different word form occurs in it*".

- *Keyword Analysis*

This is one of: "*the most widely-used methods for discovering significant words, and is achieved by comparing the frequencies of words in a corpus with frequencies of those words in a (usually larger) reference corpus*" [194].

- *Concordance*

Also referred to as keywords in context (KIWC) focuses on the context of keywords at the sentence level [6].

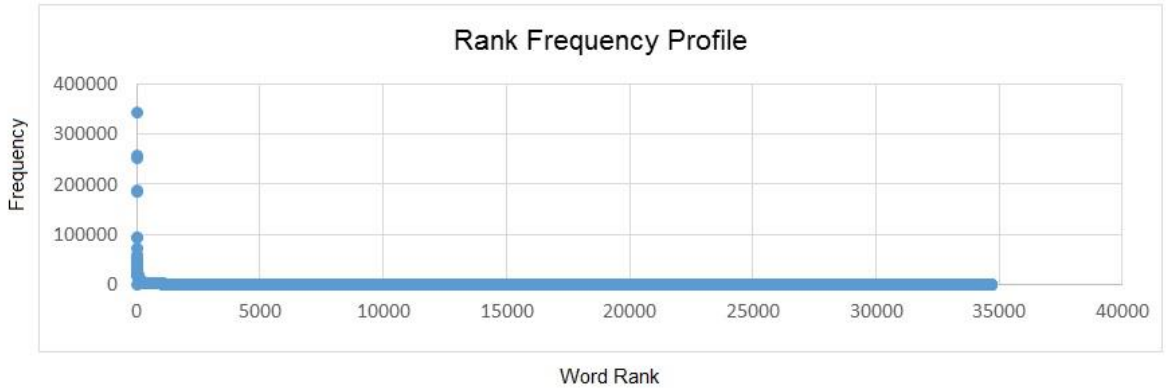


Figure 3.2: Zipf law in action over the corpus, a plot of word rank versus frequency.

- *Collocations*

The study of phrases where the meaning of words is found through several words in a sequence [6].

AntConc is a freeware corpus analysis toolkit for text analysis [195] that is used to implement the above methods.

As McEnery and Hardie [6] put it: "*corpora (plural of corpus) are an unparalleled source of quantitative information*". This is normally frequency information on directly observed linguistic features. However it is not enough just to rely on descriptive statistics to build a picture of the text composition, observed differences in language use need to be tested for significance. This is done using a significance test to ascertain how likely it is that a particular result is a coincidence. Further in this chapter, the results from the above analysis will be presented and an attempt made to explain and interpret patterns.

There has been contention in the literature as to the best significance test to use with reference to language. There is an extreme position put forth by Kilgarriff [196] that: "*Language is never, ever, ever, random*" because we speak and write with purpose. Therefore, as statistical hypothesis testing uses a null hypothesis which posits randomness therefore when examining linguistic phenomena in corpora, the null hypothesis will never be true. Further Lijffijt et al. [197] argue that: "*the use of the χ2 (chi-squared test) and log-likelihood ratio tests is problematic in this context, as they are based on the assumption that all samples are statistically independent of each other*". Words within a text are not independent and therefore inferences drawn from such tests should be treated with caution. Chi squared test further presupposes a

normal distribution of the data and as shown in the rank frequency profile corpora do not follow this pattern but produce positively skewed distributions. The log likelihood test is commonly used in corpus linguistics as it makes no assumption of a normal distribution [198].

Kilgarriff [196] concedes that although: "*randomness assumptions are always untrue, but that does not preclude them from frequently being useful*". Further, Lijffijt et al.[197] argues that it is possible to employ other tests where we assume independence at the level of text rather than individual words. This allows us to account for the distribution of words within a corpus. He endorses the use of significance testing to find consequential differences between corpora, but that assuming independence between all words may lead to overestimating the significance of the observed differences, especially for poorly dispersed words. He recommends the use of the t-test, Wilcoxon rank-sum test, or bootstrap test for comparing word frequencies across corpora.

McEnery et al. [6] argue that empiricism lies at the core of corpus linguistics and that there is no other way to test for how frequent a word is in a corpus and engage in comparative analysis other than to employ the standard statistical tools. The authors recommend that the inferential statistics that allow us to test for significance should be treated with caution.

3.4 Frequency Inspection

The most basic statistical measure is a raw frequency count. This is simply counting the number of occurrences of a word (token) in a corpus, known as the raw frequency. For the fraud reports this is shown in column two, Table D.1, Appendix D (top 60 word types). The corresponding frequency for non-fraud reports are shown in Table D.2, Appendix D. The next columns in both Appendices, show the relative frequency. This is the raw frequency divided by the number of tokens in the fraud or the non-fraud reports. This puts in context the magnitude or otherwise of the token with respect to the totality of similar cases. Additionally, according to Baron et al. [194] and Rayson [188] a full appreciation of the frequency of a token in the text is only possible through a normalised frequency which answers the question: "*how often might we assume we will see the word per x words of running text?*" [6]. In this case *x* is 1000 words, a typical base of normalisation for density scoring. So for example the word *million*

occurs 9671 times in the fraud reports. The relative frequency is attained by dividing it with the number of tokens in fraud reports being 1681800, resulting in 0.005562. The normalised frequency is obtained by multiplying the relative frequency by 1000 to get a density score of 5.562. The normalised frequency is used here for comparing the fraud and non-fraud reports as the sample sizes are unequal – 102 fraud reports (1681800 tokens) compared with 306 non-fraud reports (4674401 tokens). This allows for easier comparison.

On first inspection there seems to be a homogeneity in the words used and in some cases a similarity in frequency of word usage. As Rutherford [199] argues this stability supports: "*the contention that narratives constitute an identifiable genre and implies that where differences do arise, significance can be attached to them*". This stability is further reinforced visually by the graph in Figure 3.3 which shows that the top 300 word types including the 60 listed in Appendix D (Table D.1) and (Table D.2) confirm homogeneity in usage across fraud/non-fraud reports.

To further check for differences in the mean frequencies of word types between the fraud and non-fraud reports, significance testing was performed. A preliminary test for the equality of variances indicates that the variances of the two groups (fraud with non-fraud) were significantly different in the region of F test =1.13 p=9.95741e-19. An F-test is designed to test if two population variances are equal. If the p-less than 0.05 suggests that the observed data are inconsistent with the assumption that the null hypothesis is true, and thus that hypothesis must be rejected and the alternative hypothesis is accepted as true [200]. Therefore a two-sample t-test was performed that does not assume equal variances.

The hypotheses are as follows:-

(Null) Ho: m1 = m2	(means of the fraud and non-fraud reports are equal)
(Alternative) Ha: m1≠m2	(means are not equal)

The mean of the normalised frequencies for all 18164 types in the fraud reports were compared with mean of all normalised frequencies for word types all the non-fraud cases. The observed difference is significant (p value < 0.05) and the t stat value is greater than the t critical 2 tailed value. Therefore the null hypothesis can be rejected and the alternative accepted that there is signifcant difference between word type usage in fraud and non-fraud reports.

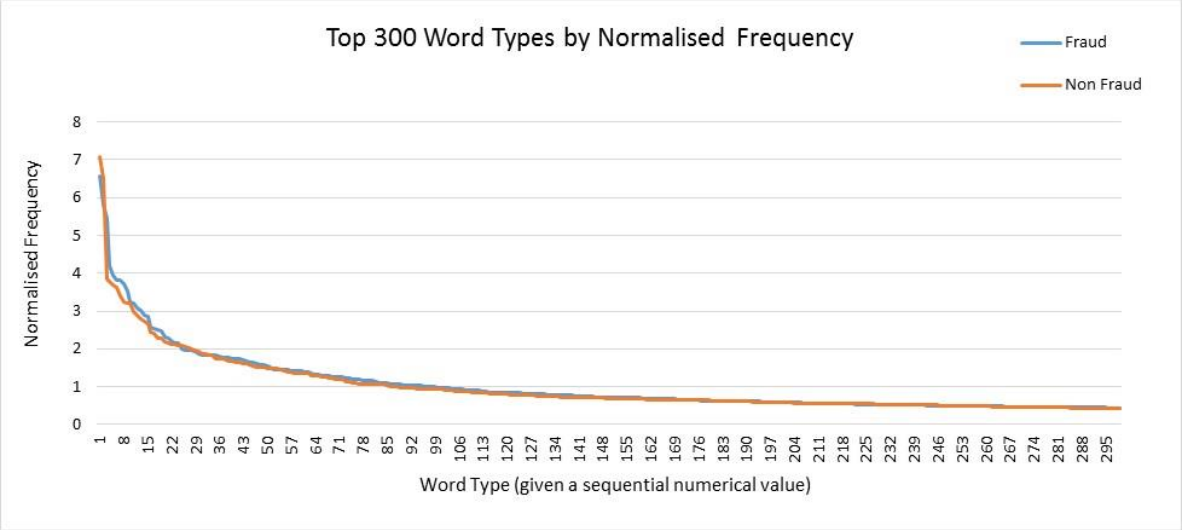


Figure 3.3: Top 300 word types in fraud and non-fraud reports.

t-Test: Two-Sample Assuming Unequal Variances		
	Fraud	*Non-Fraud*
Mean	0.036998409	0.02189475
Variance	0.031490733	0.017958853
Observations	18164	30095
Hypothesized Mean	0	
df	30575	
t Stat	9.888961954	
P(T<=t) one-tail	2.51672E-23	
t Critical one-tail	1.644903466	
P(T<=t) two-tail	5.03344E-23	
t Critical two-tail	1.960041576	

Table 3.2: Significance testing over word types (mean) in fraud and non-fraud reports.

All tokens in the fraud and reports were also lemmatised using AntConc functionality. Lemmatization is a process: "*of assigning a lemma to each word form in a corpus using an automatic tool called a lemmatizer. Lemmatization bring the benefit of searching for a base form of a word and getting all the derived forms in the result*" [375] an example of this would be: "*searching for go will also find goes, went, gone, going"* [375]. Advantages for performing lemmatization and the related less sophisticated process of stemming according to Manning [49] is domain dependent. He argues that in information retrieval for English for example it does not in aggregate improve performance, as Manning puts it: "*it helps a lot for some queries, it equally hurts performance for a lot for others*" [49].

In this domain there are some hazards to note that can go undetected for example variations of the lemma '*interest*' is deemed a canonical representative for a set of

related (inflected) word forms which include '*interested*', '*interesting*', '*interests*'. As can be noted *'interesting*' and '*interest*' as in '*interest rate*' can have very divergent meanings in this domain. When the frequencies of all these variations that use the root *'interest*' are added together, a certain inaccuracy is propagated. However like the bag of words model discussed in chapter 4 this simplification can sometimes be benefical from a classfication perspective as it aids in feature space reduction whilst often capturing information content.

Table D.3, Appendix D shows the top 60 lemmas in the fraud reports with frequencies. The corresponding frequencies for lemmas in the non-fraud reports are also listed. Figure 3.4 shows graphically the nature and strength of this relationship.

Again a preliminary test for the equality of variances was performed which showed that the mean of the two groups were significantly different. Again a two-sample t-test was performed that does not assume equal variances. Significance testing was performed over the lemmas (normalised frequencies) found in fraud reports with lemmas found in non-fraud reports. As can be seen from Table 3.3 there is a marked difference in the use of lemmas between the two reports, with a p value of < 0.05 and a t stat value that is greater than the t critical two tail value.

The above analysis using mean, standard deviation and significance testing is typically applied when using contrastive corpora. For example, Anderson and Corbett [201] reviewed a corpus based on registers of prose relating to general fiction, prose and conversation examining the mean and standard deviation of various linguistic features. They also analysed the British National Corpus using similar descriptive statistics analysis. They were able to gauge the representativeness of their corpus (a corpus of registers with the British National corpus) by comparing mean and standard deviation scores.

3.5 Keyword Analysis

Some words and phrases are considered key as they indicate key themes present in narratives [202]. They are markers of the '*aboutness*' and the style of a text [52]. Keywords are those expressions: "*that have a significantly higher or lower frequency of occurrence in a text or set of texts than we would expect given the frequency of occurrence of those expressions in a larger corpus used as a point of reference*" [201].

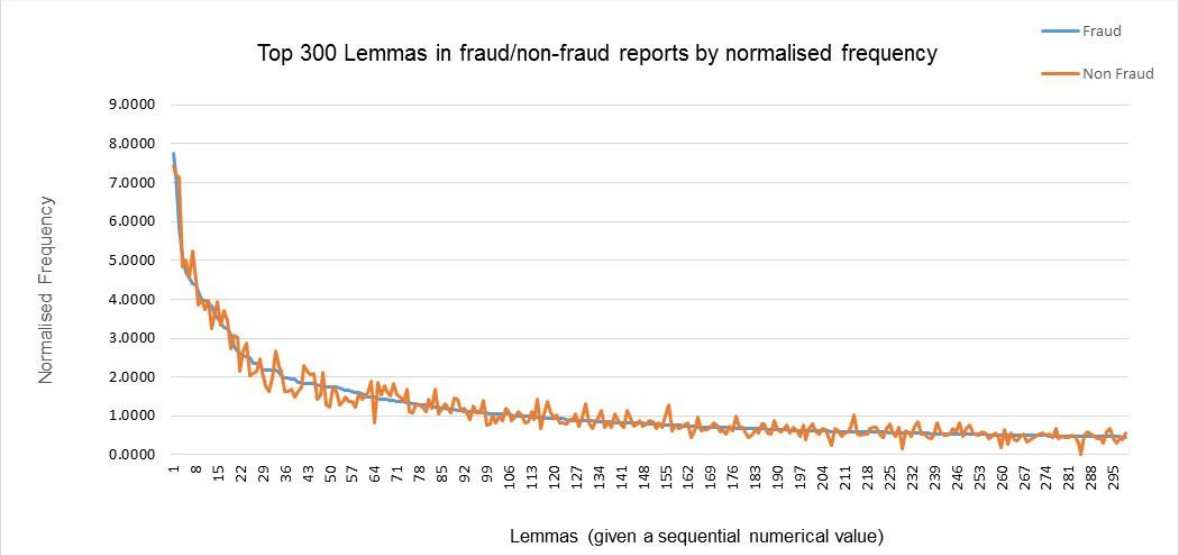


Figure 3.4: Top 300 Lemmas in fraud and non-fraud reports.

t-Test: Two-Sample Assuming Unequal Variances		
	Fraud	*Non-Fraud*
Mean	0.071093417	0.040914856
Variance	0.784882659	0.438103022
Observations	14066	24441
Hypothesized Mean	0	
df	23176	
t Stat	3.514725315	
P(T<=t) one-tail	0.000220524	
t Critical one-tail	1.644919377	
P(T<=t) two-tail	0.000441049	
t Critical two-tail	1.960066349	

Table 3.3: Significance testing over lemmas (mean) in fraud and non-fraud reports.

In this study, keywords are those whose frequency in the fraud reports is statisitcally significant when compared with the non-fraud reports (the larger corpus used as a point of reference). This notion of keyness applies to word types, lemmas and word sequences (collocations). In particular, in corpus linguistic research the notion of collocation the tendency of certain words to appear together has grown in signficance, it contributes to an identification of the '***aboutnes***' of a text [203]. Collocation analysis will be undertaken in the next section.

To guage keyness the raw frequency scores for both the (102) fraud reports and the (306) non-fraud reports are put through a log likelihood statistical test. This is the preferred option as compared to chi squared as mentioned earlier makes no assumption as to the underlying distribution of the data [6, 198].

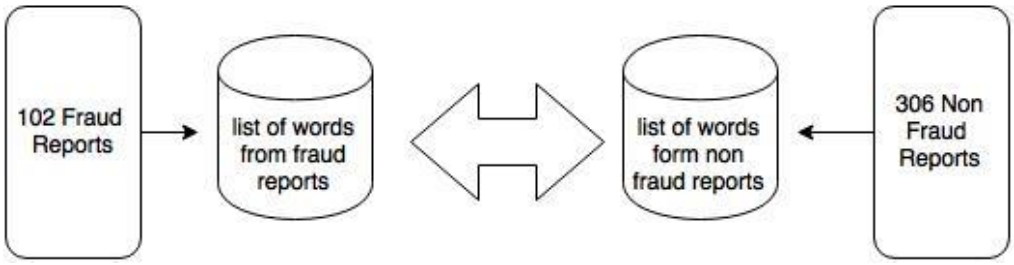


Figure 3.5: Reports set-up in AntConc to perform keyword analysis.

The log likelihood score was attained in three distinct ways.

1. The 102 fraud reports were loaded into AntConc. The 306 non-fraud reports were also loaded and set up as the reference corpus. The keyword generation method was set to log likelihood. This produced a list of keywords sorted by keyness scores. Figure 3.5 shows the approach taken by AntConc to determine keyness for words in the fraud reports. Once the reports are loaded, a word list is generated for the fraud reports and similarly another word list is generated for the non-fraud reports.

2. A study using corpus analysis methods on annual report narratives conducted by Rutherford [199] uncovered words that were deemed indicative of company health. These words were also put through a log likelihood calculation to determine if there is a difference in usage of these words between fraud and non-fraud reports. For this task log-likelihood and effect size calculator as described by Rayson [198, 204] was used.

3. Loughran and Macdonald [126, 205] developed word lists customised for the financial domain. They showed that word lists developed for other disciplines misclassify words in financial text. The word lists that they developed included negative, positive and uncertainty bearing words. As indicated by Pollach [206] such keywords can point to differences in: "*themes and attentional foci*" between the two sets of reports. These word lists were loaded into AntConc and differences in use in fraud and non-fraud reports noted through log likelihood testing.

Keyness is checked in the above three ways to ensure a robust and comprehensive check on keywords was conducted.

The log likelihood, as described by Rayson [198, 204] and extracted from http://ucrel.lancs.ac.uk/llwizard.html is calculated as shown in Table 3.4.

	Corpus 1	Corpus 2	Total
Frequency of word	a	b	a+b
Frequency of other words	c-a	d-b	c+d-a-b
Total	c	d	c+d

a and b = observed values
c = number of words in corpus 1
d = number of words in corpus 2

N Values

Table 3.4: Log likelihood calculation.

The expected values (E) are calculated according to the formula:

$$E_{i=\frac{N_i \sum_i O_i}{\sum_i N_i}}$$ N1=c, N2=d (Eq. 2.2)

Therefore for the above the calculation would be *E1=c*(a+b) / (c+d) and E2= d*(a+b)/c+d)*. As illustrated the expected values takes account of the size of the two corpora (in this case the fraud reports and non-fraud reports). Only raw frequencies are entered, as the formula normalises these values. The log likelihood is calculated according to this formula:-

$$-2\ In\ \lambda = 2 \sum_i O_i\ \ln(\frac{O_i}{E_i})$$ (Eq. 2.3)

Million	**Raw Freq Fraud Reports**	**Raw Freq Non-Fraud Reports**
$E_{i=\frac{N_i \sum_i O_i}{\sum_i N_i}}$	1681800(9752+33062)/ 1681800+4674401 =11328	4674401(9752+33062)/ 1681800+4674401 = 31485
$-2\ In\ \lambda = 2 \sum_i O_i\ \ln(\frac{O_i}{E_i})$	=2*(9752*(LOG(9752/F7)))+(33062*(LOG(33062/I7))) -567.69	

Table 3.5: Log Likelihood Calculation for the word '*million*'.

This equates to calculating log likelihood G2 as follows: G2 = 2*((a*ln(a/E1)) + (b*ln(b/E2))).To use the above logic for the word "*million*" in the corpus, the following calculations as depicted in Table 3.5 would take place. The log-likelihood is a *statistical significance* measure – it tells us how much evidence there is for a difference between two corpora. The higher the log likelihood value, the more significant is the difference between two frequency scores. Based on the above calculation, a score of 3.8 or higher is significant at the level of $p < 0.05$. A negative value indicates underuse in the fraud corpus in relation to the non-fraud reports.

The results for the top 60 words are shown in Figure 3.6 and 3.7. Thi s is keyness calculated using the first method. Some words had high keyness scores but poor dispersion in the fraud corpus for example names of companies like Enron, Adelphia. These words known are known as "*bursty*" words. They occur only in a handful of reports and do not contribute to the analysis. The aim is to find frequent words with high dispersion values as they are considered to have high currency in the language. Therefore all keywords with high keyness scores were checked for their dispersion by calculating their adjusted frequency. This involved taking the relative frequency and multiplying it by the number of documents in corpus where this word occurs divided by the total number of documents [207]. So for example with the word million in the fraud reports has a relative frequency of 9752/1681800, it occurs in all 102/102 of the reports, which gives an adjusted frequency score of 0.0057. Whereas a word like Enron with a relative frequency 629/1681800 is multiplied by the percentage of corpus parts in which Enron is found, in this case being 3/102 giving an adjusted frequency of 0.000011.

Using the first method (loading the fraud and non-fraud reports into AntConc) the top 160 keywords (by highest log linear significance scores) derived are shown in Table D.4, Appendix D and a few choice top keywords are plotted and graphed in Figure 3.6. These keywords are more pronounced in the fraud documents as compared to the non-fraud. Conversely, Table D.5, Appendix D shows words that are much more pronounced in the non-fraud documents and Figure 3.7 shows graphically these results.

Rutherford [199] analyzed word frequencies from narrative section of annual reports of companies of differing health. He looked at loss making/least profitable/most profitable/lowest geared and highest geared companies. He also examined companies by size (large and small). Rutherford [199] uncovered a number of words

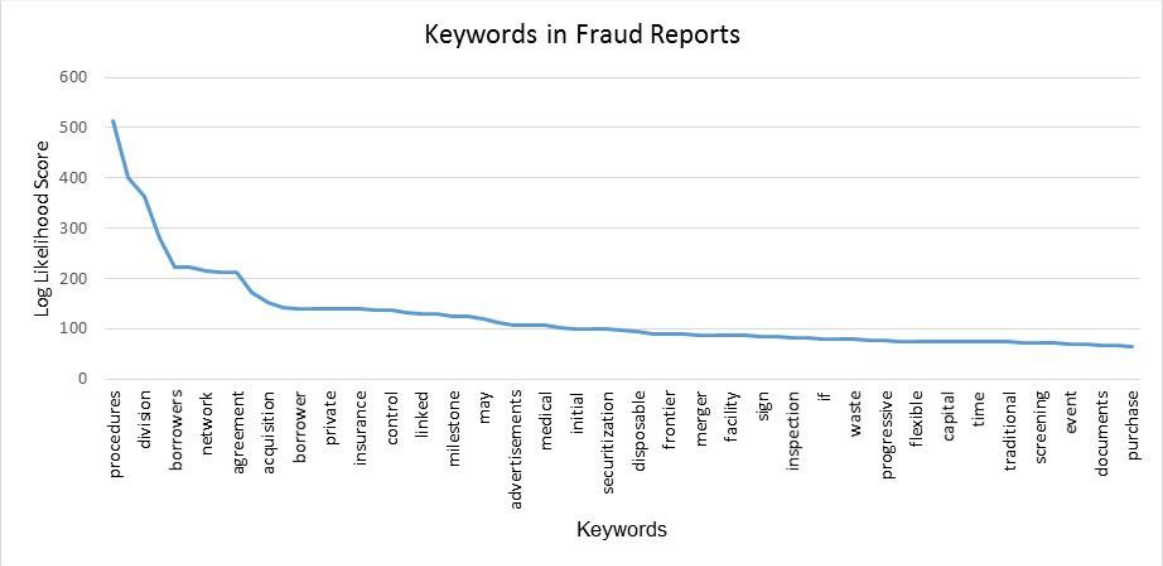


Figure 3.6: Keywords in fraud reports as identified using log likelihood score in AntConc.

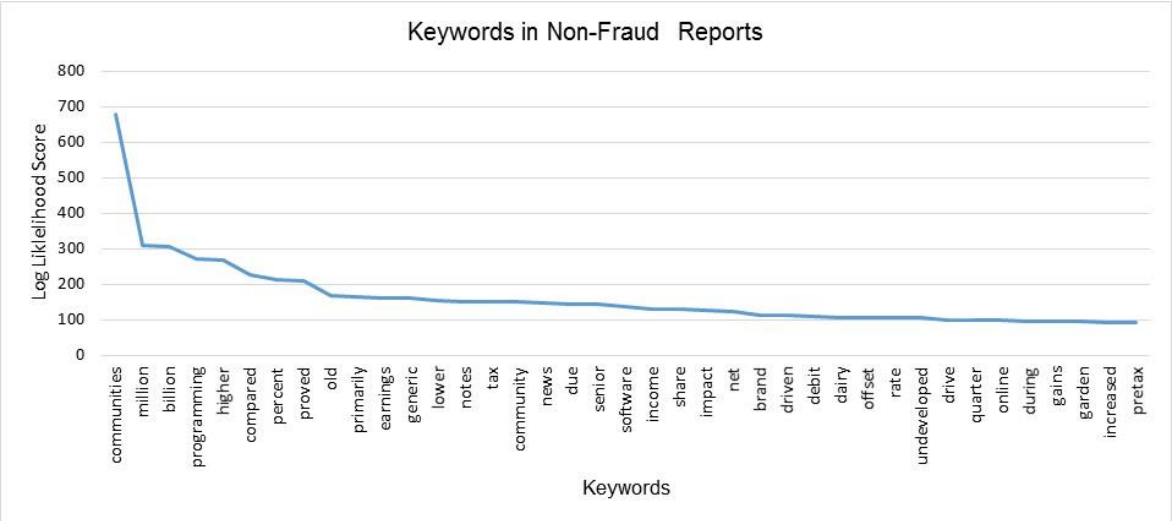


Figure 3.7: Keywords in non-fraud reports as identified using log likelihood score in AntConc.

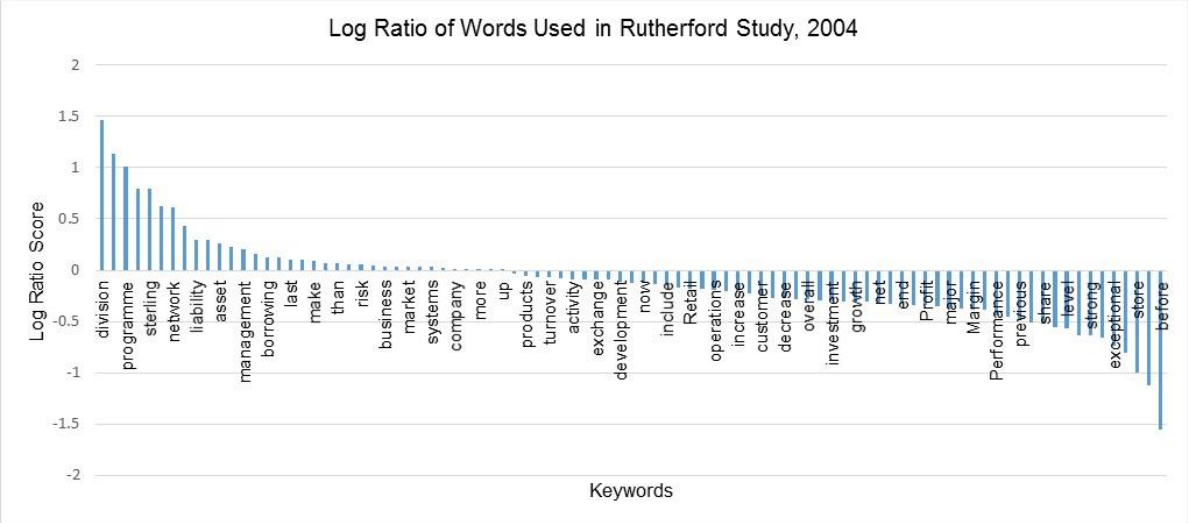


Figure 3.8: Log ratio scores [376] for keywords used in [199].

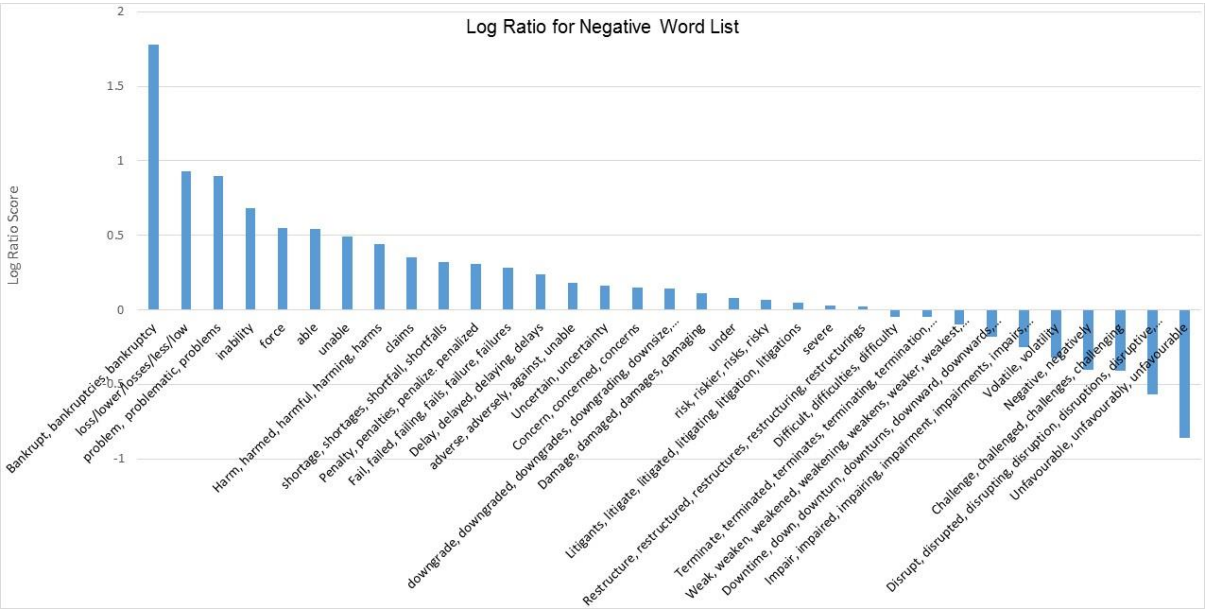


Figure 3.9: Log ratio scores [376] for negative words from [126, 143].

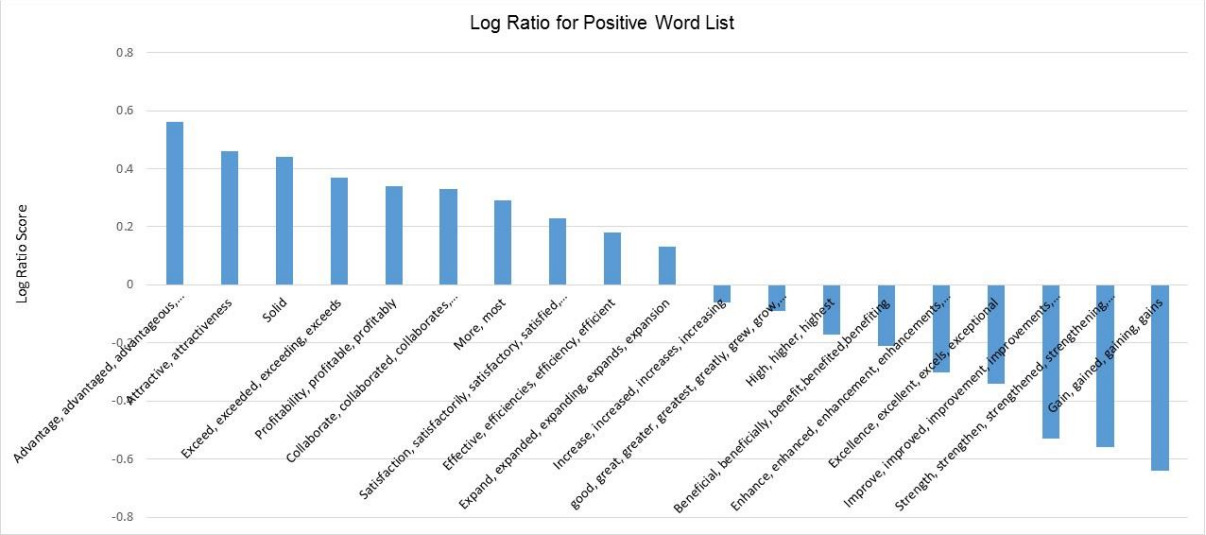


Figure 3.10: Log ratio scores [376] for positive words from [126, 143].

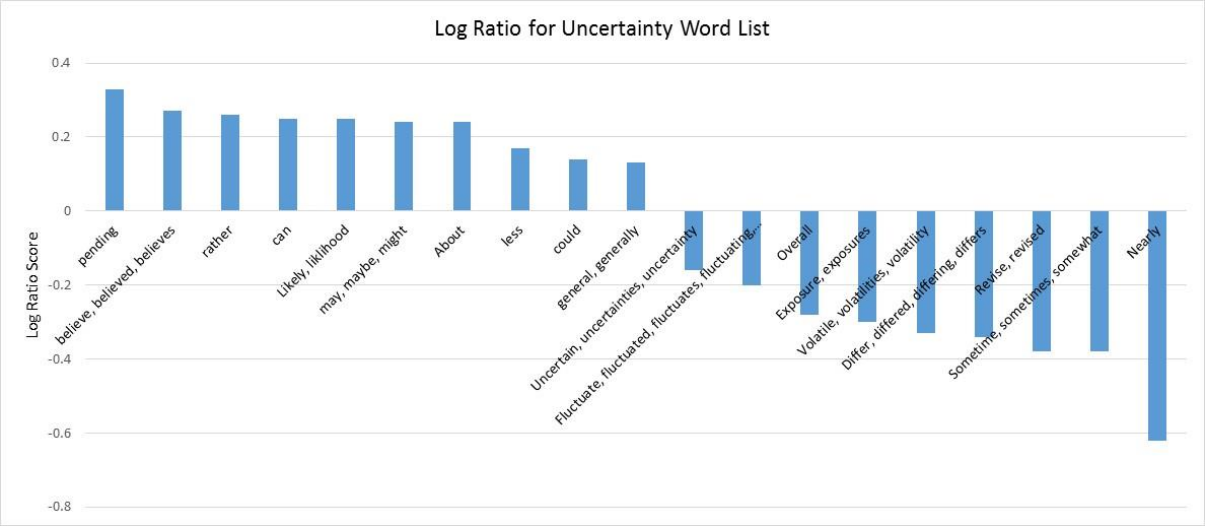


Figure 3.11: Log ratio scores [376] for uncertainty words from [126,143].

that were key in differentiating between companies with different attributes. Those same words are used here to check to see if there is a significant difference (as measured by Rayson's [198, 204] log likelihood calculator described above) in their use with respect to the fraud and non-fraud reports.

Appendix E (Table E.1 to Table E.9) contains all the results obtained from the log likelihood calculator. The log ratio scores were graphed, see Figure 3.8. The log-likelihood is a statistical significance measure – it tells us how much evidence we have for a difference between the two reports. However, it does not indicate how big/how important a given difference is. As described by Hardie [376] the log ratio would show up this difference. This ratio is calculated by dividing the relative frequency of a word found in the fraud reports with relative frequency of a word in the non-fraud reports. This ratio of relative frequencies is then converted into a binary logarithm [376].

The keyness scores shown in the Figure 3.8, depicts how big the difference between the two reports are for a particular keyword used in the Rutherford [199] study. The downward slope represents those words which are more prominent in the non-fraud reports.

The frequency of words in Loughran and Macdonald's [126] (positive, negative and uncertainty) word lists found in the corpus was noted and log likelihood and log ratio's calculated. The results for the most significant words are tabulated and are shown in Appendix E (Table E.10, E.11 and E.12) and graphically depicted in Figure 3.9, 3.10 and 3.11. Again, the downward slope represents those words which are more prominent in the non-fraud reports.

3.6 Collocations and Concordance

A central tenet of corpus linguistics is collocation analysis. It is: "*the above chance co-occurrence of two word forms*" [208] and: "*a natural extension of frequency lists*" [207]. Firth's [209] most judicious and enduring phrase: "*you shall know a word by the company it keeps*" is apt at capturing the essence of this approach. Collocations can indicate a semantic preference for certain constructions or can uncover meaning imbued in words by those words they collocate with [52] and thus give insights into the mental lexicon of the text producer [210]. Therefore: *"A collocation analysis therefore reveals discourse patterns and meanings that are neither evident from frequency lists*

of individual words nor from the readings of larger volumes of text in manual analysis" [206]. The strength of a collocation between two words can be measured by the mutual information (MI) score of these two words with a score higher than 3 being considered a strong collocation [211]. Mutual Information (MI) is calculated on the basis of the number of times you observed the pair together versus the number of times you saw the pair separately [6]. However, the MI score assigns higher scores to rare words that produce unique collocations than to collocations containing frequent words. This has to be factored into the analysis.

Collocation analysis on the corpus produce results shown in Appendix F. A node word is chosen and associated collocates examined. Two types of node words are chosen. A few words that were denoted as key through keyword analysis (using the log likelihood score) and a few words chosen that were high frequency words garnered from frequency inspection analysis. Some concordances or keywords in context were extracted and are also shown. This enables observation on what the frequency of the word is all about [188]. Node words 1-6 (in Appendix F, Figure F.1 to F.6) are keywords, some of their main collocates are graphed and example of concordances shown. Node words 1-3 (Appendix F, Figure F.7 to F.15) were the highest scoring keywords in the fraud reports. Node words 4-6 were the highest scoring keywords in the non-fraud reports. The top 50 collocates for node words 1-3 found in the fraud reports (as they were key in the fraud reports) are graphed and the same is done for node words 4-6. Each graph shows the MI score for the collocates of the node word in fraud and non-fraud reports. For each node word there is a table that denotes whether or not there is a significant difference in the occurrence of the collocates between the two reports.

Appendix F (Figure F.7 to F.15) shows the collocates and concordances for 9 high frequency words *('company*', '*business*', '*million*', '*operations*', '*sales*', '*cost*', '*financial*', '*products*', '*revenues')*. Appendix F, Figure F.16 shows significance testing performed on the mean of MI scores for collocates shown in Appendix F, Figure F.7 to F.15.

3.7 Discussion and Conclusion

A central tenet of corpus linguistics is that frequencies can only be meaningfully interpreted when compared to other frequencies [212]. Therefore the word type and

lemma frequencies of the fraud reports were compared with the word type and lemma of the non-fraud reports. At a high level of abstraction frequencies of word types display homogenous usage. However, this hides differences that are revealed when significance testing is performed. The *p* value clearly indicates for both the word types and lemmas that there are differences in word usage between the two report categories. From frequency analysis it seems that companies in the non-fraud category place greater emphasis on their operations. For example, there is a higher use of words such as '*market*', '*operation*', '*customer*', '*operate*', '*operations*', '*include*'. Greater use also of connectives '*primarily*' (used to elaborate and explain an argument). In the fraud reports, greater use of: '*capital*', '*may*', '*not*', '*acquisition*', '*end*', '*year*', '*fiscal*', '*price*', '*agreement*'. This is suggestive of a greater preoccupation with time ('*end*', '*year*', '*fiscal*'), an introduction of greater uncertainty through the use of '*may*' and '*not*' and a greater need for '*capital*' and an emphasis on '*acquisition*'.

An examination of the keyword analysis unveils the following observations:-

Words associated with financial performance (as selected by Rutherford [199]) such as '*loss*', '*result*' are more key in the fraud reports. Significantly under-used are '*performance*', '*margin*' and '*sale*'. Overall, this suggests that fraud firms are avoiding a comprehensive analysis of their operations. A lot greater focus on '*asset*', '*borrowing*', '*liability*', lower mention of '*debt*' and '*cash*' in fraud reports than non-fraud reports, suggestive of potential monetary issues.

Further, some words uncovered in the Rutherford [199] study are much more marked (by log ratio) in the in the non-fraud reports. Words such as '*tax*', '*share*', '*investment*', '*trading*' indicative of belonging to narratives that seek to describe operations. Words that are used to make comparisons such as '*increase*', '*decrease*', '*growth*', '*level*', '*due*' are all also higher again suggestive of text that seeks to provide analysis. Also '*overall*', '*total*' have high log ratios and suggest an attempt to fully describe state of affairs. Strong adjectives such as '*exceptional*', '*major*', '*strong*' are also more marked in non-fraud reports.

Keyness strength was also measured using a negative word list derived by Loughran and Mcdonald [126, 205]. Most of these words are overused in the fraud reports confirming previous research (discussed in chapter two of slightly greater negativity in deceptive text). Words such as '*bankruptcies*', '*loss*', '*problems*', '*inability*', '*shortages*' are more marked in fraud reports again suggestive of latent cash flow problems for example *"….no longer deemed collectible due to bankruptcies*" (fraud report). In non-

fraud reports words such as '*unfavourable*', '*disrupted*', '*challenge*', '*volatile*', '*impair*' are more key eg "…*company participates in a highly volatile industry*" (non-fraud report). Again suggestive of describing the business environment and the challenges posed to the company.

Using the positive word list in fraud reports '*advantage*', '*attractive*', '*solid*', '*profitable*' are more marked concealing perhaps a concern over company health and a degree of spin to portray company operations in the best light for example: " *we are committed to delivering profitable revenue and improved value for*", "*takes advantage of business opportunities."* Whereas the non-fraud reports use: '*exceed*', '*higher*' terms often used with a numerical value to denote a target for example: "….*cost of renovation could exceed $1 million*" or *"increased by $0.1 million primarily due to higher average cash balances held during"* (non-fraud report). Words used also denote more confidence such as '*excellence*', '*strength*', '*great*', '*beneficial*'.

From the uncertainty word list, words such as '*pending*', '*likely*', '*may'*, '*rather*' are terms that suggest an avoidance of detail are higher in fraud reports for example: *"…effectiveness of these rules pending the outcome of the appeal from", "…in some cases this process may require the holder", "this requirement is likely to affect the construction or expansion"* (fraud report). Greater use of words such as '*generally*' indicate a gloss over details. In non-fraud reports again '*volatile*', '*fluctuate*' are used to alert perhaps to uncertainty in the environment for example *"….the reasons our quarterly results may fluctuate include general competitive and economic environment"* (non-fraud report).

The keyness scores obtained via AntConc show that fraud companies over-use the word '*procedure*' and '*agreement*' indicating a preoccupation with how actions should be governed for example *"Operating policies and procedures are substantially the same at each*" or *"term of the long term supply agreement If we obtain FDA approval to".* They are also more focused on '*system*' as in information system for example: "*a quality manual and quality control system and employ best practices in biodiesel".* It is likely that all companies in the fraud set are likely to have franchise agreement, therefore there is a higher mention of the word '*franchise*' for example: *"Beginning in 2006, an initial franchise fee for basic services was assessed."* Further words such as '*stock*', '*stockholder*', '*securitization*', '*capital*' are all indicative of a concern with capital markets: *"to borrow funds from our initial stockholder to operate. Our initial stockholders a", "primarily due to a decrease in securitization fees due to a decrease*

in". This again is suggestive of potential cash flow issues with a consequent requirement to raise capital. '*Obtain*' is also higher indicative of a need to get access to resources for example *"facilities are required to obtain air emisssion permits for operation under"* (fraud report).

Whilst the non-fraud reports have a higher mention of the keyword '*communities*' for example: *"staff, are a part of the communities we serve. We are active in".* This indicates a greater awareness of the public image and corporate social responsibilities. Also, as already mentioned and confirmed through AntConc Keyword analysis, non-fraud reports also have higher mention of '*billion*', '*higher*', '*net*', '*percent*', '*tax*', '*pretax*' pointing to a greater emphasis on numerical quantification of business operations.

Collocation analysis reveals that collocates of some keywords and high frequency words have significantly different usage. Keywords 1-3 ('*procedures*', '*system*' and '*acquisition*') and their collocates are much more pronounced in the fraud reports. This is reinforced through significance testing which takes the mean of the mutual information score for fraud/non-fraud reports (Appendix F, Figure F.1 to F.3). Mutual Information score takes into account both the individual frequency of the words that make up the collocation and the frequency with which these words were observed together. This finding reinforces the view espoused here that fraud firms preponderantly dwell on bureaucratic issues.

Whereas collocates of keywords 4-6 (*'communities', 'higher', 'billion'*) in the non-fraud reports have significantly different usage patterns in the fraud reports. Again this emphasizes the view delineated earlier that non-fraud firms focus on social issues, they seem to be doing comparative analysis and attempt to quantify results/performance. Significance testing on the collocates of node words 4-6 reveal statistically significant patterns of usage (Appendix F, Figure F.4 to F.6 and F.16).

Collocations that contain high frequency words also have some marked out differences in usage patterns between the fraud and non-fraud firms as shown in Appendix F (Figure F.7 to Figure F.14). These differences are confirmed through significance testing as shown in Appendix F, Figure F.16. This again underlines the view that firms from these two categories emphasis different part of their operations. This can be somewhat gauged through examination of concordances as shown in Appendix F.

‘*Company*’ has similar rates of usage in both reports with similar normalised frequency. The collocates indicate that firms are providing a description of their activities. ‘*Million*’ is under-used in fraud reports (lemma, normalised frequency of 5.8 words per thousand) as opposed to non-fraud reports (lemma, normalised frequency of 7.1 words per thousand). This seems to again indicate that non-fraud firms provide more quantification. The collocates for ‘*million*’ (Appendix F, Figure F.7) give a snapshot of usage and shows that non-fraud firms focus have lower to no mention of “*deduction*’, ‘*corrected*’, ‘*collateralizing*’, ‘*transform*’ as compared to fraud firms (see graph shown in Figure F.7, Appendix F). This can be meaningful given fraudulent intent. Node word ‘*business*’ has a normalised frequency of 3.7 words per thousand words in both the reports. However, collocation such as ‘*business buys’*, ‘*business seems*’, ‘*business savy*’ are used more frequently in fraud reports. Other high frequency node word ‘*products*’, ‘*sales*’, ‘*financial*’ also show similar usage patterns (3.4 words per thousand, 2.8/2.9 words per thousand, 2.5/2.6 words per thousand respectively) with similar usage patterns with collocates in both reports. Higher mention of node word ‘*operations*’ in non-fraud reports (2.4 words per thousand) as opposed to 2.8 words per thousand. This again emphasizes that non-fraud firms focus more on their operations than fraud firms. The collocates show how usage varies in the two reports, it indicates description of activities. Node word ‘*revenue*’ shows stable patterns of usage in both fraud and non-fraud reports (1.8 and 1.7 words per 1000 respectively). However collocations such as ‘*slower revenue’*, ‘*revenue shortfalls*’, ‘*modest revenue*’ have higher mutual information scores in the fraud reports, indicating higher usage. Node word ‘*costs*’ has a higher usage in non-fraud reports with (1.9 to 1.3 words per thousand for fraud and non-fraud reports respectively). Collocation such as ‘*cost curve*’, ‘*cost profiles*’, ‘*cost estimation*’, ‘*cost producer*’, ‘*cost opportunity*’ all have a higher MI score in fraud reports, indicative of higher usage linked to higher costs of latent poor company health in the fraud firm category.

Chapter Four

DOCUMENT REPRESENTATION

"Machine Learning is all about using the right features to build the right models that achieve the right tasks"

Flach (2012)

4.1 Introduction

Successful execution of text classification is dependent on three factors. The classification model, similarity measure and document representation model [213]. As Keika et al. [213] showed the choice of document representation has a profound impact on the quality of the classifier. This step requires the documents to be in a more compact and computationally appropriate form. This representation has been attempted in a myriad of ways, though typically a document is represented: "*as a collection of terms: words, stems, phrases or other units derived or inferred from the text of the document*" [214]. In this chapter the 408 reports both fraud and non-fraud will be mapped out into 10 document representation schemes, shown in Figure 4.1. Each scheme will be delineated with data from the corpus. The composition of the corpus as indicated is made up of 408 reports – 102 fraud and 306 non-fraud. The classifiers will be run over 2 following combinations of this data:-

- Each fraud firm is matched with a similar non-fraud firm, so for the 102 fraud reports there will be 102 non-fraud reports. This data set up is referred to as matched pair design.
- Each fraud firm will be matched with 3 similar non-fraud firms, so for the 102 fraud reports there will be 306 non-fraud reports. This data set up is known as peer set design.

Each of the 10 representation schemes as shown in the large rectangular box in Figure 4.1 will be setup in a matrix for both the peer set and matched pair design composition. For example, for bigram representation, a matrix will be formed for 102 fraud reports and 102 non-fraud reports. This constitutes 204 rows in the matrix with the bigrams being the columns. For the peer set design this would be 408 rows (102

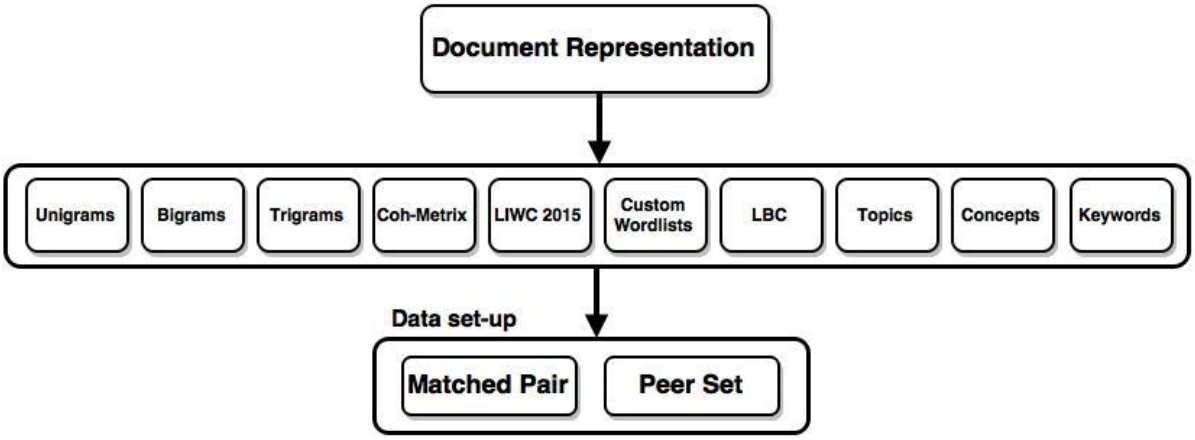


Figure 4.1: Document representation schemes

fraud reports and 308 non-fraud reports) with again the columns being the bigrams. This setup would enable an appreciation of the strength of the features used to represent the documents. In other words as the non-fraud narratives increase the linguistic features extracted through the document representation schemes should still be strong enough to enable classification of fraud and non-fraud reports.

The document representation schemes were implemented using the text miner (tm) package in R and the Natural Language Processing Toolkit (NLTK) in Python. Where necessary, extracts of code will be given to convey how the documents are transformed to features, readied for classification. A goal for all the representation schemes is to reduce the dimensions of data which can results in the classifier overfitting the data. This will be expanded further in Chapter 5. The general aim is to have a small and representative set of features for the reports.

The purpose for this exposition into document representation is that an answer is sought to the question: "*What is a good representation of the reports*?" A good representation would be that which gives the highest classification accuracy in distinguishing a fraud from a non-fraud report. The features extracted in this chapter would be representative of the reports and then according to the framework shown in chapter 3 would be put forth for feature selection. As Flach [215] argues features define a '*language*' in which we describe the relevant objects in our domain. Therefore once the features are extracted the reports, the narratives no longer matter. It is vital though that a suitable feature representation process be enacted.

Kumar et al (2016) stress that selecting the right features enhances interpretability of the model. It can improve the performance of learning algorithms and also help in reducing the computational complexity of the model. They demonstrate this in the financial forecasting domain. They propose four hybrid prediction models that are combinations of four different feature selection techniques. The empirical results obtained indicate that model performance is related to the feature set chosen.

4.2 The Vector Space Model (VSM)

Underpinning the document representation schemes outlined in this chapter is the Vector Space Model. This builds on the mathematical framework of vector spaces and linear algebra and significantly it enables the quantification of distance and similarity [218]. As Clark [22] elaborates it enables a separation of words that have different meaning. For example sentences with terms such as '*profit*', '*gain*', '*increase*' would be more closer in meaning than '*loss*', '*drop*', '*degenerate*' and this can be depicted geometrically by showing up the distance between words that are similar/non similar. The underlying distributional hypothesis is that: "*words that occur in similar context tend to have similar meanings*" [218] or as Firth [21] put it: "*you shall know a word by the company it keeps*". This method constitutes the dominant information retrieval technique for detecting the relevant documents to a keyword query [49]. Both query (treated as pseudo document) and documents in a collection are represented as points in space (a vector in vector space). "*Each document* $d_i \in D$ *is then represented as a vector* $V_{d_i} = (v_1, v_2, \dots, v_{|w|})$ *of size* $|W|$ *with its j-th dimension* v_j *quantifying the information that the j-th term* $w_j \in W$ *conveys for* d_i" [217].

Points that are close together in this space are semantically similar and points that are far apart are semantically distant [218]. The documents are then sorted out in order of increasing distance from the query. Figure 4.2 illustrates this point, it is simplified to represent only two document vectors d1, d2 and a simple query vector. The space in the corpus under study would contain a huge number of terms {t1,t2,t3….tn} but in Figure 4.2 only two terms are represented by an axis for each term: "*the document d1 has components {t1,t3…} and d2 has components {t2,…}. So V(d1) is represented closer to axis t1 and V(d2) is closer to t2. The angle θ represents the closeness of a*

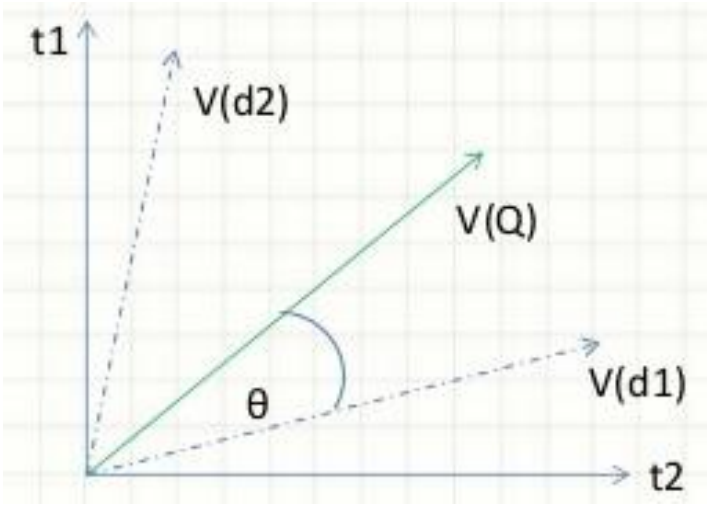


Figure 4.2: Classical vector space model [48]

document vector to the query vector V(Q). Its value is calculated by the cosine of θ" [218].

The terms are the features that best characterize the document and can be anything from strings of length *N*, single words, phrases or any set of concepts or logical predicates [299]. The features chosen for the documents in this study are those extracted from the 10 document representation schemes shown in Figure 4.1. The term information in each dimension can come in any of the following forms [217].

- A binary value indicating the existence (or absence) of a term in the corresponding document.
- An integer value indicating the number of occurrences of a term in a document (i.e., Term Frequency).
- A Term Frequency-Inverse Document Frequency (tf-idf) value.

For the document representation schemes detailed in this chapter, only option 2 and 3 of the above will be used as they are the more commonly used in constructing vector space models [49]. According to Turney and Pantel [218] the defining property of VSMs are that the values of the elements in a VSM must be derived from event frequencies, such as the number of times that a given word appears in a given context. This requirement is met by the representation schemes outlined in this chapter.

The (tf-idf) score is the most common term weighting approach used in VSMs and is calculated as shown below:-

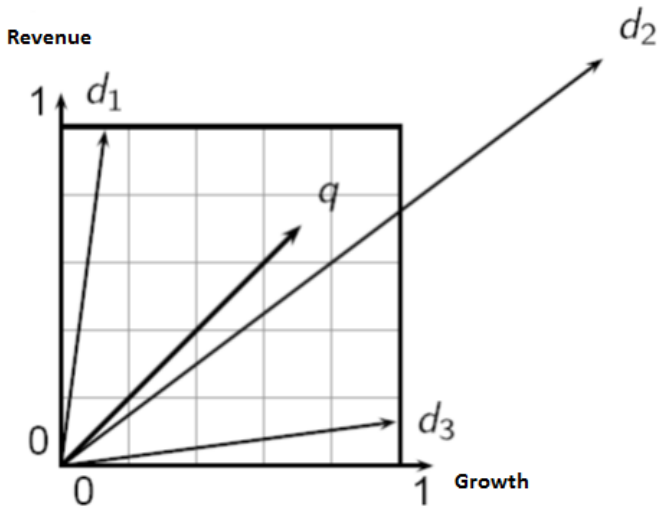


Figure 4.3: Euclidean distance used to determine similarity between query and documents.

$$w_{d,t} = tf_{d,t} * idf_t \quad \text{(Eq 4.1)}$$

The *tf* value gives a raw frequency of a term. However, this by itself is inadequate as some words like '*this*', '*a*' occur very frequently across all documents. The *idf* portion gives a measure of the uniqueness of the word, in other words how infrequently the word occurs across all documents (inverse document frequency or *idf*). Words in the document with a high *tf-idf* score occur frequently in the document and provide the most information about that specific document.

The next question that arises is that how can the proximity of documents be measured in this space? Cosine Similarity and Euclidian Distance are two different ways to measure vector similarity. The former measures the similarity of vectors with respect to the origin (the direction), while the latter measures the distance between particular points of interest along the vector (the magnitude). Euclidean distance measures the distance between particular points of interest along the vector. The lower the distance between 2 points, then the higher the similarity. Distance between two points is measure using formula below:-

$$|\overrightarrow{X} - \overrightarrow{Y}| = \sqrt{\sum_{i=1}^{n}(Xi - Yi)^2} \quad \text{Eq (4.2)}$$

However an issue with Euclidean distance is illustrated in Figure 4.3. The distance between *q* and *d2* is large even though the distribution of terms in the query *q* and the distribution of terms in the document *d2* are very similar. A way to tackle this is too

calculate the cosine between the two vectors, V(Q) and V(d1) as shown in Figure 4.2. The formula used to perform this calculation is shown below:-

$$\cos(\vec{X},\vec{Y}) = \frac{\vec{X}.\vec{Y}}{|\vec{X}|\,|\vec{Y}|} = \frac{\sum_{i=1}^{n} X_i Y_i}{\sqrt{\sum_{i=1}^{n} X_i^2}\sqrt{\sum_{i=1}^{n} Y_i^2}} \qquad \text{Eg (4.3)}$$

$\vec{X}.\vec{Y}$ is the dot product between these two vectors or cosine of the angle between $\vec{X}.\vec{Y}$ The denominator in the equation normalised by dividing each of the vectors by its length. This makes it a unit vector and thus a better measure of the true difference between documents. Cosine captures the idea that the length of the vectors is irrelevant; the important thing is the angle between the vectors [218]. However in some instances Euclidean measure may prove to be illuminating. For example, the cosine between a document which has '*machine*' occurring 3 times and '*learning*' 4 times and another document which has '*machine*' occurring 300 times and '*learning*' 400 times will hint that the two documents are pointing in almost the same direction. This gives us an appreciation of topic similarity. If magnitude was taken into account, the results would be quite different. Euclidean measures the distance between particular points of interest along the vector. The lower the distance between 2 points, then the higher the similarity. In this study the classifiers used can be deployed using both measures. Results are tabulated for the distance measure that produced better results. In this corpus fraud firms are compared with non-fraud firms of similar size and industry. This should in theory produce reports of similar length and topic. Therefore using either measure would alert to anomalies.

4.3 Bag of Words (BoW) - Unigrams

This is the standard method of text representation for document representation for most NLP tasks. Typically, it constitutes all the word types in the document with a corresponding value that encodes how often it occurs in the document Its appeal seems to be its: "*simplicity, efficiency and often surprising accuracy*" [219].

The document as the name implies is reduced to a bag of words, so grammar and syntax, all word order is lost. Sequences of words of length *n* are known as n-grams. N-grams of texts are extensively used in natural language processing tasks. They are a set of occurring words within a given window. Figure 4.4 shows the steps taken to produce a term document matrix for n-grams (unigrams, bigrams and trigrams).

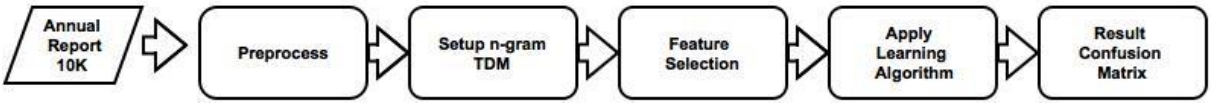


Figure 4.4: N-gram extraction from corpus and supplication of downstream processes.

γ = (

profit	10
acquisition	20
financial	30
significance	55
end	99

) = C

γ = Classifier

C = Fraud or Non-Fraud

Figure 4.5: Classification set-up for BOW unigram model.

A unigram is just one single word. For unigrams, the learning algorithm used would employ a feature set as depicted in Figure 4.5 to aid in classifying new documents.

The code from the extract in Figure 4.6 using the tm module in R shows the pre-processing that is typical of a bag of words approach.

As can be seen from the extract, the reports are stripped of punctuation (line 4), numbers (line 6), stop-words (line 7) and white space (line 9). Stop-words are words that are not content bearing words and in this approach are deemed to be just 'noise'. Typically stop-words include words such as '*a*', '*the*', '*to*'. Bird et al. [220] list stop-words typically removed in n-gram modelling tasks. Similarly, in this approach, numbers, punctuation and white space are again detractors that provide little to no value-add in the classification process. Further in an attempt to reduce dimensions of the data, stemming is undertaken. This is a: "*crude heuristic process that chops off the ends of the words*" [49]. For example words such as '*accounts*', '*accounting*' are reduced to '*account*', they also change the surface form of a word into meaningless forms such as '*having*' to '*hav*'. As in bag of words using unigrams, meaning is lost through jettisoning word order. The stemming helps in the formation of word clusters that could better aid the classification process. This pre-processing enables a reduction in the entropy of a system. For example, if you consider both upper-case and lower-case letters and perform no stemming would this would detract from

```
#clean text
cleancorpus <-function(corpus){
  corpus.tmp <- tm_map(corpus, removePunctuation)
  corpus.tmp <- tm_map(corpus.tmp, content_transformer(tolower))
  corpus.tmp <- tm_map(corpus.tmp, removeNumbers)
  corpus.tmp <- tm_map(corpus.tmp, removeWords, stopwords("english"))
  corpus.tmp <- tm_map(corpus.tmp,  stemDocument)
  corpus.tmp <- tm_map(corpus.tmp, stripWhitespace)
  return(corpus.tmp)
}
```

Figure 4.6: Pre-processing steps for n-gram modelling in R [379].

```
#build TDM

generateTDM <- function(rep, path){
  s.dir <- sprintf("%s/%s", path, rep)
  s.cor <- VCorpus(DirSource(directory = s.dir),
  readerControl = list(reader=readPlain))
  s.cor.cl <- cleancorpus(s.cor)
  s.tdm <- TermDocumentMatrix(s.cor.cl,control = list
  (weighting = function(x) weightTfIdf(x, normalize =TRUE)))
  s.tdm <- removeSparseTerms(s.tdm, 0.7)
  result <- list(name = rep, tdm = s.tdm)
}

tdm <- lapply(reports, generateTDM, path = pathname)
```

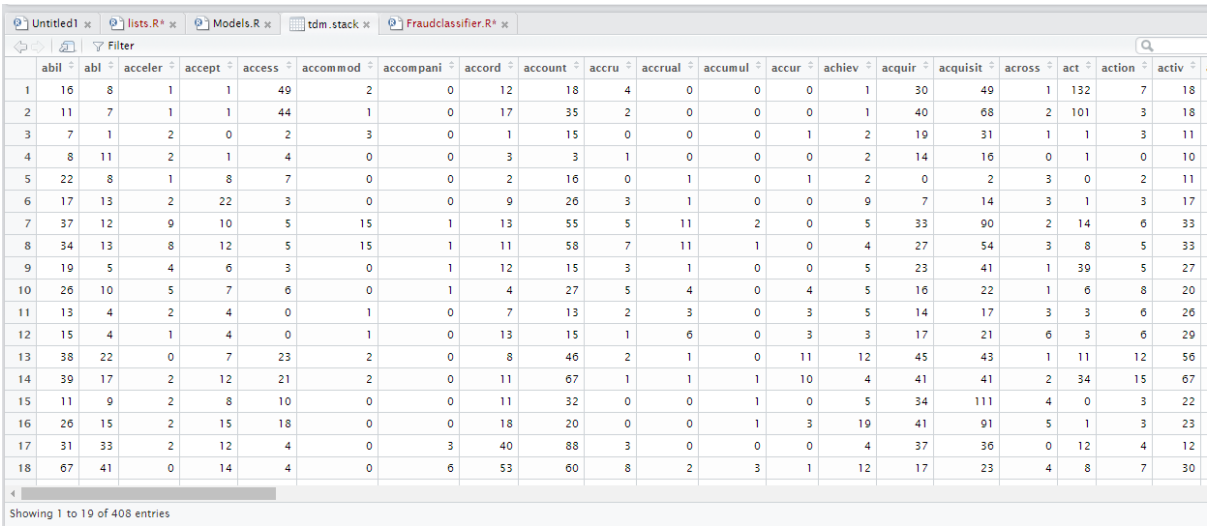
Figure 4.7: R code to generate TDM [379].

the detection of essential concepts that are carried in a word.

Entropy reduction techniques are also important from a machine learning perspective as it allows greater generalisation (to be covered in chapter 6).

Once the above pre-processing steps have been encoded in a function. The program continues, as shown in extract in Figure 4.7. Line 27 calls the function generateTDM which runs from line 16 to line 24. Line 17 and Line 18 identifies the location of the corpus and stores content in variable s.cor. In line 19, the clean corpus function in Figure 4.6 is executed. Line 21 sets up a term document matrix.

This is a two-dimensional matrix whose rows are the documents and columns are the terms or stemmed words, so each entry (*i, j*) represents the *tf-idf* of term *i* in document *j.* This *tf-idf* score enables the detection of terms that are significant in a



Untitled1 | lists.R* | Models.R | tdm.stack | Fraudclassifier.R*

Filter

	abil	abl	acceler	accept	access	accommod	accompani	accord	account	accru	accrual	accumul	accur	achiev	acquir	acquisit	across	act	action	activ
1	16	8	1	1	49	2	0	12	18	4	0	0	0	1	30	49	1	132	7	18
2	11	7	1	1	44	1	0	17	35	2	0	0	0	1	40	68	2	101	3	18
3	7	1	2	0	2	3	0	1	15	0	0	0	1	2	19	31	1	1	3	11
4	8	11	2	1	4	0	0	3	3	1	0	0	0	2	14	16	0	1	0	10
5	22	8	1	8	7	0	0	2	16	0	1	0	1	2	0	2	3	0	2	11
6	17	13	2	22	3	0	0	9	26	3	1	0	0	9	7	14	3	1	3	17
7	37	12	9	10	5	15	1	13	55	5	11	2	0	5	33	90	2	14	6	33
8	34	13	8	12	5	15	1	11	58	7	11	1	0	4	27	54	3	8	5	33
9	19	5	4	6	3	0	1	12	15	3	1	0	0	5	23	41	1	39	5	27
10	26	10	5	7	6	0	1	4	27	5	4	0	4	5	16	22	1	6	8	20
11	13	4	2	4	0	1	0	7	13	2	3	0	3	5	14	17	3	3	6	26
12	15	4	1	4	0	1	0	13	15	1	6	0	3	3	17	21	6	3	6	29
13	38	22	0	7	23	2	0	8	46	2	1	0	11	12	45	43	1	11	12	56
14	39	17	2	12	21	2	0	11	67	1	1	1	10	4	41	41	2	34	15	67
15	11	9	2	8	10	0	0	11	32	0	0	1	0	5	34	111	4	0	3	22
16	26	15	2	15	18	0	0	18	20	0	0	1	3	19	41	91	5	1	3	23
17	31	33	2	12	4	0	3	40	88	3	0	0	0	4	37	36	0	12	4	12
18	67	41	0	14	4	0	6	53	60	8	2	3	1	12	17	23	4	8	7	30

Showing 1 to 19 of 408 entries

Figure 4.8: Extract of TDM generated for unigrams (stemmed).

	Unigrams
Matched Pair	204 rows by 1425 variables
Peer Set	408 rows by 1391 variables

Table 4.1: Dimensions of TDM matrices set up for unigrams.

report in comparison to the rest of the reports in the corpus. In the code, shown in Figure 4.6 and 4.7 the frequency data generated for the term document matrix (tdm) matrix are tf-idf scores. Only tf-idf scores are used with the bag of words models as it has been shown to outperform raw frequency [22, 217, 218].

Line 23 removes sparse terms by 30% for example if a term contains ‘0’ at 30% then it is discarded. This reduction in the dimensions of the matrix improve classifier performance.

Line 24 calls the generateTDM function which in turn calls the cleancorpus function (shown in Figure 4.6). Extract of the tdm generated is shown below Figure 4.8.

The numerical values are the tf-idf frequencies. Each row as a class field with a string value ‘f’ for fraud and ‘nf’ for non-fraud. The matrices (TDM) generated have the dimensions shown in Table 4.1.

Appendix G (Table G.1) shows the most prominent stems in fraud reports for the matched pair data set-up. Once the TDM was formed, each column for fraud reports

only in the matrix (containing tf-idf scores for a stem) was added up. This action was also performed for non-fraud reports. Taking each summed up tf-idf score for a stem in the fraud reports, the corresponding stem/tf-idf score was found in non-fraud reports. Appendix G (Table G.2) performs the same process but this time looks first at the non-fraud reports and picks out the top 100 most prominent stems, sums up tf-idf scores for each stem and finds the corresponding stem/tf-idf in fraud reports. Figure 4.9 and Figure 4.10 graphically depicts the tf-idf (summed) values (column 3 and 4) from Table G.1 and Table G.2 for a number of stems.

The above process was repeated for the peer set data set up but the tf-idf scores were averaged instead of summed as there are more non-fraud reports than fraud. This allows for better comparison. Appendix G (Table G.3) shows the top 60 most prominent stems in fraud reports and their corresponding stems in non-fraud reports with the averaged tf-idf scores. Table G.4 shows results from the top 60 most prominent stems in non-fraud reports with the corresponding scores from fraud reports. Figure 4.11 and Figure 4.12 graphically depicts the tf-idf (averaged) values (column 3 and 4) from Table G.3 and Table G.4 for a number of stems.

Both feature selection and feature transformation methods, in a mutually exclusive manner are executed over these matrices to reduce dimensionality and improve the subsequent performance of the classifiers. These methods are discussed in Chapter 5. The classifier results are then shown in Chapter 6.

4.4 Bigrams and Trigrams

A bigram is a word pair. A trigram is a three word sequence. The bigrams within a sentence are all possible word pairs formed from neighbouring words in the sentence. In this sentence: "*The significance of this investment is considerable*" (Polly Peck, Annual Report 1989) the identified bigrams would be: '*the significance*', '*significance of*', '*of this*', '*this investment*', '*investment is'*, '*is considerable'*. The trigrams would be: '*the significance of*', '*of this investment*', '*investment is considerable'*.

As can be deduced, the bag of words unigram representation scheme pulls the words out of their sentence context with resulting loss of information. Using bigrams and trigrams would allow a more contextual information to be kept by representing

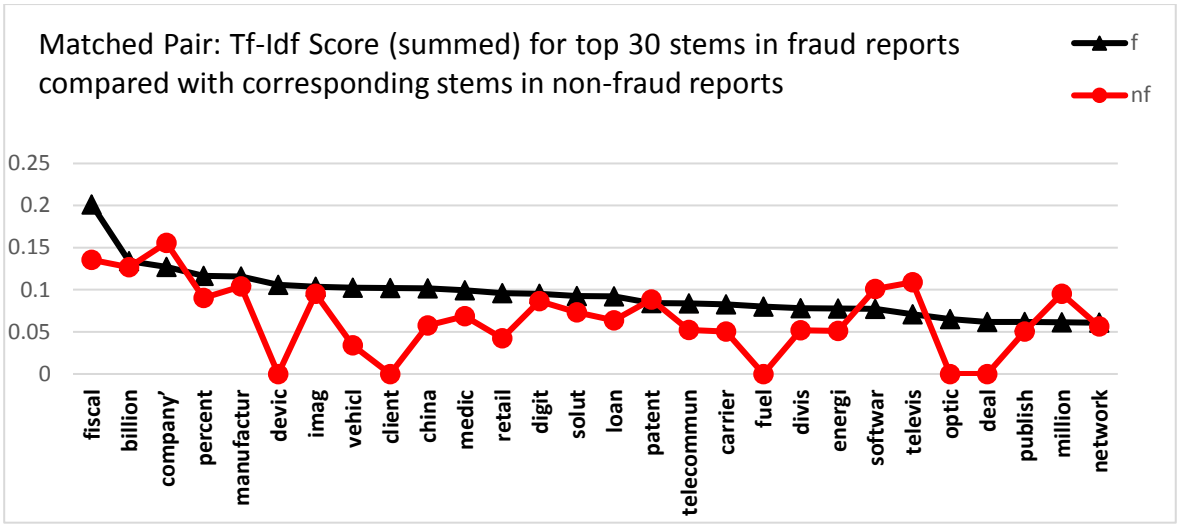


Figure 4.9: Most prominent stems in fraud reports compared to corresponding stem in non-fraud reports (matched pair set up).

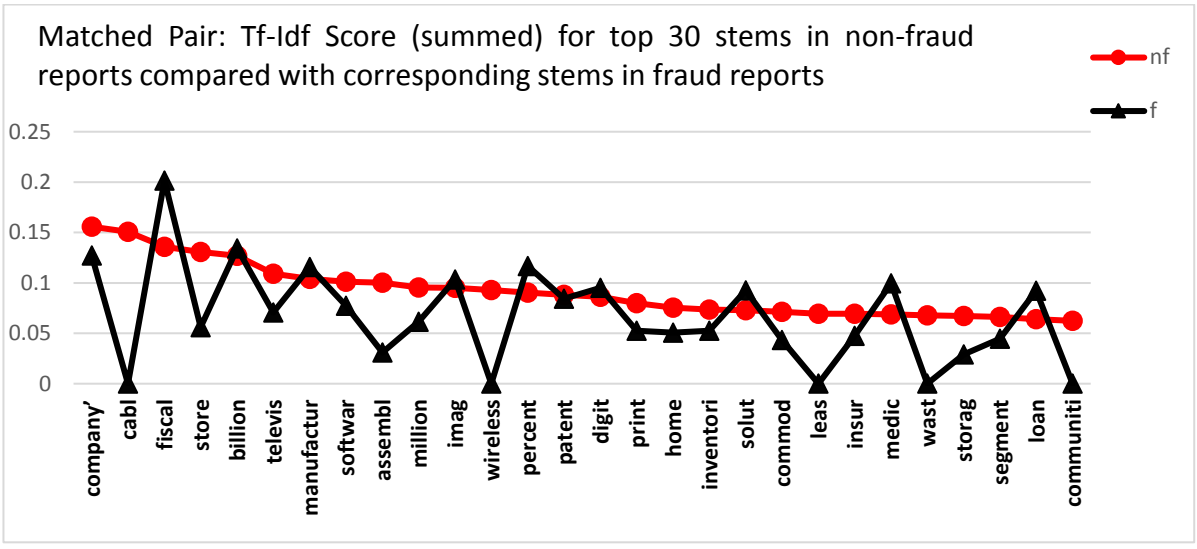


Figure 4.10: Most prominent stems in non-fraud reports compared to corresponding stem in fraud reports (matched pair set up).

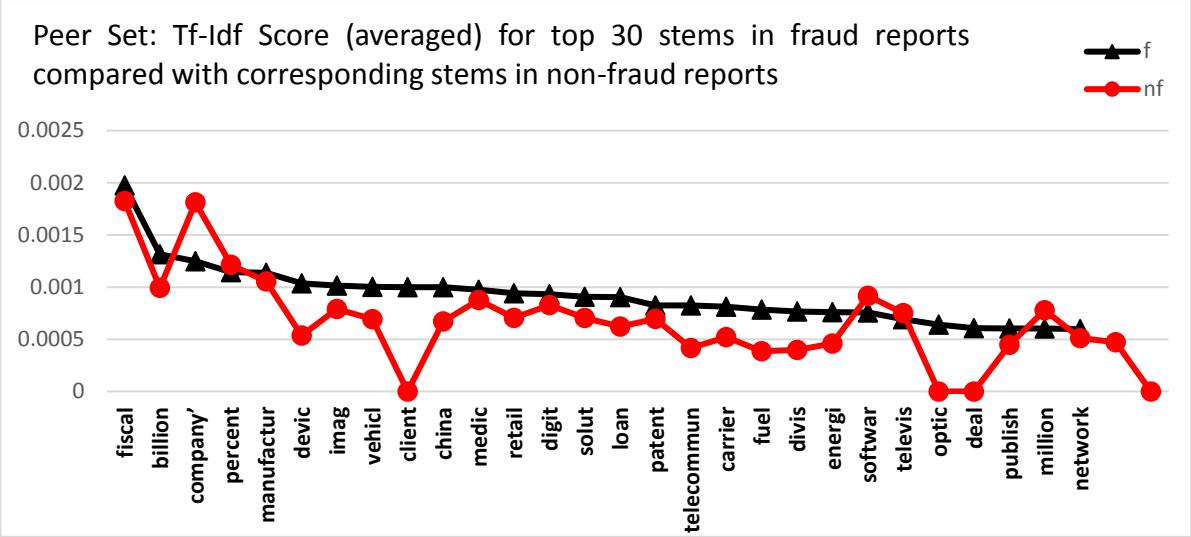


Figure 4.11: Most prominent stems in fraud reports compared to corresponding stem in non-fraud reports, for peer set data set up.

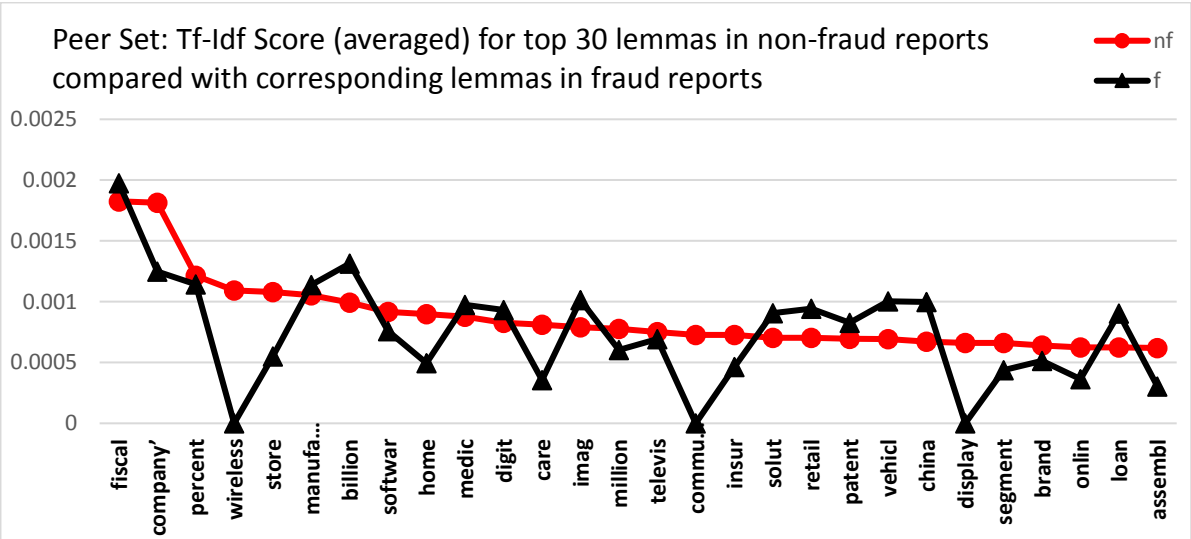


Figure 4.12: Most prominent stems in non-fraud reports compared to corresponding stem in fraud reports, for peer set data set up.

the text as word pairs [221]. There are experimental results by Tan et al. [222], Wang and Manning [223] that indicate that the use of bigrams in the text classification task results in higher performance than the unigram bag of words model.

Other such as Bekkerman [224] and Zhang et al. [221] report marginal improvements. The literature indicates that using n-grams beyond trigrams reduces text classification performance as co-occurrence patterns at larger lengths are not detected, increasing sparsity of the TDM [17].

The bag of words (unigrams) as described in section 4.3 destroys the semantic relations between words. Phrases such '*annual reports*', '*balance sheets*', '*financial year ending*' are represented in the BoW as separate words so their meaning is lost. A document representation scheme that contains these phrases will much better illuminate the topic of discussion as they are semantically richer. Such an additive use of unigrams results in better word sense disambiguation.

As with the BoW the document is again a bag of these basic building blocks. Again the corpus is loaded into R text miner (tm) and pre-processed, the code extract is shown in Figure 4.13.

The same sequence of steps for unigrams as shown in Figure 4.4 is undertaken for bigrams and trigrams. The pre-processing executed is shown in Lines 23 to 25 which removes numbers, punctuation and transforms the text to lower case. Differently from unigrams stopword removal and stemming is not used as it would give spurious bigram and trigrams that are unrepresentative of the corpus. Line 29 splits the text into 2 or 3 or more n-grams. The parameter (Weka_control enables a specification of values to be passed (2 for bigrams, 3 for trigrams and so on). Once the n-grams have been attained they are used to generate a term document matrix. As before for the unigrams, a term document matrix is generated that contain rows (the annual reports/10K for each company, with the columns being the n-grams (bigrams or trigrams), with the cell being tf-idf counts (set at line 32). The matrix is again reduced to get rid of terms that contain zeros, only terms included that have a non-zero entry at a proportion of 75 per cent in the term document matrix (tdm). The tdm is then converted to an R matrix, this facilitates greater statistical analysis on the data. Table 4.2 shows the dimensions of the resultant matrices.

An extract of a bigram TDM is shown in Figure 4.14 and an extract of trigram matrix shown in Figure 4.15. Using the same analysis as was done for unigrams, Appendix G (Table G.5 and G.6) shows prominent bigrams in fraud and non-fraud reports for

```
reports2 <- c("f","nfall")
pathname2 <- "C:/data5"
s.dir2 <- sprintf("%s/%s", pathname2, reports2)
s.cor2 <- VCorpus(DirSource(directory = s.dir2),
readerControl = list(reader=readPlain))
s.cor2 <- tm_map(s.cor2, content_transformer(tolower))
s.cor2 <- tm_map(s.cor2, removeNumbers)
s.cor2 <- tm_map(s.cor2, removePunctuation)
tdmf <- TermDocumentMatrix(s.cor2)
tdmf <- as.matrix(tdmf)
TrigramTokenizer <- function(x)
NGramTokenizer(x, Weka_control(min = 2, max = 2))
tdmf <- TermDocumentMatrix(s.cor2, control = list
(tokenize = TrigramTokenizer, weighting =function(x)
weightTfIdf(x, normalize =TRUE )))
tdmf <- removeSparseTerms(tdmf, 0.75)
s.mat <-as.matrix(tdmf)
s.mat <- t(s.mat)
sdfotWSt <- as.data.frame(s.mat, stringsASFactors = FALSE)
```

Figure 4.13: Code for bigram processing.

	Bigrams	**Trigrams**
Matched Pair	204 rows by 3500 variables	204 rows by 1086 variables
Peer Set	408 rows by 3356 variables	408 rows by 1040 variables

Table 4.2: Dimensions of TDM matrices set up for bigrams and trigrams.

matched pair data set up with comparison to the corresponding bigrams in the opposite report category. A few bigrams from Table G.5 (columns 3 and 4) in Tables G.5 and G.6 are graphed and shown in Figure 4.16 and Figure 4.17. Appendix G (Table G.7 and G.8) show prominent bigrams in fraud and non-fraud reports with comparison to the corresponding bigrams in the opposite report category for the peer set data set up. A few bigrams from Table G.7 (columns 3 and 4) and Tables G.8 are graphed and shown in Figure 4.18 and Figure 4.19.

Similarly for trigrams in the matched pair fraud category the most prominent trigrams were aligned with the corresponding trigrams/reports. The results are tabulated in Appendix G, Table G.9 and G.10. A few choice trigrams from columns 3 and 4 from these tables are plotted and results shown in Figure 4.20 and Figure 4.21. Lastly for trigrams in the peer set fraud category the most prominent trigrams were aligned with the corresponding trigrams/reports. The results are tabulated in Appendix G, Table G.11 and G.12. Results are also shown in Figure 4.22 and Figure 4.23.

4.5 Coh-Metrix

Obfuscation is a linguistic arsenal used by those who seek to engage in deception. This was highlighted in chapter two where the literature review delineated research that uncovered the linguistic correlates of deception. However to date, there has been scant progress in using robust NLP technology to detect the resultant reduced readability in text. For the longest time and still maintain currency are readability formulas such as the Gunning Fog index shown in Eq 4.4.

$$0.4\left[\left(\frac{words}{sentences}\right)+100\ \left(\frac{complex\ words}{words}\right)\right]$$

(Eq 4.4)

The above formula would be used in the following steps:-

1. Select a passage.
2. Calculate average sentence length by dividing number of words by the number of sentences.
3. Count the “complex” words: those with three or more syllables.
4. Add the average sentence length and the percentage of complex words.
5. Multiply the results by 0.4.

	a broad	a business	a change	a charge	a combination	a common	a company	a competitive	a complete	a component	a comprehensive	a cost	a customer	a decline
f adelphia 1999.txt	2	2	0	0	0	20	0	0	0	0	0	2	2	0
f Adelphia 2000.txt	1	0	0	0	0	15	0	1	0	0	0	1	2	1
f Anicom Inc 1998.txt	1	2	0	0	0	0	1	0	1	1	0	1	1	0
f Anicom Inc 1999.txt	2	1	1	2	0	0	1	0	0	0	0	0	2	0
f Applied Microsystems 2001.txt	4	0	1	0	2	0	2	1	0	0	3	0	2	0
f Applied Microsystems 2002.txt	4	0	1	0	0	0	1	0	0	0	3	0	2	0
f Assisted Living Concepts 2008.txt	1	0	2	0	3	13	0	0	0	0	0	2	0	3
f Assisted Living Concepts 2009.txt	1	1	1	0	1	12	0	0	0	0	0	0	0	6
Bally Gaming and Systems 2004.txt	0	1	5	2	0	1	0	2	0	0	1	0	0	0
Bally Gaming and Systems 2005.txt	0	0	1	4	2	0	0	2	0	0	1	0	0	1
f Beazer Homes 2005.txt	0	1	4	0	1	0	1	0	0	1	0	0	0	0
f Beazer Homes 2006.txt	0	1	4	0	1	0	1	0	0	1	1	0	0	1
f BROOKE CORPORATION 2006.txt	0	5	0	0	1	0	3	1	0	0	0	0	0	1
f BROOKE CORPORATION 2007.txt	0	7	0	0	2	0	2	1	0	0	1	1	1	1
f Cabletron Systems Inc 2000.txt	1	0	1	1	3	0	0	2	0	1	3	1	0	1
f Cabletron Systems Inc 2001.txt	5	0	0	1	5	0	0	0	0	1	2	0	0	1
f CHINA NATURAL GAS, INC. 2008.txt	0	0	1	0	0	0	0	0	0	0	1	1	0	0
f CHINA NATURAL GAS, INC. 2010.txt	0	0	3	0	1	0	2	0	0	1	1	1	0	2
naMedia Express Holdings 2008.txt	1	152	3	0	4	0	8	3	0	0	0	0	0	0
naMedia Express Holdings 2009.txt	0	6	6	0	2	0	5	2	0	1	0	1	0	2
f Computer Associates 2000.txt	4	2	0	0	1	2	0	0	3	1	3	2	0	1

Figure 4.14: An extract of TDM for bigrams.

	a broad range	a change in	a combination of	a decline in	a decrease in	a decrease of	a increase in	a large number	a lesser extent	a loss of	a majority of	a material adverse	a material effect	a material impact
f adelphia 1999.txt	2	0	0	0	0	0	0	0	0	0	0	0	0	2
f Adelphia 2000.txt	1	0	0	1	0	0	0	0	0	0	0	0	1	2
f Anicom Inc 1998.txt	1	0	0	0	0	0	0	2	0	0	0	1	1	0
f Anicom Inc 1999.txt	2	0	0	0	0	0	1	2	0	0	0	6	1	0
f Applied Microsystems 2001.txt	2	1	2	0	1	0	0	0	2	0	0	4	0	0
f Applied Microsystems 2002.txt	1	1	0	0	1	0	0	0	2	1	0	2	0	0
f Assisted Living Concepts 2008.txt	0	1	3	3	6	0	0	2	0	0	0	8	1	1
f Assisted Living Concepts 2009.txt	0	0	1	6	12	0	1	2	1	1	0	6	2	3
f Bally Gaming and Systems 2004.txt	0	4	0	0	4	0	4	0	5	0	2	2	0	0
f Bally Gaming and Systems 2005.txt	0	1	2	1	9	2	2	1	5	0	1	4	0	0
f Beazer Homes 2005.txt	0	4	1	0	0	0	0	1	1	0	0	5	0	0
f Beazer Homes 2006.txt	0	4	1	1	1	0	3	1	1	0	4	5	0	0
f BROOKE CORPORATION 2006.txt	0	0	1	1	4	0	1	5	1	1	3	22	0	0
f BROOKE CORPORATION 2007.txt	0	0	2	1	3	0	1	5	1	1	2	23	1	0
f Cabletron Systems Inc 2000.txt	1	1	3	0	1	0	0	0	1	0	2	11	0	1
f Cabletron Systems Inc 2001.txt	4	0	5	0	1	0	0	0	0	5	3	2	0	1
f CHINA NATURAL GAS, INC. 2008.txt	0	0	0	0	0	0	0	0	0	0	0	3	0	1
f CHINA NATURAL GAS, INC. 2010.txt	0	3	1	1	1	1	0	0	0	4	0	10	0	1
f ChinaMedia Express Holdings 2008.txt	0	3	4	0	0	0	0	0	0	0	2	1	0	2
f ChinaMedia Express Holdings 2009.txt	0	3	2	2	3	0	0	7	0	0	5	17	0	3

Figure 4.15: An extract of TDM for trigrams.

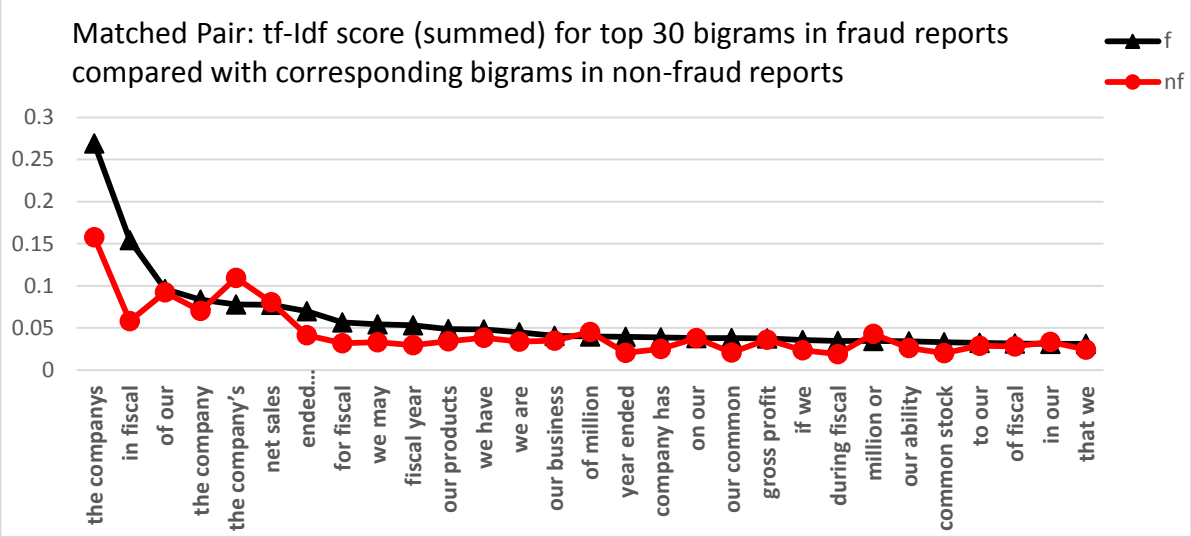


Figure 4.16: Most prominent bigrams in fraud reports compared to corresponding bigrams in non-fraud reports (matched pair set up).

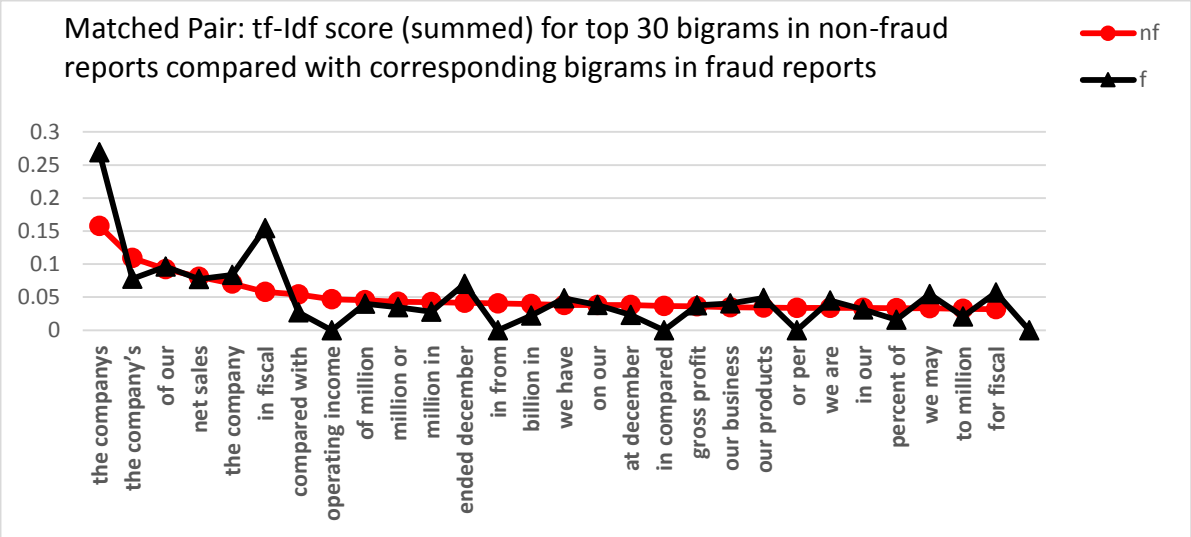


Figure 4.17: Most prominent bigrams in non-fraud reports compared to corresponding bigrams in fraud reports (matched pair set up).

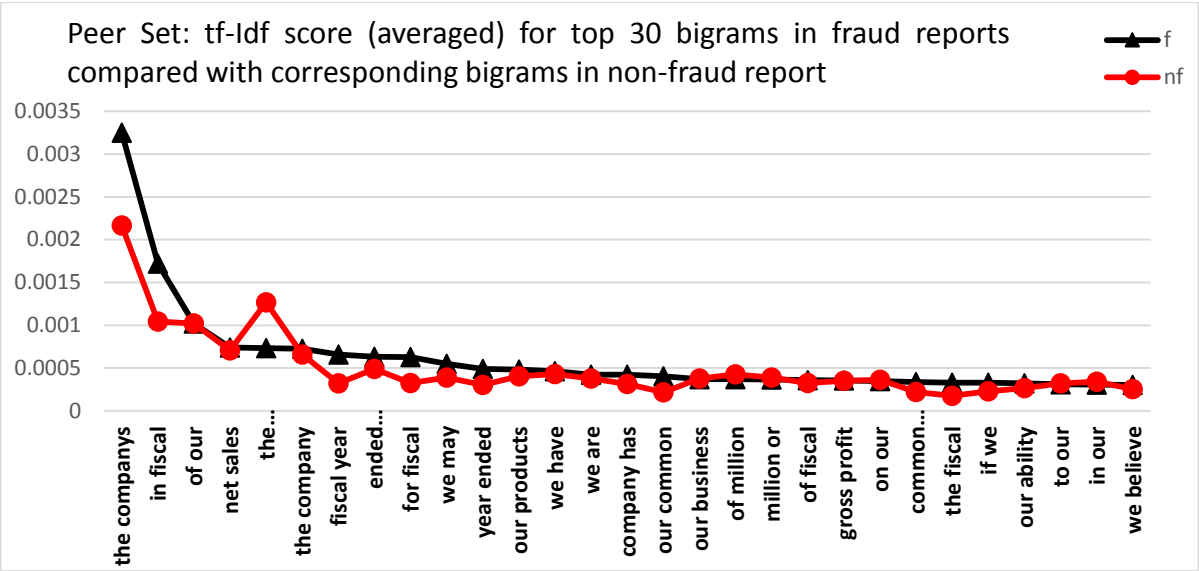


Figure 4.18: Most prominent bigrams in fraud reports compared to corresponding bigrams in non-fraud reports (peer set).

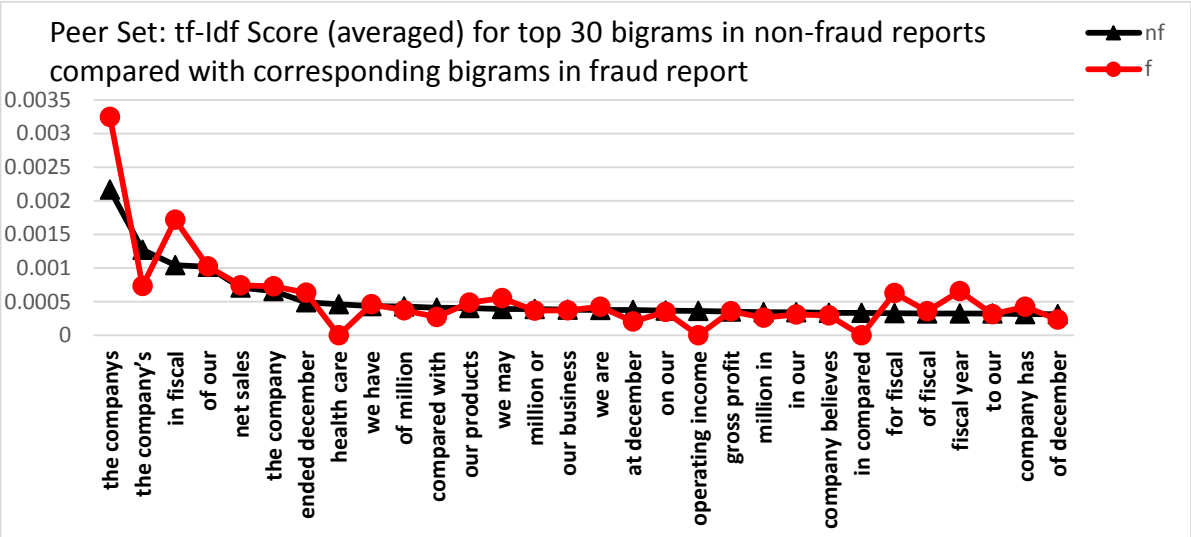


Figure 4.19: Most prominent bigrams in non-fraud reports compared to corresponding bigrams in fraud reports (peer set).

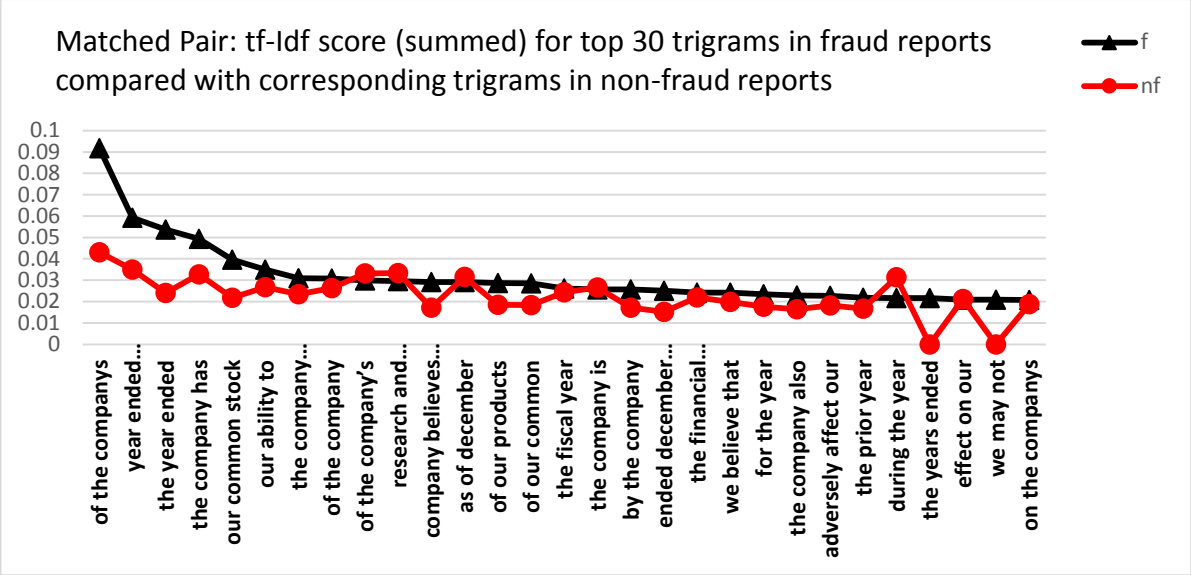


Figure 4.20: Most prominent trigrams in fraud reports compared to corresponding trigram in non-fraud reports (matched pair set up).

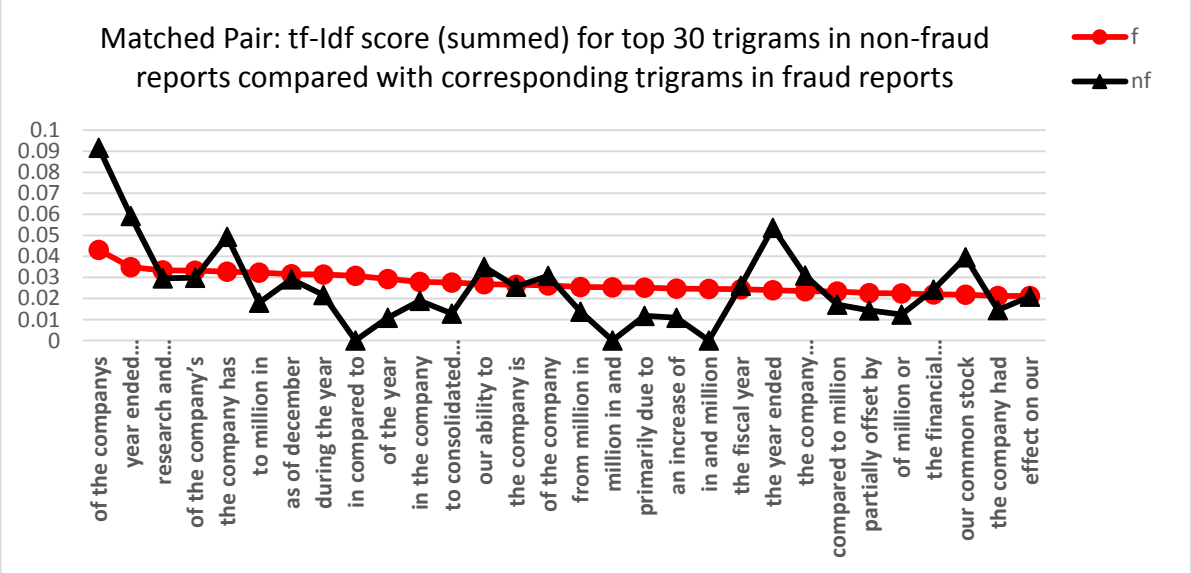


Figure 4.21: Most prominent trigrams in non-fraud reports compared to corresponding trigrams in fraud reports (matched pair set up).

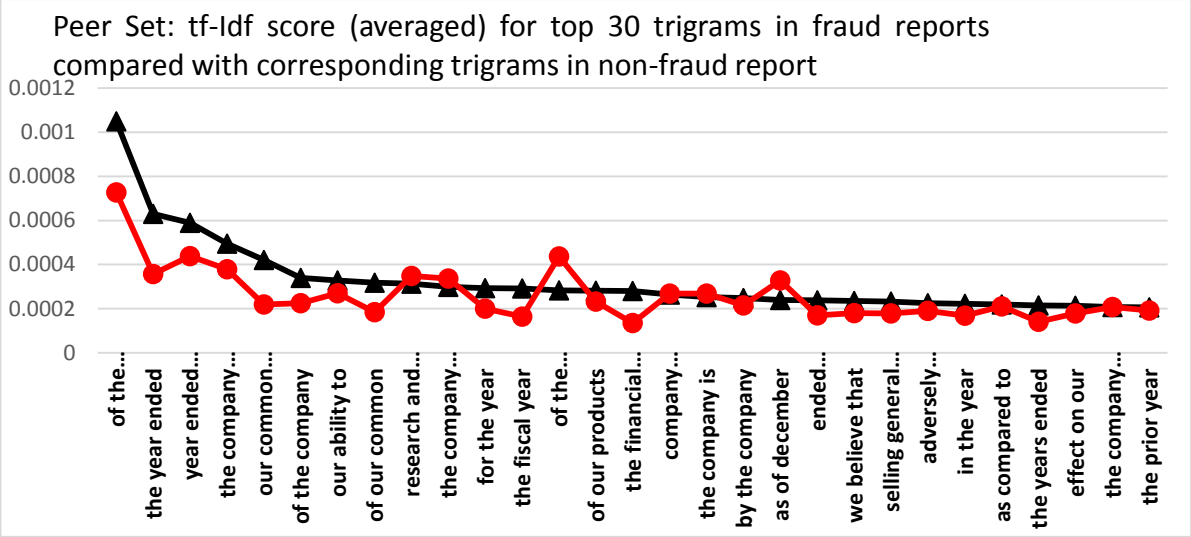


Figure 4.22: Most prominent trigrams in fraud reports compared to corresponding trigram in non-fraud reports (peer set).

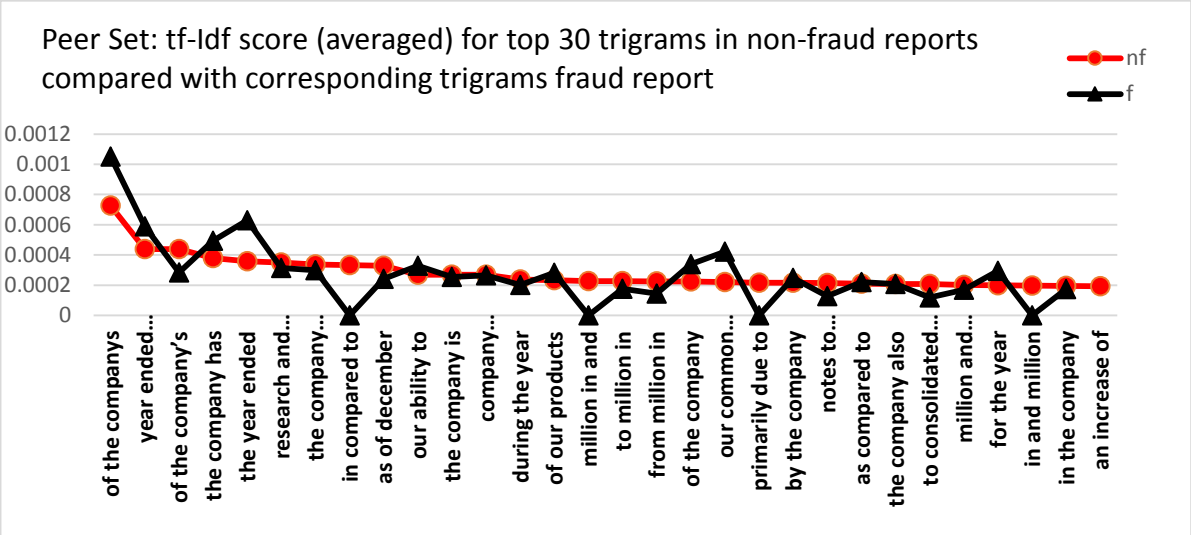


Figure 4.23: Most prominent trigrams in non-fraud reports compared to corresponding trigrams in fraud reports (peer set).

Other popular measures for readability are similarly based around the length of sentence variable.

These metrics of text complexity have been found to be highly correlated (r > .90) [50]. This correlation exists because most readability measures in use include features related to the frequency of a word in language and the length of the sentence. These measures have been discounted as they: "*ignore many language and discourse features that are theoretically influential at estimating comprehension*" [50].

Although these formulas are easy to use, limitations have also been noted. For example, Sawyer [225] argued that the early readability formulas were "*misleading and overly simplistic*". Similarly, Coupland (cited in Sheehan [226]) noted that: "*the simplicity of … readability formulas … does not seem compatible with the extreme complexity of what is being assessed*". Holland (cited in Sheehan [226]) reported a similar conclusion: "*While sentence length and word frequency do contribute to the difficulty of a document, a number of equally important variables elude and sometimes run at cross purposes to the formulas …"*. Further, significantly Loughran and Macdonald [126, 143] have totally discounted the use of traditional readability measures in financial disclosures. They argue that some longer words are commonplace business terms, thus these metrics fail to give a true picture on the readability of the text. They question the robustness of the readability studies conducted using these metrics in this domain [79, 227-229] They argue that: "*Fog and Flesch indicate that an increase in the average number of syllables decreases readability, with this factor accounting for half of each measure's inputs. However, business text commonly contains multi-syllable words used to describe operations. Words like corporation, company, directors, and executive are multi-syllable, yet are presumably easy to comprehend for anyone we consider as "average" investors. One of the longest words occurring with reasonable frequency in 10-Ks is telecommunications, which is not likely to turn most readers to their dictionaries*". Instead they find that in the financial reporting domain, readability is better defined: *"as the ability of individual investors and analysts to assimilate valuation relevant information from a financial disclosure*" [143]. They endorse the view that readability measures need to be more multidimensional that should measure how well an investor is able to: "*assimilate valuation relevant information*" [143].

Thus in this the first study where Coh-Metrix is executed over the corpus to extract indices that asses the text for readability at a deeper level. Readability measures as

those mentioned above have now been extended to include more sophisticated indices. Coh-Metrix is a tool [50] that includes these indices which give measures on cohesion relations, world knowledge, together with language and discourse characteristics. The 110 indices produced by Coh-Metrix lay a heavy emphasis on cohesiveness and cohesion in text. This is the: "*linguistic glue that holds together the events and concepts conveyed within a text*" [50]. Further, these indices give a score for measures such as referential and semantic overlap of adjacent sentences, number of connectives and a word concreteness score (words that are easy/difficult to process). Cohesive cues enable the reader to make connections between sentences and paragraphs. This is measured for example by calculating overlapping verbs and connectives (causal, intentional, temporal). Other indices measure aspects of text such as referential overlap, latent semantic similarity, narrativity (the degree to which a text tells a story with characters, events, places and things that are familiar to a reader). McNamara et al [50] give a full explanation of these indices and how they are calculated. The tools used to calculate the indices include: "*lexicons, syntactic parsers, part of speech classifiers, semantic analysis, and other advanced tools in NLP*" [50]. Coh-Metrix and the indices produced (shown in Appendix B) were delineated in chapter two. Natural language Processing techniques such as syntactic parser, latent semantic analysis have been used to develop this tool. Its validity has been strengthened through its growing usage in a number of domains [50].

As indicated in chapter 2, management obfuscate bad news through reading ease or rhetorical manipulation. The motivation is that managers make the text less clear so that information is more costly to extract and poor performance will not be reflected immediately in market prices. Similarly, the use of rhetorical language deployed through the use of pronouns, passive voice, metaphor has been used to conceal poor firm performance. Merk-Davies and Brennan [70] argue that it is not: "*what firms say" but rather "how they say it*" that leads to obfuscation. This is known as Management Obfuscation Hypothesis. The recent study by Lo et al. [230] confirm that manipulating readability is a baton used by those engaged in wilful falsification to conceal poor performance. It is therefore appropriate to examine how readability is affected in text using the corpus under study through a more rigorous tool like Coh-Metrix.

Figure 4.24 depicts the process, the whole corpus of annual reports/10-K is first thoroughly cleaned. This involved removal of all figures, tables and formatting. It also includes removing all numbers and punctuation. As McNamara et al. [50] describe the

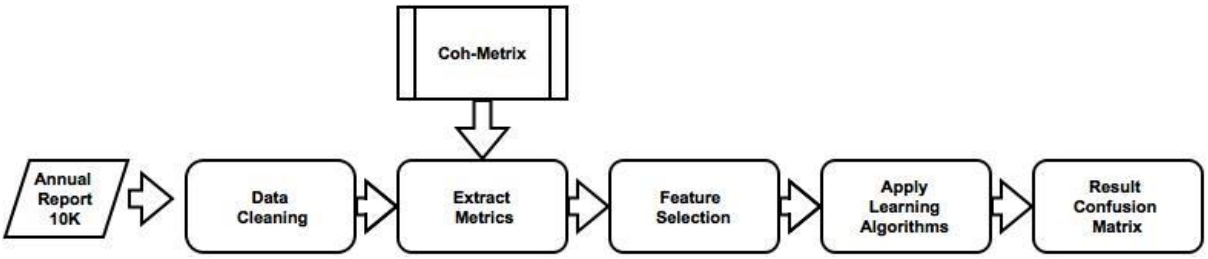


Figure 4.24: Coh-Metrix Indices extracted from corpus, then passed to downstream processes.

	DESPC	DESSC	DESWC	DESPL	DESPLd	DESSL	DESSLd	DESWLsy	DESWLsyd	DESWLlt	DESWLltd	PCNARz	PCNARp	PCSYNz	PCSYNp
adelphia 1999.txt	329	952	22829	2.89	2.18	24.35	15.32	1.90	1.12	5.59	3.14	-1.50	6.68	0.45	67.00
Adelphia 2000.txt	324	850	20589	2.62	2.21	24.61	15.95	1.91	1.11	5.61	3.12	-1.51	6.55	0.45	67.36
Anicom Inc 1998.txt	93	276	7098	2.97	1.85	26.29	15.08	1.91	1.11	5.64	3.10	-1.39	8.23	0.02	50.80
Anicom Inc 1999.txt	53	293	6986	5.53	6.69	24.32	12.44	1.87	1.08	5.53	3.02	-1.22	11.31	0.23	58.71
Applied Microsystems 2001	144	418	10545	3.07	2.19	25.14	14.23	1.94	1.14	5.71	3.23	-1.33	9.56	-0.15	44.38
Applied Microsystems 2002	144	418	10545	3.05	2.18	25.40	14.60	1.94	1.15	5.72	3.23	-1.35	9.13	-0.15	44.31
Assisted Living Concepts 2008.txt	691	1291	26548	1.87	1.63	21.02	14.11	1.89	1.09	5.44	3.02	-1.12	13.14	0.59	71.90
Assisted Living Concepts 2009.txt	622	1210	25083	1.95	1.68	21.19	14.03	1.88	1.09	5.42	3.03	-1.04	14.92	0.53	70.19
Bally Gaming and Systems 2004.txt	358	878	22830	2.45	1.78	26.38	17.79	1.87	1.09	5.36	3.03	-1.27	10.20	0.28	60.64
Bally Gaming and Systems 2005.txt	401	807	18054	2.01	1.54	22.88	13.97	1.84	1.05	5.36	2.95	-1.13	12.92	0.30	61.79
Beazer Homes 2005.txt	105	515	12403	4.91	5.21	24.52	14.57	1.87	1.10	5.48	3.10	-0.86	19.49	0.15	55.57
Beazer Homes 2006.txt	214	633	15272	2.96	2.30	24.61	13.47	1.86	1.08	5.45	3.05	-0.96	17.11	0.09	53.59
BROOKE CORPORATION 2006.txt	417	1215	31982	2.91	2.00	26.79	17.43	1.84	1.08	5.49	3.07	-1.13	12.92	0.16	55.96
BROOKE CORPORATION 2007.txt	558	1574	39047	2.82	2.00	25.29	13.83	1.87	1.11	5.55	3.11	-1.30	9.68	0.31	62.17
Cabletron Systems Inc 2000.txt	215	778	20448	3.62	3.27	26.93	15.61	1.88	1.12	5.56	3.16	-1.34	9.18	0.15	55.96
Cabletron Systems Inc 2001.txt	316	916	23398	2.90	2.17	26.14	14.11	1.90	1.11	5.58	3.15	-1.12	13.14	0.02	50.40
CHINA NATURAL GAS 2008	232	580	14025	2.50	1.91	24.69	14.85	1.79	1.02	5.26	3.03	-0.98	16.35	0.16	56.36
CHINA NATURAL GAS 2010	424	1138	31586	2.68	2.25	28.35	17.12	1.74	0.99	5.13	2.96	-0.76	22.36	-0.06	48.01
ChinaMedia Express Holdings 2008.txt	187	679	20415	3.63	3.94	30.29	18.06	1.80	1.04	5.26	3.08	-0.37	35.94	-0.23	40.90
ChinaMedia Express Holdings 2009.txt	573	1604	43888	2.80	2.22	27.87	17.35	1.82	1.06	5.37	3.09	-1.20	11.70	0.14	55.17
Computer Associates 2000.txt	111	408	9869	3.68	2.92	24.62	22.31	1.91	1.12	5.61	3.15	-1.59	5.71	0.20	57.93
Computer Associates 2001.txt	208	543	12599	2.61	1.81	23.60	13.32	1.91	1.11	5.62	3.11	-1.50	6.68	0.22	58.71
Computer Sciences Corporation 2009.txt	248	836	20929	3.37	2.31	25.62	13.53	1.85	1.07	5.42	3.04	-1.42	7.78	0.25	59.87
Computer Sciences Corporation 2010.txt	239	791	19261	3.31	2.57	24.78	13.27	1.84	1.06	5.41	3.03	-1.39	8.23	0.30	61.79
Diamond Foods 2010.txt	259	617	13271	2.38	2.13	21.84	14.18	1.83	1.05	5.39	2.98	-0.99	16.11	0.33	62.55
Diamond Foods 2011.txt	217	567	12673	2.61	2.07	22.69	13.32	1.83	1.06	5.37	3.00	-0.93	17.62	0.26	59.87
enrc 2008.txt	390	945	19716	2.42	2.25	21.38	12.54	1.74	1.03	5.20	2.93	-1.24	10.75	0.25	59.48

Figure 4.25: The matrix setup with Coh-Metrix indices extracted.

	Coh-Metrix
Matched Pair	204 rows (102 'f' reports) and (102 'nf' reports) by 110 indices
Peer Set	408 rows (102 'f' reports) and (308 'nf' reports) by 110 indices

Table 4.3: The dimension of the matrices set up for Coh-Metrix Indices.

text had to look as if: "*the writer had just finished typing it, had it checked for typos and errors by a large group of copy editors, printed if off, and handed it over to the reader*". Coh-Metrix is then executed over the reports in the corpus. It outputs 110 indices with corresponding values for each of the 408 reports. An extract of the matrix is shown in Figure 4.25.

This matrix is then loaded into R. Two matrices are set up for matched pair and peer set, the dimensions are shown in Table 4.3. The documents are then represented by these indices that constitutes the features in the classifier models. Both the peer set and matched pair matrices are put through feature selection as will be described in Chapter 5. The features (Coh-Metrix) indices chosen are then ready for classification. Appendix H, Table H.1 shows 16 of the Coh-Metrix indices plotted for fraud and non-fraud reports. Each point represents the value of the Coh-Metrix index being measured for a firm report. It can be seen that although the fraud and non-fraud reports are not clearly separable there is enough of a difference visible that that could be used to aid in the classification task.

4.6 Linguistic Inquiry and Word Count (LIWC) 2015

Linguistic Inquiry and Word Count (LIWC) [231] program employs a simple yet intuitive way to measure language use in a variety of settings. LIWC reads written text in files such as text files. Its text analysis module compares each word in the text against a user defined dictionary. Once the processing module has read and accounted for all words in a given text, it calculates the percentage of total words that match each of the dictionary categories. If LIWC analysed an annual report of 3000 words, it might find that there were 300 pronouns and 125 positive emotion words used (as picked up by appropriate dictionary). It would convert these to percentages, (300/3000 * 100) = 10% pronouns and (125/3000 * 100) = 4% positive emotion words. Words contained in the text are read and analysed by LIWC 2015 are referred to as target words. Words in the LIWC 2015 dictionary file are referred to as dictionary words.

The main engine that attempts to bring out the meaning in text in an automated manner are a group of dictionaries that tell the text analysis module which words to identify and classify. LIWC 2015 comes with three internal dictionary systems: the LIWC 2015 dictionary and the previous LIWC 2007 and LIWC 2001. The new LIWC 2015 master

dictionary is composed of almost 6,400 words, word stems, and selected emoticons. For each dictionary word, there is a corresponding dictionary entry that defines one or more word categories: *"For example, the word cried is part of five word categories: Sadness, Negative Emotion, Overall Affect, Verb, and Past Focus. Hence, if the word cried was found in the target text, each of these five sub-dictionary scale scores would be incremented. As in this example, many of the LIWC 2015 categories are arranged hierarchically. All sadness words, by definition, will be categorized as negative emotion and overall affect words"* [231].

Dictionaries were established by judges determining the "*goodness of fit*" [231] for each category. In cases of disputes several corpora and online sources were referenced to determine word's common use, inflection and meaning. Once a working version of the dictionary was constructed texts from several sources were analysed to determine how frequently dictionary words were used in various contexts such as Facebook, blog posts etc.

To uncover deception Newman et al. [127] used it to calculate the percentages of specific linguistic cues in true versus deceptive statements, yielding above-chance accuracy of classifications for different types of lies. Subsequently, researchers from a variety of fields have also applied LIWC with the same purpose (Hauch et al. [133] provide a comprehensive list). However, all of the research work cited is based on the older LIWC 2007, this is the first deception based research study that uses LIWC (2015). LIWC (2015) has been updated with new dictionaries to pick up new categories such as analytical thinking (examining formal, logical vs informal, personal), emotional tone (high = positive tone vs low = sadness, anxiety, hostility), clout (words that denote perspective of high expertise versus tentative or humble style).

There exists substantial evidence that indicates how our choice of words can reveal our inner intentions [123, 127, 132] : "*a great deal can be learnt about people's underlying thoughts, emotions, and motives by counting and categorizing the words they use to communicate*" [127]. Newman et al. [127] examined a number of narratives from a variety of sources and concluded that the language we use is like a '*fingerprint'* thus enabling identification of the true meaning behind the words we deploy. From this premise, deception detection research has derived linguistic cues to be found in written text that can aid in separating liars from truth-tellers. Some of these cues are outlined in Table 2.1 where column 2 illustrates how these cues could be manifested in text with column 3 and 4 giving reference to the authors and the underlying theories

respectively. Zhou [123] formalised these cues into nine constructs that have been used to automate deception detection with successful outcomes [103, 132]. These will be discussed in section 4.8.

Tausczik and Pennebaker [232] cite a number of reasons that give weight to using LIWC 2015 to take a closer look at language use.

1. Style words, content words, personal pronouns - reflect how people are communicating (style words) and what they are saying (content words), provide information about the subject of attention (personal pronouns). Analyses of the tense of common verbs can tell about the temporal focus of attention. Pronouns and verb tense are useful linguistic elements that can help identify focus, which in turn can show priorities, intentions, and processing. LIWC 2015 provides counts for each of these word categories.

2. Research suggests that LIWC accurately identifies emotion in language use. For example, positive emotion words (for example, '*love', 'nice', 'sweet'*) are used in writing about a positive event, and more negative emotion words (for *example 'hurt', 'ugly', 'nasty'*) are used in writing about a negative event [232].

3. Aspects of status, dominance and hierarchy can also filter into language. Studies conducted [232] confirm increased use of first-person plural as a good predictor of higher status and first-person singular was a good predictor of lower status. Further LIWC has dictionaries that contain words related to power, reward, risk, affiliation, achievement that enable identification of corresponding words in the text.

4. Exclusive words (for example *'but'*, *'without'*, *'exclude'*) are helpful in making distinctions. Indeed, people use exclusion words when they are attempting to make a distinction between what is in a category and what is not in a category. Exclusive words are used at higher rates among people telling the truth [127]. Conjunctions (for example '*and', 'also', 'although'*) join multiple thoughts together and are important for creating a coherent narrative [150]. Prepositions (for example *'to', 'with', 'above'*), cognitive mechanisms (for example *'cause', 'know', 'ought*'), and words greater than six letters are all also indicative of more complex language. Prepositions signal that the speaker is providing more complex and often, concrete information about a topic [232].

5. The use of causal words (for example '*because', 'effect', 'hence'*) and insight words (for example '*think', 'know', 'consider'*), two subcategories of cognitive mechanisms, in describing a past event can suggest the active process of reappraisal [232].

6. The language that people use to discuss an event can reveal something about the extent to which a story may have been established or is still being formed. When people are uncertain or insecure about their topic, they use tentative language (for example *'may be', 'perhaps', 'guess'*) and more filler words (for example '*blah', 'I mean', 'you know'*). Participants who recounted an event that they had already disclosed to someone else used fewer words from the tentative category than participants who recounted an undisclosed event [232]. Possibly, higher use of tentative words suggests that a participant has not yet processed an event and formed it into a story.

7. Deceptive statements compared with truthful ones are moderately descriptive, distanced from self, and more negative. Research from an experimental perspective that examined language for differences between liars and truth-tellers was expounded in chapter 2.

When using LIWC over reports in the corpus the following was produced:-

For each report LIWC outputs 90 variables:-

- a file name and word count
- 4 summary language variables (analytical thinking, clout, authenticity, and emotional tone)
- 3 general descriptor categories (words per sentence, percent of target words captured by the dictionary, and percent of words in the text that are longer than six letters)
- 21 standard linguistic dimensions (eg percentage of words in the text that are pronouns, articles, auxiliary verbs, etc.)
- 41 word categories tapping psychological constructs (e.g., affect, cognition, biological processes, drives)
- 6 personal concern categories (eg., work, home, leisure activities)
- 5 informal language markers (assents, fillers, swear words, 'netspeak') – not included in this study as irrelevant.
- 12 punctuation categories (periods, commas, etc) been added by the judges.

An example of an output file for an annual report, given by LIWC is shown in Appendix H, Figure H.1 and a description of the main variables used shown in Table H.2. Four hundred and eight files are obtained by running LIWC over each annual reports/10-K in the corpus. The output for each file is then added to a matrix, with an added

Filename	WC	Analytic	Clout	Authentic	Tone	WPS	Sixltr	Dic	function	pronoun	ppron	ipron	article	prep	auxverb	adverb	conj
f adelphia 1999.txt	22992	98.10	54.03	21.61	60.15	26.13	37.71	71.16	36.30	3.12	0.33	2.79	7.85	15.20	3.65	1.67	5.60
f Adelphia 2000.txt	20745	98.13	53.97	20.93	55.00	27.70	38.17	71.89	36.51	3.24	0.41	2.82	7.72	15.58	3.59	1.76	5.54
f Anicom Inc 1998.txt	7088	98.09	57.45	24.72	58.87	26.35	38.94	75.65	37.18	3.68	0.21	3.47	7.39	16.08	3.47	0.97	6.22
f Anicom Inc 1999.txt	6989	97.31	64.68	23.09	54.60	26.88	37.19	76.69	38.40	5.21	1.67	3.53	6.81	16.33	3.56	1.10	6.10
f Applied Microsystems 2001.txt	10498	97.60	55.07	17.30	37.92	27.41	41.63	76.22	38.79	3.30	0.45	2.85	7.62	16.16	4.17	1.81	6.84
f Applied Microsystems 2002.txt	14294	94.78	77.21	13.61	53.10	30.09	36.73	75.46	42.16	6.97	4.60	2.37	6.43	16.59	4.86	1.60	6.54
f Assisted Living Concepts 2009.txt	25272	96.95	71.53	20.32	52.81	28.11	35.99	77.21	39.56	5.71	3.51	2.20	6.15	17.08	3.74	1.15	6.33
f Assisted Living Concepts 2008.txt	26746	96.98	71.91	18.75	53.31	28.30	36.60	77.32	39.58	5.77	3.56	2.21	6.02	17.21	3.68	1.13	6.31
f Bally Gaming and Systems 2004.txt	22992	97.11	64.43	13.58	70.31	29.07	34.18	76.24	40.77	4.61	2.42	2.19	8.39	15.65	4.44	1.68	6.72
f Bally Gaming and Systems 2005.txt	18230	96.09	75.76	22.11	61.45	27.21	34.02	76.85	39.85	6.18	3.86	2.32	7.05	15.70	3.85	1.66	6.36
f Beazer Homes 2005.txt	12543	94.86	80.73	23.53	58.19	28.57	35.33	78.47	41.73	7.51	5.31	2.20	5.83	16.75	4.29	1.63	6.14
f Beazer Homes 2006.txt	15412	95.59	79.09	23.52	56.53	26.85	35.23	77.02	40.78	6.87	4.76	2.11	5.79	16.87	3.97	1.60	6.22
f Broadcom 2006.txt	37364	90.72	78.18	20.32	50.28	29.49	37.21	75.26	40.24	7.43	4.92	2.51	5.27	14.54	4.26	1.56	7.92
f Broadcom 2007.txt	42131	91.28	77.90	20.69	50.92	30.24	36.86	75.47	40.23	7.37	4.87	2.50	5.38	14.69	4.13	1.56	7.88
f BROOKE CORPORATION 2006.txt	32506	95.33	72.69	13.59	73.51	27.85	34.89	79.39	39.80	6.15	3.46	2.69	5.97	16.13	4.07	1.87	6.47
f BROOKE CORPORATION 2007.txt	39741	96.18	63.79	12.87	58.81	27.02	36.60	77.50	39.02	5.07	2.27	2.80	6.43	15.99	4.21	1.83	6.43
f Cabletron Systems Inc 2000.txt	20697	98.39	53.61	20.23	55.08	27.05	37.96	75.22	37.97	3.00	0.21	2.79	8.63	15.71	4.13	1.17	5.75
f Cabletron Systems Inc 2001.txt	23451	97.15	66.50	19.88	51.97	26.41	38.45	76.09	39.45	5.25	2.46	2.79	7.60	15.74	4.06	1.04	6.27
f CHINA NATURAL GAS, INC. 2008.txt	14191	96.39	73.73	14.55	41.44	26.28	33.56	73.40	39.58	5.67	3.54	2.12	7.12	15.76	4.40	1.35	6.15
f CHINA NATURAL GAS, INC. 2010.txt	31995	95.43	73.14	21.36	41.90	30.44	31.16	73.50	41.26	6.40	4.33	2.07	6.56	16.29	4.81	1.34	6.49
f ChinaMedia Express Holdings 2008.t	20492	91.74	80.60	12.50	81.78	33.70	32.69	80.93	47.71	8.60	5.66	2.94	8.19	15.81	6.94	1.89	6.28
f ChinaMedia Express Holdings 2009.t	44462	97.61	48.32	22.35	48.74	30.75	34.51	73.40	39.21	4.30	0.53	3.77	7.13	16.67	4.14	1.45	6.09
f Computer Associates 2000.txt	9852	98.39	53.00	21.08	60.21	25.07	39.71	74.15	37.30	2.83	0.26	2.57	8.54	15.60	3.81	1.66	5.49
f Computer Associates 2001.txt	12561	98.48	53.38	21.30	61.55	25.43	40.32	74.99	37.79	2.85	0.31	2.54	9.10	15.51	3.81	1.59	5.82
f Computer Sciences Corporation 2009	21090	98.16	61.54	19.13	58.88	24.99	36.71	75.81	38.99	3.66	1.50	2.15	8.17	16.53	3.92	1.48	6.38
f Computer Sciences Corporation 2010	19415	97.67	61.88	17.18	59.83	25.21	36.55	75.29	39.02	3.94	1.75	2.19	7.62	16.35	4.17	1.48	6.52

Figure 4.26: The matrix setup with LIWC variables extracted.

	LIWC
Matched Pair	204 rows (102 ‘f’ reports) and (102 ‘nf’ reports) by 35 variables
Peer Set	408 rows (102 ‘f’ reports) and (308 ‘nf’ reports) by 35 variables

Table 4.4: Dimensions of matrices for LIWC extracted features.

categorical variable ‘f’ for fraud reports or ‘nf’ for non-fraud. An extract of the final matrix is 408 rows long with 73 columns is shown in Figure 4.26. In order to cut down on redundancy only the main groupings of words will be passed to the classifiers in a matrix. This constitutes 35 LIWC variables For example total function words, total pronouns, affective processes, cognitive processes, perceptual processes, drives, time orientations and relativity. These groupings give the sum value for their respective elements for example personal pronouns made up of first person singular (i, me, mine), first person plural (we, us, our) and impersonal pronouns. Affective processes are broken down into positive, negative emotions. For cognitive processes specific counts on words that relate to insight, causation, discrepancy, tentative, certainty and differentiation are added to a matrix as opposed to the overall counts. Again this matrix first person plural (we, us, our) and impersonal pronouns. Affective processes are broken down into positive, negative emotions. For cognitive processes specific counts on words that relate to insight, causation, discrepancy, tentative, certainty and differentiation are added to a matrix as opposed to the overall counts. Again this matrix is constructed for both peer set and matched pair design (as shown in Table 4.4).

A few variables extracted from LIWC are graphed for both fraud and non-fraud reports and shown in Appendix H, Table H.3. Although not clearly separable there is enough separation between the fraud and non-fraud cases that a classifier could attempt to distinguish as will be shown in chapter 6.

4.7 Custom Dictionaries

CFIE-FRSE-Web tool (https:/cfie.lancaster.ac.uk:8443) [184] is used to upload financial reports and word lists. Thus enabling counts to be taken of the number of times words in word list appear in financial reports. Five custom dictionaries or word lists were loaded into CFIE-FRSE-Web tool [184] and counts taken. The motivation for using these word lists will now be expanded.

Loughran and Mcdonald [126, 143] maintain that given evidence based on past research word classifications can be an effective way in measuring tone in financial documents. Word classifications that denote positive, negative sentiments in text have been taken from the Harvard Psychosociological Dictionary, Diction and others.

However, they argue that English words have many meanings and a word categorisation scheme for one discipline is not always appropriate for another discipline. For example, the word *'volatility'* is *"the trait of being unpredictably irresolute"*. However, in the financial world, volatility is an investment's ability to go through changes in value over time. Loughran and Mcdonald [126, 143] provide evidence based on 10-Ks between 1994 and 2008 that word list such as the H4N substantially misclassify words when gauging the tone in financial documents. Misclassified words include common words used in such documents for example *'taxes', 'liabilities'*. They find that almost three-fourths of the negative words according to the Harvard list are attributable to words that are not negative in a financial context: *"Words such as tax, cost, capital, board, liability, foreign, and vice are on the Harvard list. These words also appear with great frequency in the vast majority of 10-Ks, yet often do no more than name a board of directors or a company's vice-presidents. Other words on the Harvard list, such as mine, cancer, crude (oil), tire, or capital, are more likely to identify a specific industry segment than reveal a negative financial event"*

Loughran and Mcdonald [126] develop alternative word lists that are more suitable for gauging sentiment in the financial reporting domain. To create these word lists they

develop a dictionary of words and word counts from all 10-Ks filed during 1994 to 2008. They examine all words occurring in at least 5% of the documents, to consider their most likely usage in financial documents. The word lists are the following:-

- Negative word list

This consists of 2337 words that typically have negative implications in a financial sense. Frequently occurring words in the list that are not on the H4N list include: '*restated', 'litigation', 'termination', 'discontinued', 'penalties', 'unpaid', 'investigation', 'misstatement', 'misconduct', 'forfeiture', 'serious', 'allegedly', 'noncompliance', 'deterioration'* and '*felony'*.

- Positive word list

This consists of 353 words including inflections and are substantially fewer words than in the negative word list. Loughran and Mcdonald [126] find that there are very few words that can be clearly designated as positive as writers of annual reports tend to avoid negative language instead qualifying positive words. Words in this list include '*achieve', 'attain', 'efficient', 'improve', 'profitable', or 'upturn'* that the authors maintain could also be more unilateral in tone.

- Uncertainty word list

This consists of words denoting uncertainty with emphasis on the general notion of imprecision rather than exclusively focusing on risk. The list includes 285 words such as *approximate, contingency, depend, fluctuate, indefinite, uncertain*, and *variability.*

- Litigious word list

This list categorizes words reflecting a propensity for legal contest. The list includes 731 words such as *'claimant', 'deposition', 'interlocutory', 'testimony', and 'tort'.* Loughran and Mcdonald [126] also include words like legislation and regulation, which do not necessarily imply a legal contest but may reflect a more litigious environment.

- Strong modal and weak modal:

Loughran and Mcdonald [126] extend Jordan's [234] categories of strong and weak modal words to include other terms expressing levels of confidence. Examples of strong modal words are words such as *'always', 'highest', 'must'*, and *'will'*. Examples of weak modal words are *'could', 'depending', 'might', and 'possibly'*. There are 19 strong words in our list and 27 weak words.

In creating these word lists, Loughran and Mcdonald [126] include all variants of a word eg *'high', 'higher', 'highest'*. They argue that the text processing literature shows

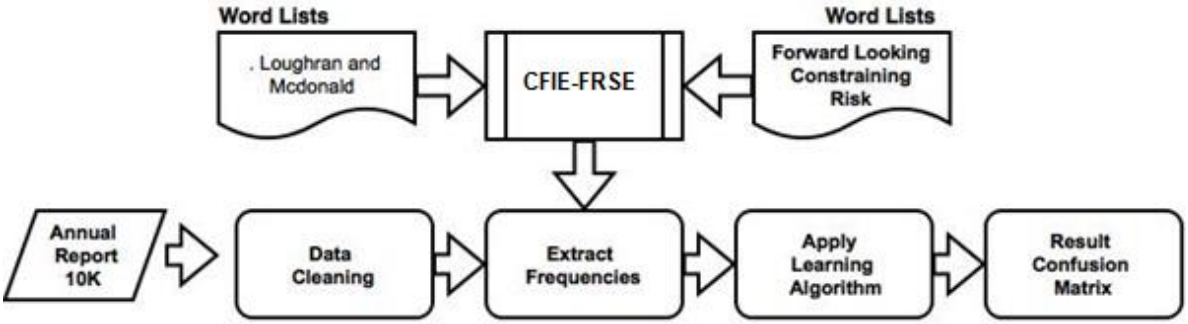


Figure 4.27: Extract counts of words in word lists found in reports using the CFIE-FRSE tool [184].

that stemming does not in general improve performance. They find evidence that some word lists are related to market reactions around the 10-K filing date, trading volume, unexpected earnings, and subsequent stock return volatility. In particular, they find that their negative word list is significantly related to announcement returns. In sum, they conclude that textual analysis through the use of the above domain appropriate word lists can contribute to understanding the impact of information on stock returns. Therefore given the appropriateness of these word lists for the financial domain and their predictive ability in the area of stock returns and trading volumes, these five word list are used to aid in discriminating a fraud from a non-fraud firm in this study. These word lists are loaded into CFIE-FRSE-Web tool and counts taken for the number of times words in a word list (litigious, uncertainty, strong modal and weak modal, negative, positive) are found in the corpus. Three further word lists were also used to determine if they aid in the discriminating task.

Bonsall et al. [235] argue that sentiment analysis or determining tone has taken a centre stage into studies into content analysis of financial narratives. They maintain that there has been little focus into the '*informativeness*' of such reports by examining its temporal (forward-looking) components. Similarly El-Haj et al. [31] count up the number of forward looking words in a corpus of UK Preliminary Earning Announcements (PEAs) to determine attribution bias. Such counts are then passed to machine learning classifiers to check for correct designation of attribution in the documents. Using a sample of quarterly earnings announcements from 2004-2014, Bozanic et al. [236] also find strong evidence that forward-looking disclosures represent informative disclosures.

Previous to these studies there have been landmark studies done by Beattie et al. [84] which outlined a methodology for examining narrative sections of annual reports. Examination of temporal components featured strongly in this methodology. In order to test out whether temporal components can aid in the discrimination task, a forward looking word list is used. This word list was developed by El- Haj et al. [184] as part of a Corporate Financial Information Environment (CFIE) project. They use this word list to quantify forward looking words in UK annual reports.

Following the near collapse of the financial system after 2008, there have been attempts made to monitor financial stability. Again word lists have been deployed to measure either excitement about gain or anxiety about loss [237]. For example, word list that capture the near bankruptcy situation of a firm would be revealing. According to Rezzae [2] and as shown in Chapter two, firms that are subsequently caught for FSF have been under pressure as a result of for example poor operating profits and/or cash flow issues. Bodnaruk et al. [238] construct a word list to aid in identifying firms that face liquidity issues or are financially constrained. This list consists of 184 words and like Loughran and McDonald [126] examine tens of thousands of words that appear in at least 5% of all 10-K filings. Commonly used constraining words from the list include: '*required', 'obligations', 'requirements', 'permitted', 'comply', and 'imposed'.* They maintain that managers anticipating financial challenges will use a more constraining tone in 10-K filings to communicate their concerns to shareholders, thereby lowering their exposure to subsequent litigation. Bodnaruk et al. [238] use this word list to construct a measure of financial constraints and use this measure to predict subsequent events associated with either deterioration or improvement of external financing conditions, events which we label *"liquidity events"*. These events are dividend omissions, dividend increases, equity recycling (paying out equity proceeds to shareholders in the form of share buybacks and dividends) and underfunded pension plans. The authors find that these word lists do help in identifying such liquidity events. They find that a more frequent usage of constraining words is strongly related to a higher likelihood of future dividend omission (+10.32%), increases (-6.46%), equity recycling (-23.24%), and underfunded pensions (+2.34%). Simply, the authors have added textual analysis as another tool to be used with the traditional mix of numerical variables that are traditionally used to gauge the level of financial constraints.

Fraud/Non Fraud Firms	Word_Count	positivity_Freq	Uncert1_Freq	negativity_Freq	orwardLooking_Fre	Litiguous	Modal Strong	Modal Weak	Constraining	Risk	clas
f adelphia 1999.txt	22491	2.27	1.36	1.13	2.29	1.18	0.35	0.40	0.91	0.67	f
f Adelphia 2000.txt	20313	2.27	1.34	1.24	2.30	0.91	0.31	0.40	0.92	0.68	f
f Anicom Inc 1998.txt	6993	3.70	1.36	1.46	2.59	0.14	0.24	0.19	0.35	0.26	f
f Anicom Inc 1999.txt	6911	3.57	1.59	1.79	3.83	0.23	0.43	0.57	0.44	0.28	f
f Applied Microsystems 2001	10313	2.12	2.47	3.23	3.99	0.33	0.26	0.99	0.50	0.20	f
f Applied Microsystems 2002	14053	1.91	1.92	3.20	4.28	0.56	0.25	0.74	0.58	0.30	f
f Assisted Living Concepts 2008.txt	26681	1.80	1.60	2.72	2.83	0.65	0.13	0.73	0.88	0.55	f
f Assisted Living Concepts 2009.txt	25144	1.82	1.64	2.79	2.97	0.69	0.15	0.76	0.92	0.56	f
f Bally Gaming and Systems 2004.txt	22742	2.16	1.53	1.75	3.17	0.91	0.39	0.61	0.94	0.61	f
f Bally Gaming and Systems 2005.txt	18090	2.35	1.83	2.56	3.34	0.75	0.27	0.76	0.61	0.33	f
f Beazer Homes 2005.txt	12377	2.70	2.35	2.96	4.13	1.04	0.19	0.99	1.14	0.48	f
f Beazer Homes 2006.txt	15218	2.75	2.02	2.98	3.42	0.93	0.15	0.88	1.04	0.51	f
f BROOKE CORPORATION 2006.txt	31659	2.37	1.86	2.56	3.13	0.68	0.28	0.75	0.54	0.29	f
f BROOKE CORPORATION 2007.txt	38789	2.25	1.72	2.62	2.69	0.93	0.25	0.69	0.62	0.36	f
f Cabletron Systems Inc 2000.txt	20113	2.70	1.92	2.17	4.27	0.18	0.54	0.63	0.30	0.16	f
f Cabletron Systems Inc 2001.txt	23160	2.76	1.86	2.37	4.05	0.34	0.27	0.77	0.43	0.23	f
f CHINA NATURAL GAS Inc 2008	13800	1.78	2.04	1.71	4.02	1.14	0.33	0.93	0.76	0.37	f
f CHINA NATURAL GAS Inc 2010	31116	2.06	2.00	2.63	4.67	1.34	0.35	1.20	0.84	0.42	f
f ChinaMedia Express Holdings 2008.txt	20342	1.56	2.23	2.45	8.63	0.86	1.12	1.56	0.80	0.36	f
f ChinaMedia Express Holdings 2009.txt	43384	2.12	1.54	2.10	3.82	1.28	0.30	0.94	0.70	0.45	f
f Computer Associates 2000.txt	9682	2.81	1.59	1.50	2.32	0.36	0.30	0.30	0.37	0.23	f
f Computer Associates 2001.txt	12362	2.46	1.71	1.82	3.09	0.57	0.38	0.56	0.38	0.22	f
f Computer Sciences Corporation 2009.txt	20801	2.80	2.03	2.81	3.09	1.15	0.19	0.73	0.71	0.35	f
f Computer Sciences Corporation 2010.txt	19137	2.76	2.11	3.09	3.47	1.33	0.21	0.84	0.79	0.37	f
f Diamond Foods 2010.txt	13293	2.38	2.48	2.77	4.13	0.33	0.23	1.15	0.95	0.38	f
f Diamond Foods 2011.txt	12643	2.46	2.67	2.99	4.63	0.49	0.28	1.37	1.10	0.41	f

Figure 4.28: The matrix setup from counts obtained from word lists.

	Custom Dictionaries
Matched Pair	204 rows (102 'f' reports) and (102 'nf' reports) by 9 variables
Peer Set	408 rows (102 'f' reports) and (308 'nf' reports) by 9 variables

Table 4.5: Dimensions of matrices for custom dictionaries extracted features.

In light of the above argument, it seems plausible to use such a list to aid in distinguishing a fraud from a non-fraud firm. The word list as developed by Bodnaruk et al. [238] was input into CFIE-FRSE and a count taken of words in dictionary that are also found in the reports.

Another word list with phrases deemed to denote comprehensive risk categories, as devised by Matties and Coners [239] risk based words is also used to attain counts in the corpus. They devised these keywords based on a review of the relevant regulations and risk management concepts, such as IFRS (International Financial Reporting Standards), FASB (Financial Accounting Standards Board).

The process of obtaining counts of words in word list found in the reports is depicted in Figure 4.27. The corpus was cleaned as was done for the extraction of Coh-Metrix indices. The files and word list were then loaded into the CFIE-FRSE tool. This tool simply takes counts of words in text found in word list. An extract of the matrix established is shown in Figure 4.28. Two matrices are produced with the composition shown in Table 4.5, ready to pass to feature selection and then the classifiers.

Appendix H, Table H.4 shows the counts of words and a few chosen word lists plotted. Again it can be discerned that there is separation between the fraud and non-fraud cases that could be picked up by classifiers. This will be outlined in chapter 6.

4.8 Linguistics-Based Cues (LBC)

Extensive research has been conducted to derive cues to deception to enable automated detection. This was covered in chapter two. Once identified these cues could then be designated features for classifier models that can be used as an early warning application used by auditors to alert of possible misdemeanours.

Chapter two reviewed the nature of deception and its manifestation in text. From the findings of research into deception and criminality, Zhou et al. [123] derived cues that they assert would filter through to the language used by those engaged in deception and lies. The relevant ones as identified by Humpherys et al. [103] for the financial reporting domain are delineated in Table 4.6.

In a financial reporting environment, the hypothesis is that firms that are facing losses may exclude negative news, include misleading positive statements, or create an optimistic outlook based on false premises to obfuscate the true state of the company. As was covered in chapter 2, the language that those intent on deception and lies would use more pleasant terms, more affect, more modal verbs. Verbs such as *'would', 'should', and 'could'*, lower the level to commitment.

Such language increases uncertainty, lowers personal responsibility and creates a distance between events and personal action. Consequently, those engaged in FSF would leave similar traces in the language traces in the language they use for financial documents such as the annual report/10-K. Further, according to Management Obfuscation Hypothesis, managers would increase complexity of their statements by writing longer sentences, to deflect responsibility managers would refer to groups instead of individuals and by communicating in a passive voice. As Humpherys et al. [103] argue that: "*the passive voice (e.g., 'mistakes were made by the company', and 'performance was adversely affected') allows the filing company to disassociate itself from the message by either omitting the actor or making the actor the object of a statement*".

Affect	
Affect Ratio	Number of affect words/Total number of words
Imagery Ratio	Number of imagery words/Total number of words;
Pleasantness Ratio	Number of pleasantness words.
Complexity	
Average Sentence Length	Number of words/Total number of sentences
Average Word Length	Number of syllables/Total number of words
Pausality	Number of punctuation marks/Total number of sentences
Diversity	
Content Word Diversity	Percentage of unique content words/Total number of content words
Function Word Diversity	Number of function words/Total number of sentences
Lexical Diversity:	Percentage of unique words or terms out of total words
Expressivity	
Emotiveness	Ratio of adjective and adverbs to nouns and verbs
Non-immediacy	
Group References	First person plural pronoun count/Total number of verbs
Other References	Count of all other singular or plural pronouns/Total number of verbs
Passive Verb Ratio	Number of passive verbs/Total number of verbs
Quantity	
Modifier Quantity	Total number of modifiers
Sentence Quantity	Total number of sentences
Verb Quantity	Total number of verbs
Word Quantity	Total number of words
Specificity	
Sensory Ratio	Number of words referencing the five senses/No of words
Temporal Immediate Ratio	Number of words that reference temporal or spatial information/Total number of words
Uncertainty	
Modal Verb Ratio	Number of modal verbs/Total number of verbs

Table 4.6: Zhou et al 2004 LBCs derived from the corpus.

	LBC
Matched Pair	204 rows (102 'f' reports) and (102 'nf' reports) by 20 variables
Peer Set	408 rows (102 'f' reports) and (308 'nf' reports) by 20 variables

Table 4.7: Dimension of matrices constructed using LBCs.

	AFFECT			COMPLEXITY			DIVERSITY			EXPRESSIVITY
	Affect	*Imagery*	*Pleasantness*	*Avg Sent Length*	*Avg Word Length*	*Pausality*	*Content Word*	*Function Word Diversity*	*Lexical Diversity*	*Emotiveness*
f adelphia 1999.txt	2.55	393.21	2.18	26.13	1.90	0.02	377.39	0.04	0.12	198.59
f Adelphia 2000.txt	2.47	393.27	2.01	27.70	1.91	0.02	378.87	0.04	0.13	196.78
f Anicom Inc 1998.txt	3.41	400.09	2.57	26.35	1.91	0.06	384.03	0.13	0.19	200.54
f Anicom Inc 1999.txt	4.11	396.12	2.82	26.88	1.87	0.05	379.98	0.13	0.19	206.21
f Applied Microsystems 2001	3.53	391.10	2.11	27.41	1.94	0.04	379.12	0.09	0.34	203.53
f Applied Microsystems 2002	3.83	392.97	2.64	30.09	1.94	0.03	380.86	0.10	0.32	201.32
f Assisted Living Concepts 2008.txt	3.68	387.02	2.54	28.11	1.89	0.01	365.41	0.03	0.09	194.02
f Assisted Living Concepts 2009.txt	3.69	385.94	2.56	28.30	1.88	0.01	364.27	0.03	0.10	196.34
f Bally Gaming and Systems 2004.txt	3.48	401.68	2.91	29.07	1.87	0.01	382.40	0.05	0.11	183.83
f Bally Gaming and Systems 2005.txt	3.70	401.83	2.79	27.21	1.84	0.02	382.01	0.05	0.13	193.79
f Beazer Homes 2005.txt	4.22	395.64	2.96	28.57	1.87	0.02	380.71	0.08	0.15	201.49
f Beazer Homes 2006.txt	4.04	397.83	2.83	26.85	1.86	0.02	381.69	0.06	0.13	205.37
f BROOKE CORPORATION 2006.txt	5.23	385.22	3.88	27.85	1.84	0.01	361.52	0.03	0.09	180.78
f BROOKE CORPORATION 2007.txt	4.28	392.35	2.99	27.02	1.87	0.01	367.00	0.02	0.08	187.01
f Cabletron Systems Inc 2000.txt	3.78	392.48	2.66	27.05	1.88	0.02	372.51	0.05	0.10	205.81
f Cabletron Systems Inc 2001.txt	3.73	392.92	2.56	26.41	1.90	0.02	375.20	0.04	0.10	207.06
f CHINA NATURAL GAS Inc 2008	3.30	407.33	2.08	26.28	1.79	0.02	391.94	0.07	0.15	197.87
f CHINA NATURAL GAS Inc 2010	3.37	400.59	2.13	30.44	1.74	0.01	379.03	0.04	0.09	192.26
f ChinaMedia Express Holdings 2008.txt	4.47	373.52	3.78	33.70	1.80	0.02	351.18	0.07	0.10	192.87
f ChinaMedia Express Holdings 2009.txt	3.13	396.76	2.18	30.75	1.82	0.01	371.84	0.02	0.07	196.67
f Computer Associates 2000.txt	3.14	394.34	2.47	25.07	1.91	0.04	378.53	0.09	0.17	197.47
f Computer Associates 2001.txt	3.07	392.23	2.48	25.43	1.91	0.03	376.47	0.07	0.15	199.14
f Computer Sciences Corporation 2009.tx	4.11	389.83	2.93	24.99	1.85	0.02	371.03	0.05	0.10	201.11
f Computer Sciences Corporation 2010.tx	4.30	389.31	3.05	25.21	1.84	0.02	370.50	0.05	0.12	207.93
f Diamond Foods 2010.txt	3.97	404.43	2.66	25.75	1.83	0.02	388.76	0.07	0.15	196.14

Figure 4.29: An extract of matrix derived from LBC shown in Table 4.6.

Therefore it is appropriate to extract the ratios in Table 4.6 from the corpus and put them through the classifier models to check for success in the discrimination task.

The values for the ratio such as number of verbs, group references an others were extracted using Coh-Metrix and LIWC. The ratios were calculated for each report and matrices of dimensions shown in Table 4.7 were composed. Figure 4.29 shows an extract of a matrix that is passed to downstream processes.

A few ratios were also individually plotted from the matrix (extract shown in Figure 4.29) to visualise any distinctions between fraud non-fraud reports represented by the LBCs. Again there is distinction between the two categories coming through that could filter through to the classifiers and aid in the discrimination task.

4.9 Topic Modelling

As Brown and Crowley [240] point out that although textual analysis methods as those discussed in this chapter provide incremental power in identifying misreporting they: "*examine how content is being disclosed as opposed to what is being disclosed*". A perusal of the corpus under study indicates that fraud firms display a propensity for irrelevance by: "*talking a lot about things that really don't matter much*" [241]. Lewis [241] maintains that this irrelevance is an attempt to mask the risks that the firms are facing.

In an attempt to lift the lid on the content in this corpus to reveal thematic content, a class of generative probabilistic models known as Topic Models or formally Latent Dirichlet Allocation is bought into use. This is a probabilistic model for uncovering the underlying semantic structure of a document collection based on a hierarchical bayesian analysis of the original texts [242] or a way to way to infer the latent structure or topics in a corpus or collection of documents.
In such a model, the annual reports/10-K are a mixture of topics where a topic that spit out words with certain probabilities [242]. Each topic is represented as a multinomial probability distribution over words.

4.9.1 Gibbs Sampling Based Latent Dirichlet Allocation (LDA) Algorithm

As explained by Blei [243] the LDA model works on the assumption that an author produces documents in the following manner:-

- Decide on the number of words in a document, according to a poisson distribution.
- Choose a topic mixture for the documents, according to a Dirichlet distribution over a set of K topics. For example, the author would choose the document to be based on two topics, could be 1/3 for product development and 2/3 for raising capital.
- Generate each word in the document by:
  - Picking a topic
  - Using the topic to generate words.

Assuming this generative model for a collection of documents, LDA then tries to backtrack from the documents to find a set of topics that are likely to have generated the collection.
Blei [243] further provides an example to highlight the steps above. It has been retrofitted to the corpus under study:-

- Pick 5 words to be the number of words in a document D.
- Decide that D will be ½ about product development and ½ about raising capital.
- Pick the first word to come from the product development topic which then gives you the word ‘innovation’.
- Pick the second word to come from the raising capital topic which gives the word ‘interest’.
- Pick the third word to come from the product development topic, giving you ‘feature’.

- Pick the fourth word to come from the raising capital topic, giving you ‘shares’.
- Pick the fifth word to come from the product development, giving you ‘marketing’.

Consequently this document generated based on the LDA model will be *‘innovation’, ‘interest’, ‘feature’, ‘shares’, ‘marketing’.* As can be noted LDA is a bag-of-words model.

Given that a document is generated using the above methodology, how does LDA reverse the process to arrive at the topics that generated the words in a document? This is the scenario that we have with the corpus under study. The LDA approach is deployed using a sampling based algorithm, Gibbs sampling. It is as described by Blei [243] as follows:-

- A fixed number of topics to discover is chosen. In this corpus 25 were selected and thought optimal. In the few studies that have been conducted into using LDA to uncover financial misreporting between 25 and 30 have been chosen [240].
- Go through each document, and randomly assign each word in a document to one of the K topics.

At a first cut, this random assignment gives both topic representations of all the documents and word distributions of all the topics, though not very accurate.

To improve on this, for each document *d*:-

- Go through each word *w* in *d*

And for each topic *t*, compute two things:

- p(topic *t* | document *d*) = the proportion of words in document d that are currently assigned to topic *t*.
- p(word *w* | topic *t*) = the proportion of assignments to topic *t* over all documents that come from this word *w*.
- Reassign *w* a new topic, where we choose topic *t* with probability p(topic *t* | document *d*) *p(word *w*|topic *t*). This is the probability that topic *t* generated word *w*, so the current word’s topic is resampled with this probability.

In other words, it is assumed that all topic assignments except for the current word in question are correct, and then updating the assignment of the current word using our model of how documents are generated.

After repeating the previous step a large number of times, eventually word assignment to topics is anticipated to be more accurate. These assignments are used to estimate the topic mixtures of each document (by counting the proportion of words assigned to

each topic within that document) and the words associated to each topic (by counting the proportion of words assigned to each topic overall).

The above logic can be made more concrete.

A word type *W* is randomly placed in a topic *Z*, to begin the frequency of this word type *W* in *Z* is multiplied by the number of words in document *D* that already belong to *Z*. The result represents the probability that this word came from Z. Formula shown in Eq 4.5:-

$$P(Z|W,D) = \frac{\#\ of\ word\ W\ in\ topic\ Z + \beta_w}{total\ tokens\ in\ Z + \beta} * (\#words\ in\ D\ that\ belong\ to\ Z + \alpha)$$

(Eq 4.5)

$\beta\ and\ \alpha$ are hyperparameters that capture the probability the word belongs to topic *z*. The above could just be a random guess that the word type belongs to topic *z*. This is then improved upon through using the above equation. This is done for the whole corpus, word by word and each word reassigned to a topic. Eventually the words will become more common in topics where they are already common. Also topics will become more common where they are already common. Ultimately, the model will gradually become more consistent as topics focus on specific words and documents. The above process is summarised by Blei [243]: "*Documents exhibit multiple topics. Each documents exhibits the topics in different proportions. Each word in each document is drawn from one of the topics, where the selected topic is chosen from the per document distribution over topics*". Blei [243] further adds that all the documents in the collection share the same set of topics but each document exhibits those topics in different proportions. As can be inferred from the above equation this generative process defines a joint probability distribution over both the observed and hidden random variables. This joint distribution: "*is used to compute the conditional distribution of the hidden variables given the observed variables. This conditional distribution is also called the posterior distribution*" [243]. The observed variables are the words of the documents, the hidden variables are the topic structure. The computational problem of: "*inferring the hidden topic structure from the documents is the problem of computing the posterior distribution, the conditional distribution of the hidden variables given the documents"* [243].

The graphical model for LDA is shown in Figure 4.30. This is extracted from Blei [243] and is widely used to illustrate the mechanics of LDA.

In Figure 4.30, the hidden nodes, the topic proportions, assignments, and topics are unshaded. The observed nodes, the words of the documents are shaded. The rectangles are '*plate*' notation, which denotes replication. The *N* plate denotes the collection words within documents; the *D* plate denotes the collection of documents within the collection. Using these parameters this joint defines a posterior $p(\theta, z, \beta|w)$, therefore the task of the algorithm is to infer per-word topic assignment $Z_{d,n}$, per-document topic proportions θ_d, per-corpus topic distributions β_k. To approximate to this posterior, Gibbs sampling as depicted above was used. There are others that could also be used [242].

LDA is similar to another algorithm Latent Semantic Analysis. This will be discussed in chapter five. However LDA describe a class of statistical models in which the semantic properties of words and documents are expressed in terms of probabilistic topics.

In sum, the posterior can be thought of as the reversal of the generative process described initially. Given the observed corpus, the posterior is a distribution of the hidden variables (the topics) which generated it.

4.9.2 Mallet and Results

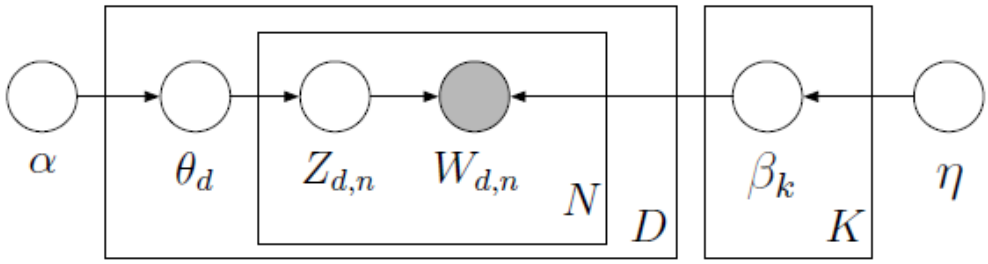
To execute the LDA over the corpus under study, MALLET was used [244]. This is a Java-based package for statistical NLP and includes functionality for Gibbs based LDA. The corpus was first converted into Mallet internal format. This format represents data as lists of instances. All Mallet instances include a data object. The files in the corpus are kept in the order that they were loaded into Mallet. All stopwords are stripped out as they obstruct analysis using the default English dictionary.

In Mallet, the command prompt is used to call the Java based programs that execute LDA over the corpus as below:-

bin\mallet train-topics --input tut4.mallet --num-topics 25 --optimize-interval 10 --output-state topic-state.gz --output-topic-keys tutorial_keys.txt --output-doc-topics tutorial_composition.txt

This command:-

- opens *tut4.mallet* file (the internal Mallet file that contains the corpus).
- trains MALLET to find 25 topics.



α $Proportions\ parameter$
θ_d $Per-document\ topic\ proportions$
$Z_{d,n}$ $Per-\ word\ topic\ assignment$
$W_{d,n}$ $Observed\ word$
β_k $Per-corpus\ topic\ distributions$
η $Topic\ Parameter$

Figure 4.30: The LDA process.

- outputs every word in your corpus of materials and the topic it belongs to into a compressed file.
- outputs a text document showing you what the top key words are for each topic (tutorial_keys.txt).
- outputs a text file indicating the breakdown, by percentage, of each topic. within each original text file you imported (tutorial_composition.txt).
- optimize-interval - This option turns on hyperparameter optimization, which allows the model to better fit the data by allowing some topics to be more prominent than others. As recommended by Mallet developers, it was set to 10, as optimising every 10 iterations was thought to be reasonable.

The compressed file contains one file of 3497418 lines long that assigns every word in the corpus to a topic. Extract is shown in Figure 4.31.

An extract of the file that contains details on the topics, weights attached to them (estimated distribution of the topic across the corpus) and the words most strongly associated with them is shown in Figure 4.32.

The final extract is the matrix that will be sent to the classifiers to determine if based on topics, fraud and non-fraud firms can be differentiated. The weights attached to each topic as assigned by the LDA algorithm is extracted for each file to construct a matrix to pass to feature selection and then the classifiers. An extract of the matrix is shown in Figure 4.33. Appendix G, Table H.6 shows a few plots of topics and the differences between the topic weights attached to them through LDA for a selection of fraud and non-fraud reports.

```
3497384  407 c:\mallet\data5\nf3Harris 2007.txt 17519 2091 instruments 10
3497385  407 c:\mallet\data5\nf3Harris 2007.txt 17520 2703 minimize 10
3497386  407 c:\mallet\data5\nf3Harris 2007.txt 17521 2951 currency 10
3497387  407 c:\mallet\data5\nf3Harris 2007.txt 17522 1168 risk 3
3497388  407 c:\mallet\data5\nf3Harris 2007.txt 17523 443 international 9
3497389  407 c:\mallet\data5\nf3Harris 2007.txt 17524 221 transactions 3
3497390  407 c:\mallet\data5\nf3Harris 2007.txt 17525 3422 gains 3
3497391  407 c:\mallet\data5\nf3Harris 2007.txt 17526 1850 losses 24
3497392  407 c:\mallet\data5\nf3Harris 2007.txt 17527 1902 resulting 3
3497393  407 c:\mallet\data5\nf3Harris 2007.txt 17528 2951 currency 3
3497394  407 c:\mallet\data5\nf3Harris 2007.txt 17529 580 rate 3
3497395  407 c:\mallet\data5\nf3Harris 2007.txt 17530 2819 fluctuations 4
3497396  407 c:\mallet\data5\nf3Harris 2007.txt 17531 261 material 24
3497397  407 c:\mallet\data5\nf3Harris 2007.txt 17532 81 effect 3
3497398  407 c:\mallet\data5\nf3Harris 2007.txt 17533 787 results 4
3497399  407 c:\mallet\data5\nf3Harris 2007.txt 17534 1837 fiscal 10
3497400  407 c:\mallet\data5\nf3Harris 2007.txt 17535 922 impact 3
3497401  407 c:\mallet\data5\nf3Harris 2007.txt 17536 1435 inflation 3
3497402  407 c:\mallet\data5\nf3Harris 2007.txt 17537 1030 extent 4
3497403  407 c:\mallet\data5\nf3Harris 2007.txt 17538 509 feasible 24
3497404  407 c:\mallet\data5\nf3Harris 2007.txt 17539 5199 consistently 4
3497405  407 c:\mallet\data5\nf3Harris 2007.txt 17540 1320 practice 4
3497406  407 c:\mallet\data5\nf3Harris 2007.txt 17541 5773 adjusting 3
3497407  407 c:\mallet\data5\nf3Harris 2007.txt 17542 838 prices 4
3497408  407 c:\mallet\data5\nf3Harris 2007.txt 17543 1824 reflect 3
3497409  407 c:\mallet\data5\nf3Harris 2007.txt 17544 922 impact 3
3497410  407 c:\mallet\data5\nf3Harris 2007.txt 17545 1435 inflation 24
3497411  407 c:\mallet\data5\nf3Harris 2007.txt 17546 4192 salaries 13
3497412  407 c:\mallet\data5\nf3Harris 2007.txt 17547 16522 fringe 9
3497413  407 c:\mallet\data5\nf3Harris 2007.txt 17548 676 benefits 4
3497414  407 c:\mallet\data5\nf3Harris 2007.txt 17549 1296 employees 4
3497415  407 c:\mallet\data5\nf3Harris 2007.txt 17550 672 cost 10
3497416  407 c:\mallet\data5\nf3Harris 2007.txt 17551 88 purchased 4
3497417  407 c:\mallet\data5\nf3Harris 2007.txt 17552 1770 materials 9
3497418  407 c:\mallet\data5\nf3Harris 2007.txt 17553 19 services 20
```

Figure 4.31: File extract showing word types in file assigned to topics.

```
0     0.30917    company believes sales approximately management net acquired
interest operations year systems time expects information generally material
operating significant results
1     0.02256    china prc rmb cme million game foreign advertising services
online revenues tax travel business internet income increase regulations
information
2     0.03792    million loans loan interest bank risk income credit securities
net market losses financial capital portfolio assets december commercial
increase
3     0.50806    million cash tax income financial due costs net operations
segment primarily increased related compared rate operating interest results
higher
4     0.61609    business future financial results including stock significant
market ability result adversely time based affect operations additional price
continue factors
5     0.19482    business year market billion growth group sales products
management businesses world strong share million investment operating markets
continued performance
6     0.03488    care health services medicare medicaid medical state programs
healthcare members plans program states providers hospitals benefits managed
facilities federal
7     0.04301    group year gaming million uk share profit board casino
directors publishing retail division shares dividend period machines betting
casinos
8     0.03264    production group million prices demand operations cost ore
costs xstrata mining projects coal cent biodiesel fy tonnes iron economic
9     0.11886    products product sales fiscal customers market development
manufacturing technology million systems technologies net customer applications
research design inventory revenue
10    0.04975    sales fiscal percent systems business products digital million
year imaging communications equipment color production materials canon sony
income display
11    0.0509     sales products ford vehicles business vehicle automotive
united corporation systems visteon percent product equipment engine segment
markets parts industrial
12    0.01619    products system systems fda medical procedures sensei sales
```

Figure 4.32: File extract showing final assignment of topics and weights.

	Topic 0	Topic 1	Topic 2	Topic 3	Topic 4	Topic 5	Topic 6	Topic 7	Topic 8	Topic 9	Topic 10	Topic 11	Topic 12	Topic 13
file:/c:/mallet/data5/f%20adelphia%201999.txt	0.04476023	1.79E-06	3.01E-06	0.00155	0.057657	1.55E-05	2.77E-06	3.42E-06	2.59E-06	9.44E-06	3.95E-06	4.04E-06	1.29E-06	0.005174
file:/c:/mallet/data5/f%20Adelphia%202000.txt	0.036553499	1.98E-06	3.33E-06	0.001713	0.050541	0.002827	3.06E-06	3.78E-06	2.87E-06	1.04E-05	4.37E-06	4.47E-06	1.42E-06	0.006771
file:/c:/mallet/data5/f%20Anicom%20Inc%201998.txt	0.330376287	5.87E-06	9.86E-06	0.107803	0.131498	0.022417	9.07E-06	1.12E-05	8.49E-06	0.187025	1.29E-05	1.32E-05	4.21E-06	0.077273
f%20Anicom%20Inc%201999.txt	0.241864849	6.03E-06	2.77E-04	0.090437	0.189583	0.01528	9.32E-06	2.79E-04	8.72E-06	0.184908	1.33E-05	1.36E-05	4.33E-06	0.04438
f%20Applied%20Microsystems%202001.txt	0.136122172	0.017818441	6.43E-06	0.058959	0.270038	3.72E-04	5.92E-06	7.30E-06	5.54E-06	0.384985	8.44E-06	8.64E-06	2.75E-06	0.119972
f%20Applied%20Microsystems%202002.txt	0.011051891	1.39E-04	4.13E-04	0.051584	0.323991	2.65E-05	4.74E-06	0.012647	4.44E-06	0.423284	6.76E-06	1.43E-04	2.20E-06	0.06526
f%20Assisted%20Living%20Concepts%202008.txt	0.002612164	7.36E-05	2.73E-06	0.086367	0.192993	1.40E-05	2.51E-06	3.09E-06	2.35E-06	8.55E-06	3.58E-06	3.66E-06	1.16E-06	8.46E-06
f%20Assisted%20Living%20Concepts%202009.txt	2.37E-05	1.73E-06	2.91E-06	0.107066	0.197914	1.49E-05	2.68E-06	3.30E-06	7.92E-05	9.12E-06	3.82E-06	3.90E-06	1.24E-06	9.02E-06
f%20Bally%20Gaming%20and%20Systems%202004.txt	0.016241877	0.01704964	0.001167	0.038296	0.140841	9.94E-05	2.90E-06	0.339462	2.71E-06	0.082089	0.006657	0.002749	1.35E-06	0.051403
f%20Bally%20Gaming%20and%20Systems%202005.txt	0.009246029	2.36E-06	3.97E-06	0.133547	0.18015	1.25E-04	3.65E-06	0.245737	3.42E-06	0.125234	5.21E-06	0.007125	1.70E-06	0.065869
f%20Beazer%20Homes%202005.txt	9.78E-04	3.50E-06	0.001556	0.072005	0.27013	3.02E-05	5.41E-06	1.62E-04	5.06E-06	1.84E-05	7.71E-06	7.89E-06	2.51E-06	1.82E-05
f%20Beazer%20Homes%202006.txt	0.0026841	2.84E-06	4.78E-06	0.127661	0.230458	2.45E-05	4.39E-06	5.42E-06	4.11E-06	1.50E-05	6.27E-06	6.41E-06	2.04E-06	1.48E-05
f%20BROOKE%20CORPORATION%202006.txt	0.014845736	1.33E-06	0.006618	0.016157	0.204254	1.30E-04	2.06E-06	2.54E-06	1.93E-06	7.02E-06	2.94E-06	3.01E-06	9.56E-07	6.94E-06
f%20BROOKE%20CORPORATION%202007.txt	0.019696014	2.36E-04	0.052611	7.28E-04	0.180638	0.001982	0.00282	2.02E-06	1.53E-06	3.81E-04	4.93E-05	2.39E-06	7.61E-07	5.52E-06
f%20Cabletron%20Systems%20Inc%202000.txt	0.089268069	2.05E-06	3.45E-06	0.111096	0.183352	1.77E-05	3.17E-06	3.91E-06	2.97E-06	0.483197	4.52E-06	4.63E-06	1.47E-06	0.039905
f%20Cabletron%20Systems%20Inc%202001.txt	0.059766712	1.81E-06	3.04E-06	0.110876	0.231169	0.002418	2.79E-06	3.44E-06	2.61E-06	0.442709	3.98E-06	4.08E-06	1.30E-06	0.080173
f%20CHINA%20NATURAL%20GAS,%20INC.%202008.txt	0.044890934	0.297500875	5.15E-06	0.022494	0.235473	0.013889	1.41E-04	5.85E-06	4.44E-06	1.62E-05	6.76E-06	0.012103	2.20E-06	1.52E-04
f%20CHINA%20NATURAL%20GAS,%20INC.%202010.txt	0.006608104	0.37808099	2.40E-06	0.031454	0.226329	3.92E-04	2.21E-06	2.72E-06	0.001142	7.53E-06	3.15E-06	0.005198	1.03E-06	7.45E-06
f%20ChinaMedia%20Express%20Holdings%202008.txt	0.016868252	0.010018346	0.003946	2.67E-04	0.692338	1.27E-04	3.72E-06	0.006611	3.48E-06	1.27E-05	0.003308	5.42E-06	1.73E-06	1.25E-05
f%20ChinaMedia%20Express%20Holdings%202009.txt	0.015581025	0.634442303	1.59E-06	0.002867	0.195506	0.010136	1.69E-04	1.80E-06	1.37E-06	0.004357	0.003476	2.13E-06	6.78E-07	4.92E-06
f%20Computer%20Associates%202000.txt	0.132079367	4.14E-06	0.005148	0.133768	0.126444	0.027946	6.41E-06	7.90E-06	5.99E-06	0.04611	9.13E-06	9.35E-06	2.97E-06	0.445483
f%20Computer%20Associates%202001.txt	0.145907137	3.25E-06	5.46E-06	0.12446	0.147969	0.034908	5.03E-06	6.20E-06	4.70E-06	0.035041	7.17E-06	7.34E-06	2.33E-06	0.452883
f%20Computer%20Sciences%20Corporation%202009.txt	0.05115367	2.02E-06	3.39E-06	0.495128	0.171309	6.43E-04	3.12E-06	3.84E-06	2.92E-06	1.06E-05	5.41E-04	4.55E-06	1.45E-06	0.258412
f%20Computer%20Sciences%20Corporation%202010.txt	0.049611833	2.19E-06	3.69E-06	0.479826	0.198387	8.94E-04	3.39E-06	4.18E-06	3.17E-06	1.16E-05	4.84E-06	4.95E-06	1.57E-06	0.258906
f%20Diamond%20Foods%202010.txt	0.010152877	3.17E-06	5.32E-06	0.102008	0.337209	2.74E-05	0.001128	6.04E-06	4.58E-06	0.023465	6.98E-06	7.15E-06	0.001406	1.65E-05
f%20Diamond%20Foods%202011.txt	0.001560804	3.41E-06	5.74E-06	0.139365	0.341197	6.35E-04	5.28E-06	6.51E-06	0.002125	0.021063	7.53E-06	7.71E-06	2.45E-06	1.69E-04
f%20enrc%202008.txt	0.006416676	2.22E-06	3.73E-06	0.082581	0.050463	0.196717	3.43E-06	4.23E-06	0.615446	1.17E-05	4.89E-06	5.00E-06	1.59E-06	0.001485

Figure 4.33: File extract showing final assignment of topics and weights.

	Topics
Matched Pair	204 rows (102 'f' reports) and (102 'nf' reports) by 25 topics
Peer Set	408 rows (102 'f' reports) and (308 'nf' reports) by 25 topics

Table 4.8: Dimension of matrices constructed using Topic modelling.

Table 4.8 below shows the dimensions of the topic matrices that are ready for feature selection.

As argued by Brown and Crowley [240] this method offers a unique advantage in that researchers are not required to know the topics commonly discussed in annual reports at a given point in time. It therefore remains free from personal bias on what could be a topic in these reports. The functionality afforded by LDA to analyze the actual content of a large collection of financial statements in an automated manner is itself a step forward. Previously such tasks remained confined to manual/ semi-automated content analysis approaches.

4.10 Concept Mining

Concept Mining is used to extract the concepts embedded in the reports. These concepts can be either words or phrases. The process for concept identification and

extraction used over the corpus is depicted in Figure 4.34.

The end objective of this process above was to examine all tokens for the reports in the corpus and to pick up all synonyms, those that had high word similarity scores using WordNet [245]. This would provide a condensed list of word types with a measure of the strength of its occurrence in the corpus. This routine would be performed for each document. This results in a vector of concepts scores with the class of the document affixed (‘*f*’ or ‘*nf*’).

Concept mining process as described above aims to pick up all related words and group into a concept. This would provide a better appreciation of content in the reports and highlight differences between the two report types. Therefore when the final matrix is constituted and passed to the classifiers it would aid in discrimination. Words are used to convey ideas and traditionally all similar words or phrases in the text are picked up using a thesaurus, typically Princeton’s Wordnet. Wordnet is a: “*large lexical database of English. Nouns, verbs, adjectives and adverbs are grouped into sets of cognitive synonyms (synsets), each expressing a distinct concept. Synsets are interlinked by means of conceptual-semantic and lexical relations”* [246].

The process as described in Figure 4.34 would take each word from a report, get the synset for it using WordNet, for example the word *‘liability’* has been determined through Part Of speech (POS) tagging that it has been used as a noun in the corpus. To determine all the different senses or synoynms of this word the following command is input to wordnet through python.

wordnet.synsets('liability', pos=wordnet.NOUN)

This return the following:-

[Synset('liability.n.01'), Synset('indebtedness.n.01'), Synset('liability.n.03')]

The output denotes the different senses the word liability is commonly used for, Wordnet gives the following definitions for the synsets:-

wordnet.synset('liability.n.01').definition()

'the state of being legally obliged and responsible'

wordnet.synset('indebtedness.n.01').definition()

'an obligation to pay money to another party'

wordnet.synset('liability.n.03').definition()

'the quality of being something that holds you back'

In the process described, each synset of a word is compared to the synset of every word in a report using the following command, comparing *‘liability’* to *‘loss’*:-

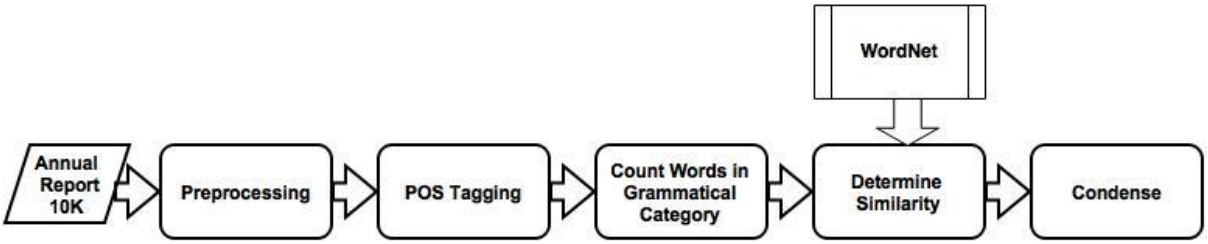


Figure 4.34: Concept mining process executed over the corpus.

liability.wup_similarity(loss)

The above command would return a similarity score between 0 and 1. Jurafsky and Martin [17] provide an explanation as to how this similarity score was attained. It is based on its organisational structure within WordNet. It computes path similarity based on the shortest number of edges from one word sense to another word sense, assuming a hierarchical structure like WordNet. In general, word senses which have a longer path distance are less similar than those with a very short path distance, e.g. man, dog versus man, tree (expectation is that man is more similar to dog than it is to tree).

The process in Figure 4.34 is further expanded with python code and data extracts below. An extract from an annual report for firm Adelphia, year 2000 is shown below. The Rigas family that founded the company was accused by the Securities Exchange Commission, in 2002 of fraud on a massive scale which included falsification of financial results.

"Adelphia is a leader in the telecommunications industry with cable television and local telephone operations. Adelphia's operations consist of providing telecommunications services primarily over networks, which are commonly referred to as broadband networks because they can transmit large quantities of voice, video and data by way of digital or analog signals. As of December 31, 2000, Adelphia owned or managed cable television systems ("Systems") with broadband networks that passed in front of 9,020,540 homes and served 5,741,368 basic subscribers. John J. Rigas, the Chairman, President, Chief Executive Officer and founder of Adelphia, has owned and operated cable television systems since 1952"

The first step of pre-processing the file in Python 3.5 is shown in Figure 4.35. The file is opened and contents read into a python list. Each line from file is cleared of digits,

all text is put into lower case, all brackets, roman numerals removed. Each line read is also split into tokens (by individual words that make up a sentence). In the last line all tokens are given part of speech tags. An example extract of the output in list tokens as depicted in the last line of code from Figure 4.35 is shown in Figure 4.36.

Once the code in Figure 4.35 has been executed, the program continues as shown in code extract Figure 4.37. This code removes all stop words and then puts all nouns, verbs, adjectives, adverbs into separate lists. An example of an output from the list Ncounts that contains all nouns is shown in Figure 4.38. For example, the word adelphia occurred 3 times, angeles 1 and so on. The program then calls the function shown in Figure 4.39. This take in as an argument a list containing tokens of a particular grammatical structure, like the Ncounts list data structure shown in Figure 4.38. For example if the Ncounts list is passed to it as an argument it would perform the following tasks:-

1. The first for loop, begins taking the first entry in Ncounts – 'adelphia'.
2. It then accesses WordNet and picks up synonyms for 'adelphia'.
3. As 'adelphia' has no synonyms it would continue down the list until it came to a word that has a synonym. For the list Ncounts this would be 'cities'.
4. Another for loop is started to pick up all words in the rest of the data structure Ncounts in this instance.
5. Synsets for all words in the list are also picked up.
6. It then compares the synonyms for in this instance 'cities' (the first token in Ncounts that has synonyms) with all other words in the list.
7. If two words have a similarity score greater than 0.5, then the two words with the similarity score and a combined count of occurrences of the two words is added to another list data structure (newlist). This data structure (newlist) is the output that is returned by the function. An extract of it is shown in Figure 4.40.

As can be seen, the first token in the nouns list (Ncounts) that has synonyms is compared to all other tokens in the list. If similarity score is greater than 0.5 then the pair of tokens is added to the new list shown in Figure 4.40 with the score and combined count of the occurrences.

Once the above data has been derived the program to mine concepts continues as depicted in Figure 4.41. This function takes in the following arguments:-

- *file2*, this would be the name of the report ("Adelphia 2000').

- The output from get_simscore function - *sim1(new lis*t) list as shown in Figure 4.40.
- The part of speech (*pos*) that is being examined '*nouns*' in the examples above.
- *Cat*, for category being 'f' or 'nf'.

This function get_concepts, iterates through list sim1 to achieve the following goals:-

- Tokens are amalgamated to word types eg in the example above all occurrences of token '*areas*' with other tokens is reduced to one entry. This is done by counting up the number of times a token occurred in a combination with another token (variable called *tally* in excerpt shown in Figure 4.41). For example the token '*areas*' occurred 11 times in a combination with other tokens.
- The similarity scores from WordNet for each occurrence of a token ('area') with another token is added up as shown in the sim (newlist) list in Figure 4.40. The line (simscore + = flist [i]) from Figure 4.41 performs this task. For example the total similarity score for *'areas*' would be 6.84.
- This total similarity score is then divided by output from step 1 (tally) to get an average similarity score for word type or concept (eg 'areas'). For areas this is 0.62.
- The function also adds up the total occurrences of the token being examined ('*areas*') with all other tokens in the list. For example, from list *sim1* the total occurrence of '*areas*' with '*cities*' is 2 times, '*areas*' and '*colorado*' is again 2 times and so on. These scores are added to provide an indication of the strength of the new derived concept in the text. The strength for '*cities*' counting up all the occurrences with other tokens is 24.
- The above derived data for each concept is then added to a list data structure (tlist) with the report name for example - f Adelphia 2000 and the category of the report, 'f', in this instance. The function in figure 4.41 returns this data list. An excerpt from the list is shown in Figure 4.42.

Finally the program executes the last function shown in Figure 4.43. This function takes the list con1 shown in Figure 4.42 as an argument and writes each entry in the list to a file. The entries in the file have the format shown in Figure 4.44.

This file is loaded into Excel for further processing. The similarity score and the number of counts or occurrences of concept in text is multiplied and then divided by the total number of concepts found in the file. This normalises the scores and enables better comparability. Also the concept and POS to which it referred is concatenated. This is

known as feature engineering as these two feature are strongly related to each other and are better amalgamated. As Domingos [247] puts it: "*the raw data is not in a form that is amenable to learning, but you can construct features from it that are. This is typically where most of the effort in a machine learning project goes. It is often also one of the most interesting parts, where intuition, creativity and "black art" are as important as the technical stuff".* The data is reconstituted in excel, now takes the format shown in Figure 4.45.

However it is still not in a form that could be successfully passed to the classifiers. The reports for each concepts are repeated and would not pass data normalisation processes typically used to remove redundancy in databases. Further the format in Figure 4.45 is not in the standard matrix form that was the end result for the other document representation schemes shown earlier. Therefore, another python program is written to obtain a matrix that would remove redundancy and reconstitute the data in the matrix in a more standard way used. This is shown in Figure 4.46.

The program begins by reading in the data as depicted in Figure 4.45 into a python list data structure. It then puts into another list (s2) all unique concepts (this is done by putting in all concepts into a python *set* data structure that removes all duplicates). All filenames or report names are put into another data structure with again all repeating values removed. The concept names would denote header information or the column names in the matrix. For each file the concepts scores are retrieved for all concepts that are in the s2 list, zeros are added to concepts that have no weighting in the file.

Once a vector has been formed for each file it is written to a file. The final matrix formed is shown Figure 4.47. This matrix contains 102 fraud firms and 102 non-fraud firms, results in 204 rows. Concepts are 5595 columns wide. This matrix composition conforms to the general rule in a corpus where most words or concepts occur only once or a few times in a document [248]. This results in sparse matrices which as will be shown in Chapter 5 would need to go through sparsity reduction methods in R to enable the classifiers to generalise better.

The above technique is within the realm of data mining which is about finding insights which are statistically reliable, unknown previously and actionable from data [249]. The above concepts could only have been achieved through data mining and not through query and reporting tools. Further, as Phua et al. [250] argues that the: "*data must be available, relevant, adequate and clean*". The corpus was shown to be

balanced and representative as outlined in chapter 3 and fulfils the criteria set by McNamara et al. [50]. The matrix shown in Figure 4.47 will now go through further processing highlighted in chapter 5 before being passed to the classifiers.

Figure 4.48 to Figure 4.51 show the top 30 concepts in matrices constructed of dimension given in Table 4.9 plotted. These concepts are also tabulated in Appendix G, Table G.13 to Table G.16 (the top 60 by greatest difference in concept scores). The concepts scores are examined from the fraud report, identifying the most prominent concepts matched with the corresponding concepts in the non-fraud reports. Both the matched pair and peer set data set ups are used to identify prominent concepts. The converse is also done from the non-fraud angle to identify concepts prominent in non-fraud reports as compared to the corresponding concept in the fraud reports. A few choice concepts are also plotted in Appendix H, Table H.7 to demonstrate that their remains differences in linguistic use between the two category of reports.

```
f=open("C:/data6/wa/excerpt1.txt", 'r')
file1 = 'f Adelphia 2000'
tokens=[]
sent_list = []

###all sentences

for line in f:
    line = ''.join(i for i in line if not i.isdigit())
    line = line.lower()
    line = re.sub('[()]',"", line)  ## get rid of brackets
    line = re.sub(r"\b(i)\b|\b(ii)\b|\b(iii)\b|\b(iv)\b|\b(v)\b|\b(vi)\b(xi)\b(xii)\b", "", line)
    sent_list+= nltk.sent_tokenize(line)


## removes punctuaution

tokenizer = RegexpTokenizer(r'\w+')

for sent in sent_list:
    tokens+= tokenizer.tokenize(sent)
f.close()

tokens = nltk.pos_tag(tokens)
```

Figure 4.35: Pre-processing the reports in Python 3.5.

```
>>> tokens
[('adelphia', 'NN'), ('leader', 'NN'), ('telecommunications', 'NNS'), ('industry', 'NN'), ('cable', 'NN'), ('television', 'NN'), ('local', 'JJ'), ('telephone', 'NN'),
('operations', 'NNS'), ('adelphia', 'VBP'), ('operations', 'NNS'), ('consist', 'NN'), ('providing', 'VBG'), ('telecommunications', 'NNS'), ('services', 'NNS'), ('prima
rily', 'RB'), ('networks', 'NNS'), ('commonly', 'RB'), ('referred', 'VBN'), ('broadband', 'NN'), ('networks', 'NNS'), ('transmit', 'VB'), ('large', 'JJ'), ('quantities
', 'NNS'), ('voice', 'NN'), ('video', 'NN'), ('data', 'NNS'), ('way', 'NN'), ('digital', 'JJ'), ('analog', 'JJ'), ('signals', 'NNS'), ('december', 'NNP'), ('adelphia',
'NN'), ('owned', 'VBD'), ('managed', 'VBD'), ('cable', 'NN'), ('television', 'NN'), ('systems', 'NNS'), ('systems', 'NNS'), ('broadband', 'NN'), ('networks', 'NNS'), (
'passed', 'VBD'), ('front', 'NN'), ('homes', 'NNS'), ('served', 'VBD'), ('basic', 'JJ'), ('subscribers', 'NNS'), ('john', 'VBP'), ('j', 'NN'), ('rigas', 'VBP'), ('chai
rman', 'NN'), ('president', 'NN'), ('chief', 'JJ'), ('executive', 'JJ'), ('officer', 'NN'), ('founder', 'NN'), ('adelphia', 'NN'), ('owned', 'VBN'), ('operated', 'VBN'
), ('cable', 'NN'), ('television', 'NN'), ('systems', 'NNS'), ('since', 'IN'), ('cable', 'NN'), ('systems', 'NNS'), ('owned', 'VBN'), ('company', 'NN'), ('company', 'N
N'), ('systems', 'NNS'), ('located', 'VBN'), ('states', 'NNS'), ('puerto', 'VB'), ('rico', 'NN'), ('organized', 'VBN'), ('six', 'CD'), ('core', 'NN'), ('clusters', 'NN
S'), ('los', 'VBP'), ('angeles', 'NNS'), ('pony', 'NN'), ('western', 'JJ'), ('pennsylvania', 'NN'), ('ohio', 'NN'), ('western', 'JJ'), ('new', 'JJ'), ('york', 'NN'), (
'new', 'JJ'), ('england', 'NN'), ('florida', 'NN'), ('virginia', 'NN'), ('colorado', 'NN'), ('springs', 'NNS'), ('company', 'NN'), ('systems', 'NNS'), ('located', 'VBN
'), ('primarily', 'RB'), ('suburban', 'JJ'), ('areas', 'NNS'), ('large', 'JJ'), ('medium', 'NN'), ('sized', 'JJ'), ('cities', 'NNS'), ('within', 'IN'), ('largest', 'JJ
S'), ('television', 'NN'), ('markets', 'NNS'), ('december', 'NN'), ('broadband', 'NN'), ('networks', 'NNS'), ('company', 'NN'), ('systems', 'NNS'), ('passed', 'VBN'),
('front', 'NN'), ('homes', 'NNS'), ('served', 'VBD'), ('basic', 'JJ'), ('subscribers', 'NNS'), ('adelphia', 'VBP'), ('also', 'RB'), ('provides', 'VBZ'), ('management',
'NN'), ('consulting', 'NN'), ('services', 'NNS'), ('partnerships', 'NNS'), ('corporations', 'NNS'), ('limited', 'JJ'), ('liability', 'NN'), ('companies', 'NNS'), ('eng
aged', 'VBN'), ('ownership', 'NN'), ('operation', 'NN'), ('cable', 'NN'), ('television', 'NN'), ('systems', 'NNS'), ('managed', 'JJ'), ('entities', 'NNS'), ('john', 'V
BP'), ('j', 'NN'), ('rigas', 'NN'), ('members', 'NNS'), ('immediate', 'JJ'), ('family', 'NN'), ('collectively', 'RB'), ('rigas', 'NNS'), ('family', 'NN'), ('including'
, 'VBG'), ('entities', 'NNS'), ('control', 'VBP'), ('controlling', 'VBG'), ('ownership', 'NN'), ('interests', 'NNS'), ('entities', 'NNS'), ('december', 'NN'), ('broadb
and', 'NN'), ('networks', 'NNS'), ('cable', 'NN'), ('systems', 'NNS'), ('owned', 'VBN'), ('rigas', 'NNS'), ('family', 'NN'), ('partnerships', 'NNS'), ('corporations',
'NNS'), ('passed', 'VBN'), ('front', 'NN'), ('homes', 'NNS'), ('served', 'VBD'), ('basic', 'JJ'), ('subscribers', 'NNS')]
```

Figure 4.36: An extract showing tokens from file given their POS tags.

```
## Remove stopwords
x=len(tokens)-1
stop = set(stopwords.words('english'))
while x >= 0:
  if tokens[x][0] in stop:
     del tokens[x]
  x = x - 1

## pick up the POS categories

nouns = []
verbs = []
adjectives = []
adverbs = []
missing = []
Ncounts =[]
Vcounts = []
Adjcounts = []
Advcounts = []

for i in range(0, len(tokens)-1):

    if tokens [i][1] == 'NN' or tokens [i][1] == 'NNS' or tokens [i][1] == 'NNP': ## noun
         nouns.append(tokens[i][0])
    elif tokens [i][1] == 'VB' or tokens [i][1] == 'VBG' or tokens [i][1] == 'VBZ'
    or tokens [i][1] == 'VBD' or tokens [i][1] == 'VBP' or tokens [i][1] == 'VBN': ## verb
         verbs.append(tokens[i][0])
    elif tokens [i][1] == 'JJ': ##adjective
        adjectives.append(tokens[i][0])
    elif tokens [i][1] == 'RB': ##adverb
        adverbs.append(tokens[i][0])
    else:
        missing.append(tokens[i][0])

Ncounts = [[x,nouns.count(x)] for x in set(nouns)]
Ncounts.sort()
Vcounts = [[x,verbs.count(x)] for x in set(verbs)]
Vcounts.sort()
Adjcounts = [[x,adjectives.count(x)] for x in set(adjectives)]
Adjcounts.sort()
Advcounts = [[x,adverbs.count(x)] for x in set(adverbs)]
```

Figure 4.37: Python code that separates out word types based on POS category.

```
>>> Ncounts
[['adelphia', 3], ['angeles', 1], ['areas', 1], ['broadband', 4], ['cable', 6], ['chairman', 1], ['cities', 1], ['clusters', 1], ['colorado', 1], ['companies', 1], ['
ompany', 4], ['consist', 1], ['consulting', 1], ['core', 1], ['corporations', 2], ['data', 1], ['december', 3], ['england', 1], ['entities', 3], ['family', 3], ['flor
da', 1], ['founder', 1], ['front', 3], ['homes', 3], ['industry', 1], ['interests', 1], ['j', 2], ['leader', 1], ['liability', 1], ['management', 1], ['markets', 1],
'medium', 1], ['members', 1], ['networks', 5], ['officer', 1], ['ohio', 1], ['operation', 1], ['operations', 2], ['ownership', 2], ['partnerships', 2], ['pennsylvania
, 1], ['pony', 1], ['president', 1], ['quantities', 1], ['rico', 1], ['rigas', 3], ['services', 2], ['signals', 1], ['springs', 1], ['states', 1], ['subscribers', 2],
['systems', 9], ['telecommunications', 2], ['telephone', 1], ['television', 5], ['video', 1], ['virginia', 1], ['voice', 1], ['way', 1], ['york', 1]]
```

Figure 4.38: Output from list data structure in Python showing all noun word types and counts.

```
def get_simscore(poscount, newlist):
    d=0.00
    value = 0.5
    syns2 = []
    syns1 = []

    for  m1 in range(0, len(poscount)):
        syns1 = wordnet.synsets(poscount[m1][0])
        for  m2 in range(0, len(poscount)):
            if m2 > m1:
                print (poscount[m1], poscount[m2])
                syns2 = wordnet.synsets(poscount[m2][0])
                if syns1!=[] and syns2!=[]:
                        d =  syns1[0].wup_similarity(syns2[0])
                        if d is not None:
                            if d>=value:
                                d = round(d,2)
                                newlist.append ([d, poscount[m1][0], poscount[m2][0],
                                                 poscount[m1][1] + poscount[m2][1]])
```

Figure 4.39 Python function that computes similarity of word types using WordNet.

```
>>> sim1
[[0.67, 'areas', 'cities', 2], [0.62, 'areas', 'colorado', 2], [0.62, 'areas', 'england', 2], [0.62, 'areas', 'florida', 2], [0.5, 'areas', 'front', 4], [0.53, 'areas'
, 'homes', 4], [0.62, 'areas', 'ohio', 2], [0.62, 'areas', 'pennsylvania', 2], [0.71, 'areas', 'rigas', 4], [0.71, 'areas', 'states', 2], [0.62, 'areas', 'virginia', 2
], [0.6, 'cable', 'signals', 7], [0.55, 'cable', 'video', 7], [0.67, 'chairman', 'leader', 2], [0.6, 'chairman', 'members', 2], [0.57, 'chairman', 'officer', 2], [0.52
, 'chairman', 'president', 2], [0.6, 'chairman', 'subscribers', 3], [0.67, 'cities', 'areas', 2], [0.74, 'cities', 'colorado', 2], [0.74, 'cities', 'england', 2], [0.7
4, 'cities', 'florida', 2], [0.74, 'cities', 'ohio', 2], [0.74, 'cities', 'pennsylvania', 2], [0.9, 'cities', 'rigas', 4], [0.82, 'cities', 'states', 2], [0.74, 'citie
s', 'virginia', 2], [0.67, 'clusters', 'core', 2], [0.73, 'clusters', 'data', 2], [0.55, 'clusters', 'networks', 6], [0.62, 'colorado', 'areas', 2], [0.74, 'colorado',
'cities', 2], [0.7, 'colorado', 'england', 2], [0.9, 'colorado', 'florida', 2], [0.9, 'colorado', 'ohio', 2], [0.9, 'colorado', 'pennsylvania', 2], [0.67, 'colorado',
'rigas', 4], [0.89, 'colorado', 'states', 2], [0.9, 'colorado', 'virginia', 2], [1.0, 'companies', 'company', 5], [0.62, 'companies', 'corporations', 3], [0.5, 'compan
ies', 'data', 2], [0.71, 'companies', 'family', 4], [0.67, 'companies', 'industry', 2], [0.5, 'companies', 'networks', 6], [0.67, 'companies', 'partnerships', 3], [1.0
, 'company', 'companies', 5], [0.62, 'company', 'corporations', 6], [0.5, 'company', 'data', 5], [0.71, 'company', 'family', 7], [0.67, 'company', 'industry', 5], [0.5
, 'company', 'networks', 9], [0.67, 'company', 'partnerships', 6], [0.67, 'core', 'clusters', 2], [0.73, 'core', 'data', 2], [0.55, 'core', 'networks', 6], [0.62, 'cor
porations', 'companies', 3], [0.62, 'corporations', 'company', 6], [0.62, 'corporations', 'family', 5], [0.71, 'corporations', 'industry', 3], [0.82, 'corporations', '
partnerships', 4], [0.73, 'data', 'clusters', 2], [0.5, 'data', 'companies', 2], [0.5, 'data', 'company', 5], [0.73, 'data', 'core', 2], [0.5, 'data', 'family', 4], [0
.6, 'data', 'networks', 6], [0.5, 'data', 'quantities', 2], [0.55, 'december', 'quantities', 4], [0.67, 'december', 'springs', 4], [0.62, 'england', 'areas', 2], [0.74
, 'england', 'cities', 2], [0.7, 'england', 'colorado', 2], [0.7, 'england', 'florida', 2], [0.7, 'england', 'ohio', 2], [0.7, 'england', 'pennsylvania', 2], [0.67, 'e
ngland', 'rigas', 4], [0.78, 'england', 'states', 2], [0.7, 'england', 'virginia', 2], [0.5, 'entities', 'quantities', 4], [0.71, 'family', 'companies', 4], [0.71, 'fa
mily', 'company', 7], [0.62, 'family', 'corporations', 5], [0.5, 'family', 'data', 4], [0.67, 'family', 'industry', 4], [0.5, 'family', 'networks', 8], [0.67, 'family'
, 'partnerships', 5], [0.62, 'florida', 'areas', 2], [0.74, 'florida', 'cities', 2], [0.9, 'florida', 'colorado', 2], [0.7, 'florida', 'england', 2], [0.9, 'florida',
'ohio', 2], [0.9, 'florida', 'pennsylvania', 2], [0.67, 'florida', 'rigas', 4], [0.89, 'florida', 'states', 2], [0.9, 'florida', 'virginia', 2], [0.5, 'front', 'areas'
, 4], [0.53, 'homes', 'areas', 4], [0.71, 'homes', 'rigas', 6], [0.67, 'industry', 'companies', 2], [0.67, 'industry', 'company', 5], [0.71, 'industry', 'corporations'
, 3], [0.67, 'industry', 'family', 4], [0.75, 'industry', 'partnerships', 3], [0.62, 'interests', 'liability', 2], [0.53, 'interests', 'operation', 2], [0.6, 'j', 'qua
ntities', 3], [0.67, 'leader', 'chairman', 2], [0.53, 'leader', 'medium', 2], [0.67, 'leader', 'members', 2], [0.63, 'leader', 'officer', 2], [0.5, 'leader', 'pony', 2
], [0.57, 'leader', 'president', 2], [0.67, 'leader', 'subscribers', 3], [0.53, 'leader', 'systems', 10], [0.5, 'leader', 'telecommunications', 3], [0.62, 'liability',
'interests', 2], [0.62, 'liability', 'operation', 2], [0.5, 'liability', 'way', 2], [0.67, 'management', 'markets', 2], [0.62, 'management', 'operations', 3], [0.59, '
management', 'services', 3], [0.67, 'markets', 'management', 2], [0.67, 'markets', 'operations', 3], [0.75, 'markets', 'services', 3], [0.53, 'medium', 'leader', 2], [
0.86, 'medium', 'systems', 10], [0.93, 'medium', 'telecommunications', 3], [0.75, 'medium', 'telephone', 2], [0.82, 'medium', 'television', 6], [0.6, 'members', 'chair
man', 2], [0.67, 'members', 'leader', 2], [0.57, 'members', 'officer', 2], [0.52, 'members', 'president', 2], [0.6, 'members', 'subscribers', 3], [0.55, 'networks', 'c
lusters', 6], [0.5, 'networks', 'companies', 6], [0.5, 'networks', 'company', 9], [0.55, 'networks', 'core', 6], [0.6, 'networks', 'data', 6], [0.5, 'networks', 'famil
y', 8], [0.5, 'networks', 'quantities', 6], [0.57, 'officer', 'chairman', 2], [0.63, 'officer', 'leader', 2], [0.57, 'officer', 'members', 2], [0.5, 'officer', 'presid
ent', 2], [0.57, 'officer', 'subscribers', 3], [0.62, 'ohio', 'areas', 2], [0.74, 'ohio', 'cities', 2], [0.9, 'ohio', 'colorado', 2], [0.7, 'ohio', 'england', 2], [0.9
, 'ohio', 'florida', 2], [0.9, 'ohio', 'pennsylvania', 2], [0.67, 'ohio', 'rigas', 4], [0.89, 'ohio', 'states', 2], [0.9, 'ohio', 'virginia', 2], [0.53, 'operation', '
interests', 2], [0.62, 'operation', 'liability', 2], [0.55, 'operation', 'way', 2], [0.62, 'operations', 'management', 3], [0.67, 'operations', 'markets', 3], [0.59, '
operations', 'services', 4], [0.57, 'ownership', 'quantities', 3], [0.5, 'ownership', 'signals', 3], [0.67, 'partnerships', 'companies', 3], [0.67, 'partnerships', 'co
```

Figure 4.40: Output from similarity function shown in Figure 4.39.

```
def get_concepts(file2, flist, pos, cat):

    pword=''
    simscore = 0
    tally = 0
    first_time = True
    tlist =[]
    occur = 0

    for i in range(0,len (flist)):

        print(i)

        if first_time:
            pword = flist[i][1]
            tally += 1
            simscore += flist[i][0]
            print(pword, tally, simscore)
            occur += flist[i][3]
            first_time = False
        else:
            if pword != flist[i][1]:
               tlist.append([file2, pword, round(simscore/tally,2), occur, pos, cat])
               tally  = 1
               simscore = flist[i][0]
               occur = flist[i][3]
               pword = flist[i][1]
               print('changed', pword)
            else:
               tally += 1
               simscore += flist[i][0]
               occur += flist[i][3]

        if i == len(flist)-1:           ##last entry
               tlist.append([file2, pword, round(simscore/tally,2), occur, pos, cat])
    return tlist
```

Figure 4.41: Python function that condenses word types to concepts.

```
>>> con1
[['f Adelphia 2000', 'areas', 0.62, 28, 'nouns', 'f'], ['f Adelphia 2000', 'cable', 0.57, 14, 'nouns', 'f'], ['f Adelphia 2000', 'chairman', 0.59, 11, 'nouns', 'f'], [
'f Adelphia 2000', 'cities', 0.76, 20, 'nouns', 'f'], ['f Adelphia 2000', 'clusters', 0.65, 10, 'nouns', 'f'], ['f Adelphia 2000', 'colorado', 0.8, 20, 'nouns', 'f'],
['f Adelphia 2000', 'companies', 0.67, 25, 'nouns', 'f'], ['f Adelphia 2000', 'company', 0.67, 43, 'nouns', 'f'], ['f Adelphia 2000', 'core', 0.65, 10, 'nouns', 'f'],
['f Adelphia 2000', 'corporations', 0.68, 21, 'nouns', 'f'], ['f Adelphia 2000', 'data', 0.58, 23, 'nouns', 'f'], ['f Adelphia 2000', 'december', 0.61, 8, 'nouns', 'f'
], ['f Adelphia 2000', 'england', 0.7, 20, 'nouns', 'f'], ['f Adelphia 2000', 'entities', 0.5, 4, 'nouns', 'f'], ['f Adelphia 2000', 'family', 0.63, 37, 'nouns', 'f'],
['f Adelphia 2000', 'florida', 0.8, 20, 'nouns', 'f'], ['f Adelphia 2000', 'front', 0.5, 4, 'nouns', 'f'], ['f Adelphia 2000', 'homes', 0.62, 10, 'nouns', 'f'], ['f Ad
elphia 2000', 'industry', 0.69, 17, 'nouns', 'f'], ['f Adelphia 2000', 'interests', 0.57, 4, 'nouns', 'f'], ['f Adelphia 2000', 'j', 0.6, 3, 'nouns', 'f'], ['f Adelphi
a 2000', 'leader', 0.59, 28, 'nouns', 'f'], ['f Adelphia 2000', 'liability', 0.58, 6, 'nouns', 'f'], ['f Adelphia 2000', 'management', 0.63, 8, 'nouns', 'f'], ['f Adel
phia 2000', 'markets', 0.7, 8, 'nouns', 'f'], ['f Adelphia 2000', 'medium', 0.78, 23, 'nouns', 'f'], ['f Adelphia 2000', 'members', 0.59, 11, 'nouns', 'f'], ['f Adelph
ia 2000', 'networks', 0.53, 47, 'nouns', 'f'], ['f Adelphia 2000', 'officer', 0.57, 11, 'nouns', 'f'], ['f Adelphia 2000', 'ohio', 0.8, 20, 'nouns', 'f'], ['f Adelphia
2000', 'operation', 0.57, 6, 'nouns', 'f'], ['f Adelphia 2000', 'operations', 0.63, 10, 'nouns', 'f'], ['f Adelphia 2000', 'ownership', 0.53, 6, 'nouns', 'f'], ['f Ade
lphia 2000', 'partnerships', 0.72, 21, 'nouns', 'f'], ['f Adelphia 2000', 'pennsylvania', 0.8, 20, 'nouns', 'f'], ['f Adelphia 2000', 'pony', 0.5, 2, 'nouns', 'f'], ['
f Adelphia 2000', 'president', 0.53, 11, 'nouns', 'f'], ['f Adelphia 2000', 'quantities', 0.54, 30, 'nouns', 'f'], ['f Adelphia 2000', 'rigas', 0.71, 42, 'nouns', 'f']
, ['f Adelphia 2000', 'services', 0.64, 10, 'nouns', 'f'], ['f Adelphia 2000', 'signals', 0.58, 14, 'nouns', 'f'], ['f Adelphia 2000', 'springs', 0.64, 6, 'nouns', 'f'
], ['f Adelphia 2000', 'states', 0.83, 20, 'nouns', 'f'], ['f Adelphia 2000', 'subscribers', 0.59, 15, 'nouns', 'f'], ['f Adelphia 2000', 'systems', 0.73, 55, 'nouns',
'f'], ['f Adelphia 2000', 'telecommunications', 0.77, 27, 'nouns', 'f'], ['f Adelphia 2000', 'telephone', 0.71, 21, 'nouns', 'f'], ['f Adelphia 2000', 'television', 0.
76, 33, 'nouns', 'f'], ['f Adelphia 2000', 'video', 0.57, 11, 'nouns', 'f'], ['f Adelphia 2000', 'virginia', 0.8, 20, 'nouns', 'f'], ['f Adelphia 2000', 'voice', 0.67,
2, 'nouns', 'f'], ['f Adelphia 2000', 'way', 0.56, 8, 'nouns', 'f']]
>>> 0.684/11
```

Figure 4.42: An example output from function shown in Figure 4.41.

```
def writetofile(list1):
        with open("C:/data6/wa/tempwrite.txt", "a+") as f:
            for i in list1:
                f.write(("%s\n" % i))
```

Figure 4.43: Each line from output shown in Figure 4.42 is written to a file.

```
['f adelphia 1999', 'areas', 0.63, 798, 'noun', 'f']
['f adelphia 1999', 'arrangements', 0.6, 112, 'noun', 'f']
['f adelphia 1999', 'art', 0.6, 267, 'noun', 'f']
['f adelphia 1999', 'aspects', 0.7, 80, 'noun', 'f']
['f adelphia 1999', 'asset', 0.6, 242, 'noun', 'f']
['f adelphia 1999', 'assets', 0.61, 636, 'noun', 'f']
['f adelphia 1999', 'assistance', 0.67, 356, 'noun', 'f']
['f adelphia 1999', 'associations', 0.65, 692, 'noun', 'f']
['f adelphia 1999', 'assumption', 0.59, 178, 'noun', 'f']
['f adelphia 1999', 'assurance', 0.58, 117, 'noun', 'f']
['f adelphia 1999', 'assurances', 0.55, 113, 'noun', 'f']
['f adelphia 1999', 'atm', 0.59, 17, 'noun', 'f']
['f adelphia 1999', 'attachment', 0.6, 151, 'noun', 'f']
['f adelphia 1999', 'attachments', 0.58, 147, 'noun', 'f']
['f adelphia 1999', 'attempts', 0.67, 354, 'noun', 'f']
['f adelphia 1999', 'auction', 0.57, 301, 'noun', 'f']
['f adelphia 1999', 'authorities', 0.63, 1838, 'noun', 'f']
['f adelphia 1999', 'authority', 0.61, 438, 'noun', 'f']
['f adelphia 1999', 'authorization', 0.81, 145, 'noun', 'f']
['f adelphia 1999', 'authorizations', 0.79, 129, 'noun', 'f']
['f adelphia 1999', 'availability', 0.56, 184, 'noun', 'f']
['f adelphia 1999', 'average', 0.52, 117, 'noun', 'f']
['f adelphia 1999', 'award', 0.58, 321, 'noun', 'f']
['f adelphia 1999', 'awards', 0.57, 319, 'noun', 'f']
['f adelphia 1999', 'b', 0.59, 302, 'noun', 'f']
['f adelphia 1999', 'backbone', 0.51, 37, 'noun', 'f']
['f adelphia 1999', 'balances', 0.59, 124, 'noun', 'f']
['f adelphia 1999', 'ban', 0.71, 116, 'noun', 'f']
['f adelphia 1999', 'band', 0.64, 653, 'noun', 'f']
['f adelphia 1999', 'bands', 0.63, 651, 'noun', 'f']
['f adelphia 1999', 'bandwidth', 0.56, 153, 'noun', 'f']
['f adelphia 1999', 'bank', 0.55, 68, 'noun', 'f']
['f adelphia 1999', 'banks', 0.58, 428, 'noun', 'f']
['f adelphia 1999', 'bargaining', 0.67, 5, 'noun', 'f']
['f adelphia 1999', 'base', 0.62, 334, 'noun', 'f']
['f adelphia 1999', 'basis', 0.55, 579, 'noun', 'f']
['f adelphia 1999', 'baton', 0.63, 169, 'noun', 'f']
['f adelphia 1999', 'beginning', 0.6, 346, 'noun', 'f']
['f adelphia 1999', 'bell', 0.66, 251, 'noun', 'f']
['f adelphia 1999', 'benchmark', 0.61, 104, 'noun', 'f']
['f adelphia 1999', 'benchmarks', 0.58, 88, 'noun', 'f']
['f adelphia 1999', 'benefit', 0.7, 208, 'noun', 'f']
['f adelphia 1999', 'benefits', 0.68, 189, 'noun', 'f']
```

Figure 4.44: Concepts written to file.

f adelphia 1999	ability-noun	0.3309	f
f adelphia 1999	absence-noun	0.0454	f
f adelphia 1999	accelerate-verb	0.0436	f
f adelphia 1999	access-noun	0.2983	f
f adelphia 1999	accommodate-verb	0.0183	f
f adelphia 1999	accordance-noun	0.2183	f
f adelphia 1999	according-verb	0.0007	f
f adelphia 1999	accounted-verb	0.0058	f
f adelphia 1999	accounting-noun	0.156	f
f adelphia 1999	account-noun	0.0183	f
f adelphia 1999	accounts-noun	0.0134	f
f adelphia 1999	account-verb	0.0071	f
f adelphia 1999	accretion-noun	0.0222	f
f adelphia 1999	acquired-verb	1.255	f
f adelphia 1999	acquire-verb	0.3409	f
f adelphia 1999	acquiring-verb	0.3303	f
f adelphia 1999	acquisition-noun	0.7012	f
f adelphia 1999	acquisitions-noun	1.3596	f
f adelphia 1999	action-noun	0.2353	f
f adelphia 1999	actions-noun	0.3466	f
f adelphia 1999	activating-verb	0.0014	f
f adelphia 1999	activities-noun	0.5819	f
f adelphia 1999	activity-noun	0.1904	f
f adelphia 1999	act-noun	0.9068	f
f adelphia 1999	acts-verb	0.2409	f
f adelphia 1999	act-verb	0.3714	f
f adelphia 1999	added-verb	0.0553	f
f adelphia 1999	adding-verb	0.0519	f

Figure 4.45: Final concepts after manipulation in Excel.

```
f=open("C:/data6/wa/concepts/allconcepts.csv", 'r')
###all sentences
s1 = set()   # file
s2 = set()  # concepts
list1 = []
for line in f:
    line = line.strip(' \t\n\r')
    line = line.split(",")
    s1.add(line[0])
    s2.add(line[1].strip('\t'))
    list1.append([line[0],line[1].strip('\t'), line[2].strip('\t'), line[3]])
f.close()
## contains all files s1
s1 = list(s1)
s1.sort()
## contains all concepts
s2 = list(s2)
s2.sort()
s2.insert(0, 'Col Header')
s2.insert(len(s2), 'class')
## header info to append to file (would have concepts as hdrs)
a1 = []
a1.append(s2[0:len(s2)])
with open("C:/data6/wa/concepts/foo2.txt", "a+") as f:
        f.write(("%s\n" % a1[0]))
#get row info, each file and concept count
a2 = []
list2 = []
for i in s1:
    list2 = [0]*len(s2)
    list2[0] = i    ## add header
    for i2 in range(len(list1)):
                if list1[i2][0] == i:
                    index1 = s2.index (list1[i2][1])
                    list2 [index1] =  list1[i2][2]
                    list2[len(list2)-1]= list1[i2][3]
    a2.append(list2[0:len(list2)])
def writetofile(list1):
        with open("C:/data6/wa/concepts/foo2.txt", "a+") as f:
            for i in list1:
                f.write(("%s\n" % i))
```

Figure 4.46 Code that forms the final matrix from data as was shown in Figure 4.45.

File|Concept	'abandon-verb'	'abandoned-verb'	bandonment-no	andonments-n	'abbreviated-verb'	'abdominal-noun'	'abilities-noun'	ıbility-nou
f ChinaMedia Express Holdings 2008	0	0	0	0	0	0	'0.2357'	'1.4488'
f ChinaMedia Express Holdings 2009	0	0	0	0	0	0	0	'0.8184'
f Global Crossing 2000	0	0	'0.2866'	0	0	0	0	'0.6189'
f Global Crossing 2002	'0.0037'	'0.0027'	'0.4208'	'0.3417'	0	0	0	'0.8449'
f HANSEN MEDICAL INC.2008	0	0	0	0	0	0	0	'0.6922'
f HANSEN MEDICAL INC.2009	'0.0006'	0	0	0	0	'0.0102'	0	'0.9814'
f adelphia 1999	0	0	0	0	0	0	0	'0.3309'
f adelphia 2000	0	0	0	0	0	0	0	0
f worldcom 2001	0	0	0	0	0	0	0	0
nf 3 D Corporation 2000	0	0	0	0	0	0	0	'0.8539'
nf 3 D Corporation 2001	0	0	0	0	0	0	0	0
nf ACCENTURE PLC 2009	0	0	0	0	0	0	0	'1.1968'
nf ACCENTURE PLC 2010	0	0	0	0	0	0	0	'1.4602'
nf BROOKDALE SENIOR LIVING INC. 2008	0	0	'0.3326'	0	0	0	'0.1986'	'1.4431'
nf BROOKDALE SENIOR LIVING INC. 2009	0	0	'0.3259'	0	0	0	'0.1964'	'1.3846'
nf CONTINENTAL RESOURCES INC. 2010	0	0	'0.3264'	0	0	0	0	'0.8638'
nf CONTINENTAL RESOURCES INC. 2011	0	0	'0.1551'	0	0	0	0	'0.5'
nf Evans Bancorp Inc 2009	0	0	0	0	0	0	0	'0.3707'
nf Level 3 communications 2000	0	0	0	0	0	0	0	'0.1919'
nf Level 3 communications 2002	'0.0013'	0	0	0	0	0	0	'0.3131'
nf SCHERING-PLOUGH CORPORATION 2008	0	0	0	0	'0.0127'	0	0	'0.5108'
nf SCHERING-PLOUGH CORPORATION 2009	'0.0007'	0	'0.1569'	0	0	0	0	'0.1469'

Figure 4.47: The final matrix structure as will be passed to downstream processes.

	Topics
Matched Pair	204 rows (102 'f' reports) and (102 'nf' reports) by 5595 concepts
Peer Set	408 rows (102 'f' reports) and (308 'nf' reports) by 6947 concepts

Table 4.9: Dimensions of constructed matrices.

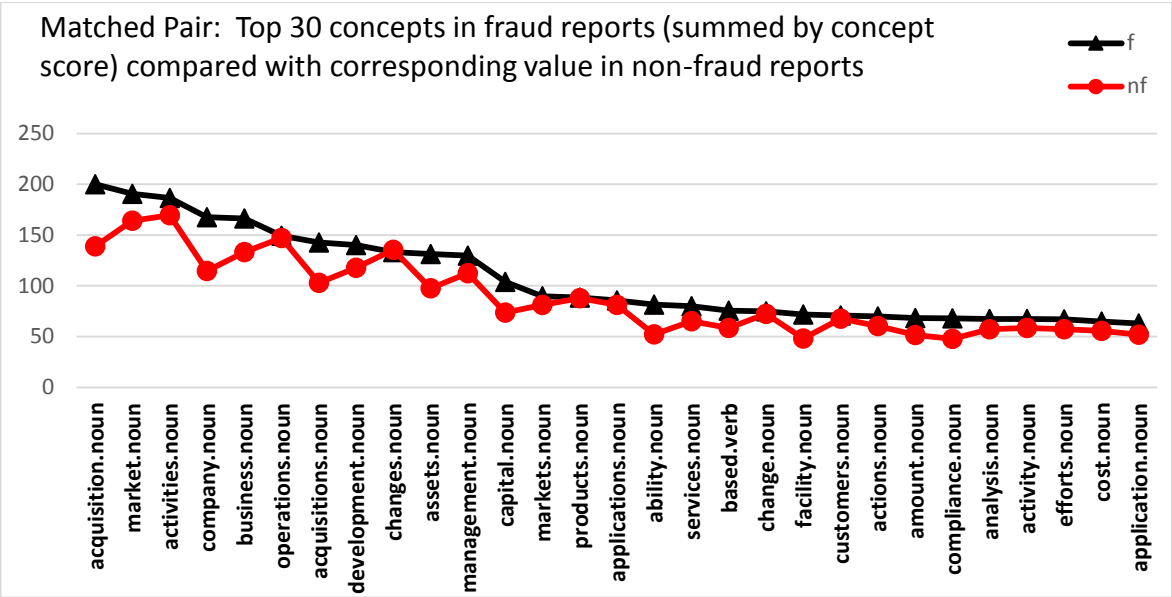


Figure 4.48: Top 30 concepts extracted from matched pair matrix constructed for concepts in fraud reports matched with corresponding concept in non-fraud reports.

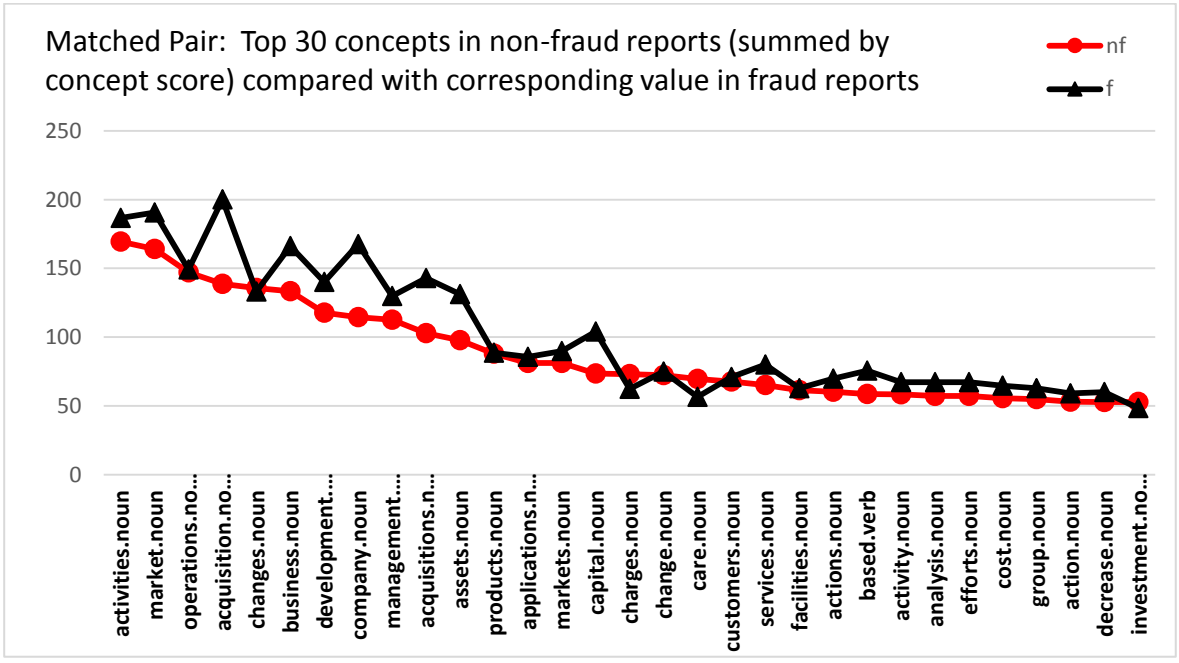


Figure 4.49: Top 30 concepts extracted from matched pair matrix constructed for concepts in non-fraud reports matched with corresponding concept in fraud reports.

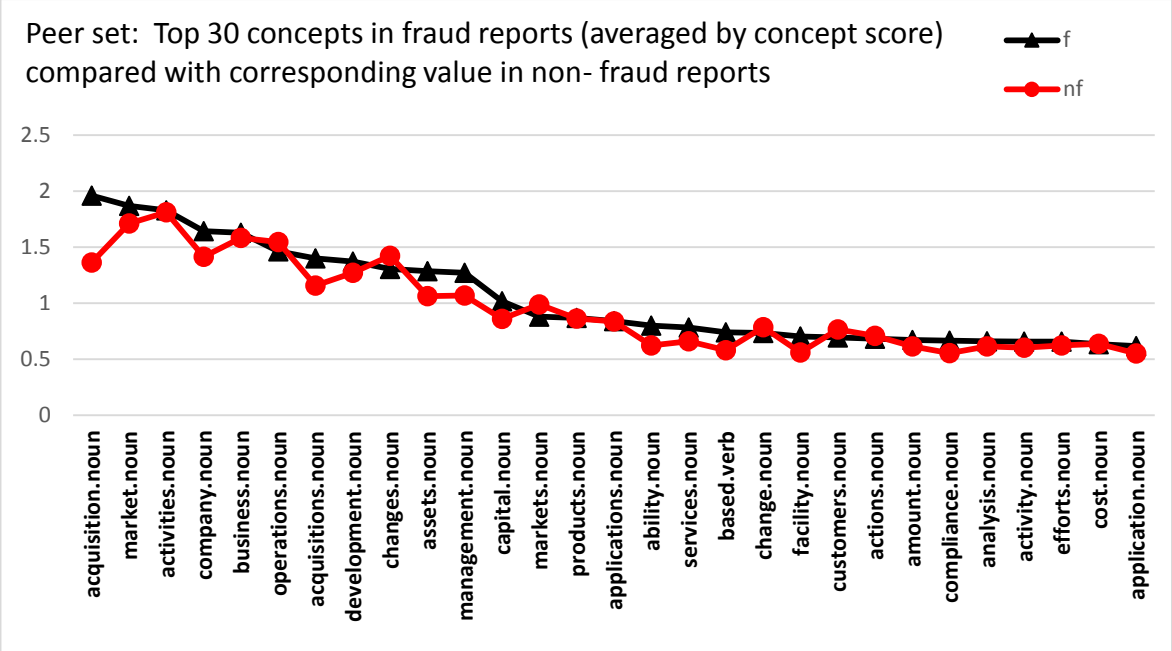


Figure 4.50: Top 30 concepts extracted from peer set matrix constructed for concepts in fraud reports matched with corresponding concept in non- fraud reports.

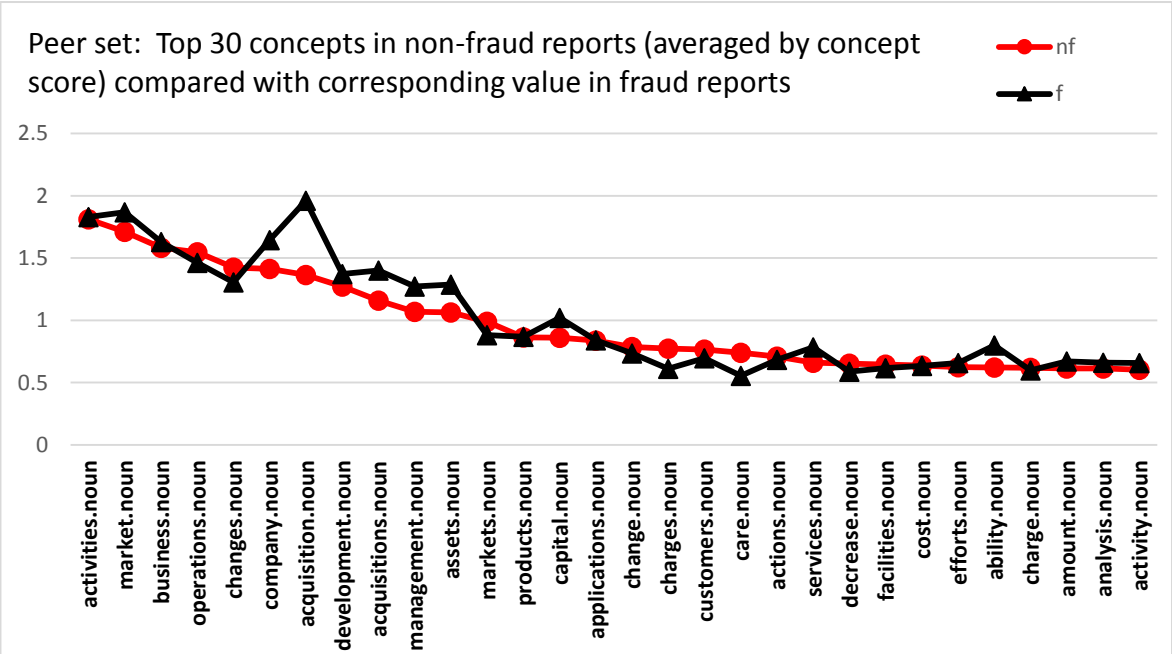


Figure 4.51: Top 30 concepts extracted from peer set matrix constructed for concepts in non-fraud reports matched with corresponding concept in fraud reports.

4.11 Keywords from Corpus Analysis

As was communicated in chapter 3, a cornerstone technique used to investigate language in the nascent discipline of corpus linguistics is keyword analysis. Words that were denoted as key using this techniques described in chapter 3 are now to be used as features to be passed to the classifiers to aid in the discrimination task. A selection of top 300 words that were found to be key in the fraud reports and non-fraud reports (Appendix D, Table D.4 and D.5) were used.

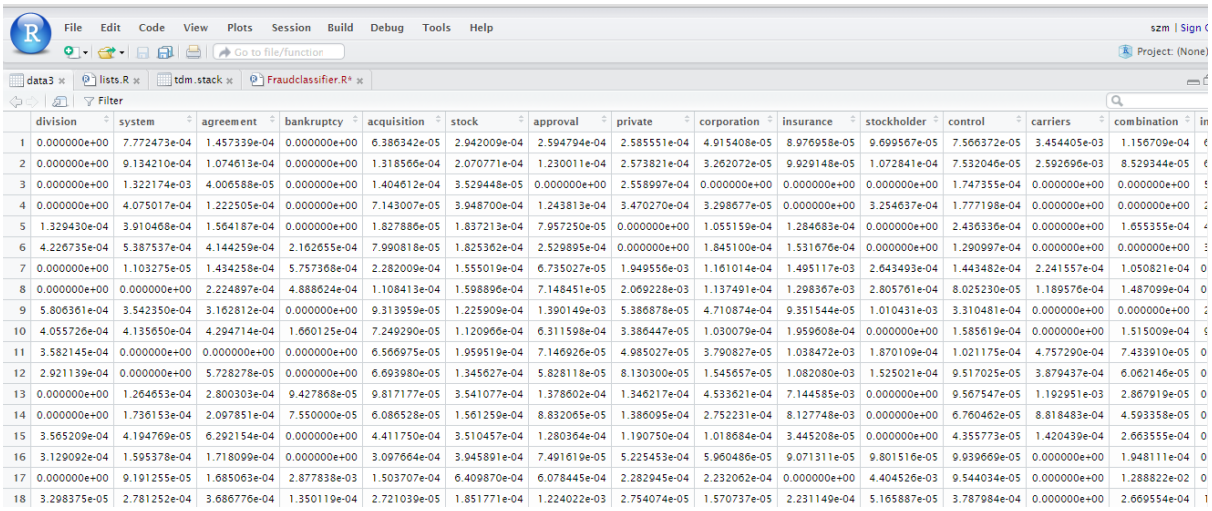
To set up the matrix for the corpus that can be passed to the classifiers the following process was undertaken:-

1. A term document matrix (tdm.stack) based on tf-idf scores for all word types (unigrams) in corpus was generated using the program extracts shown in Figures 4.6 and 4.7. However this time no stemming was performed to enable pick up of keywords from corpus. This resulted in a matrix 408 rows long and 2022 columns wide.

2. The r command subset (tdm.stack, select = c(keywords)) was executed over the corpus. This enabled the filtering out of the keywords (identified in chapter 3 as keywords) from the unigrams term document matrix. However only 250 keywords were found to exist in the matrix. The 250 words selected are shown in Appendix H, Table H.8. Therefore, the new matrix, based on keywords discovered through keyword analysis, was made up of 250 columns wide and 408 rows long. An excerpt from this matrix is shown in Figure 4.52. The matrix is now ready to go through feature selection algorithms, as will be outlined in chapter 5 and then will go through classifiers in chapter 6.

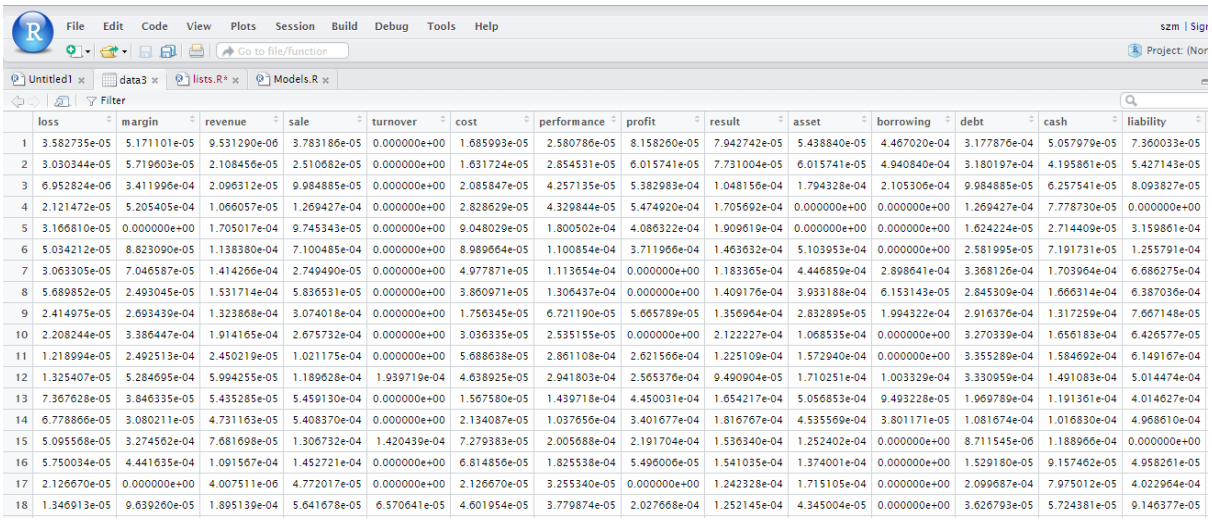
The other keywords that were selected to be differentiators between fraud and non-fraud reports were the words uncovered by Rutherford [138] through again using techniques from Corpus Linguistics toolkit. Again there were some words that are shown in Appendix E from the Rutherford study that were not found in the corpus. Only 70 were found in the tdm matrix, setup as described in section 4.3. These 70 words are shown in Appendix H (Table H.9). An excerpt from the matrix formed from these words is shown Figure 4.53. This matrix will also be put through feature selection to enable better generalisation of classifiers as will be described in chapter 5 and 6.

The dimensions of matrices that will be input to feature selection routines are shown in Table 4.7 and Table 4.8.



	division	system	agreement	bankruptcy	acquisition	stock	approval	private	corporation	insurance	stockholder	control	carriers	combination
1	0.000000e+00	7.772473e-04	1.457339e-04	0.000000e+00	6.386342e-05	2.942009e-04	2.594794e-04	2.585551e-04	4.915408e-05	8.976958e-05	9.699567e-05	7.566372e-05	3.454405e-03	1.156709e-04
2	0.000000e+00	9.134210e-04	1.074613e-04	0.000000e+00	1.318566e-04	2.070771e-04	1.230011e-04	2.573821e-04	3.262072e-05	9.929148e-05	1.072841e-04	7.532046e-05	2.592696e-03	8.529344e-05
3	0.000000e+00	1.322174e-03	4.006588e-05	0.000000e+00	1.404612e-04	3.529448e-05	0.000000e+00	2.558997e-04	0.000000e+00	0.000000e+00	0.000000e+00	1.747355e-04	0.000000e+00	0.000000e+00
4	0.000000e+00	4.075017e-04	1.222505e-04	0.000000e+00	7.143007e-05	3.948700e-04	1.243813e-04	3.470270e-04	3.298677e-05	0.000000e+00	3.254637e-04	1.777198e-04	0.000000e+00	0.000000e+00
5	1.329430e-04	3.910468e-04	1.564187e-04	0.000000e+00	1.827886e-05	1.837213e-04	7.957250e-05	0.000000e+00	1.055159e-04	1.284683e-04	0.000000e+00	2.436336e-04	0.000000e+00	1.655355e-04
6	4.226735e-04	5.387537e-04	4.144259e-04	2.162655e-04	7.990818e-05	1.825362e-04	2.529895e-04	0.000000e+00	1.845100e-04	1.531676e-04	0.000000e+00	1.290997e-04	0.000000e+00	0.000000e+00
7	0.000000e+00	1.103275e-05	1.434258e-04	5.757368e-04	2.282009e-04	1.555019e-04	6.735027e-05	1.949556e-03	1.161014e-04	1.495117e-03	2.643493e-04	1.443482e-04	2.241557e-04	1.050821e-04
8	0.000000e+00	0.000000e+00	2.224897e-04	4.888624e-04	1.108413e-04	1.598896e-04	7.148451e-05	2.069228e-03	1.137491e-04	1.298367e-03	2.805761e-04	8.025230e-05	1.189576e-04	1.487099e-04
9	5.806361e-04	3.542350e-04	3.162812e-04	0.000000e+00	9.313959e-05	1.225909e-04	1.390149e-03	5.386878e-05	4.710874e-04	9.351544e-05	1.010431e-03	3.310481e-04	0.000000e+00	0.000000e+00
10	4.055726e-04	4.135650e-04	4.294714e-04	1.660125e-04	7.249290e-05	1.120966e-04	6.311598e-04	3.386447e-05	1.030079e-04	1.959608e-04	0.000000e+00	1.585619e-04	0.000000e+00	1.515009e-04
11	3.582145e-04	0.000000e+00	0.000000e+00	0.000000e+00	6.566975e-05	1.959519e-04	7.146926e-05	4.985027e-05	3.790827e-05	1.038472e-03	1.870109e-04	1.021175e-04	4.757290e-04	7.433910e-05
12	2.921139e-04	0.000000e+00	5.728278e-05	0.000000e+00	6.693980e-05	1.345627e-04	5.828118e-05	8.130300e-05	1.545657e-05	1.082080e-03	1.525021e-04	9.517025e-05	3.879437e-04	6.062146e-05
13	0.000000e+00	1.264653e-04	2.800303e-04	9.427868e-05	9.817177e-05	3.541077e-04	1.378602e-04	1.346217e-04	4.533621e-04	7.144585e-03	0.000000e+00	9.567547e-05	1.192951e-03	2.867919e-05
14	0.000000e+00	1.736153e-04	2.097851e-04	7.550000e-05	6.086528e-05	1.561259e-04	8.832065e-05	1.386095e-04	2.752231e-04	8.127748e-03	0.000000e+00	6.760462e-05	8.818483e-04	4.593358e-05
15	3.565209e-04	4.194769e-05	6.292154e-04	0.000000e+00	4.411750e-04	3.510457e-04	1.280364e-04	1.190750e-04	1.018684e-04	3.445208e-05	0.000000e+00	4.355773e-05	1.420439e-04	2.663555e-04
16	3.129092e-04	1.595378e-04	1.718099e-04	0.000000e+00	3.097664e-04	3.945891e-04	7.491619e-05	5.225453e-04	5.960486e-05	9.071311e-05	9.801516e-05	9.939669e-05	0.000000e+00	1.948111e-04
17	0.000000e+00	9.191255e-05	1.685063e-04	2.877838e-03	1.503707e-04	6.409870e-04	6.078445e-04	2.282945e-04	2.232062e-04	0.000000e+00	4.404526e-03	9.544034e-05	0.000000e+00	1.288822e-02
18	3.298375e-05	2.781252e-04	3.686776e-04	1.350119e-04	2.721039e-05	1.851771e-04	1.224022e-03	2.754074e-05	1.570737e-05	2.231149e-04	5.165887e-05	3.787984e-04	0.000000e+00	2.669554e-04

Figure 4.52: Matrix constructed from keywords.



	loss	margin	revenue	sale	turnover	cost	performance	profit	result	asset	borrowing	debt	cash	liability
1	3.582735e-05	5.171101e-05	9.531290e-06	3.783186e-05	0.000000e+00	1.685993e-05	2.580786e-05	8.158260e-05	7.942742e-05	5.438840e-05	4.467020e-04	3.177876e-04	5.057979e-05	7.360033e-05
2	3.030344e-05	5.719603e-05	2.108456e-05	2.510682e-05	0.000000e+00	1.631724e-05	2.854531e-05	6.015741e-05	7.731004e-05	6.015741e-05	4.940840e-04	3.180197e-04	4.195861e-05	5.427143e-05
3	6.952824e-06	3.411996e-04	2.096312e-05	9.984885e-05	0.000000e+00	2.085847e-05	4.257135e-05	5.382983e-04	1.048156e-04	1.794328e-04	2.105306e-04	9.984885e-05	6.257541e-05	8.093827e-05
4	2.121472e-05	5.205405e-04	1.066057e-05	1.269427e-04	0.000000e+00	2.828629e-05	4.329844e-05	5.474920e-04	1.705692e-04	0.000000e+00	0.000000e+00	1.269427e-04	7.778730e-05	0.000000e+00
5	3.166810e-05	0.000000e+00	1.705017e-04	9.745343e-05	0.000000e+00	9.048029e-05	1.800502e-04	4.086322e-04	1.909619e-04	0.000000e+00	0.000000e+00	1.624224e-05	2.714409e-05	3.159861e-04
6	5.034212e-05	8.823090e-05	1.138380e-04	7.100485e-04	0.000000e+00	8.989664e-05	1.100854e-04	3.711966e-04	1.463632e-04	5.103953e-04	0.000000e+00	2.581995e-05	7.191731e-05	1.255791e-04
7	3.063305e-05	7.046587e-05	1.414266e-04	2.749490e-05	0.000000e+00	4.977871e-05	1.113654e-04	0.000000e+00	1.183365e-04	4.446859e-04	2.898641e-04	3.368126e-04	1.703964e-04	6.686275e-04
8	5.689852e-05	2.493045e-05	1.531714e-04	5.836531e-05	0.000000e+00	3.860971e-05	1.306437e-04	0.000000e+00	1.409176e-04	3.933188e-04	6.153143e-05	2.845309e-04	1.666314e-04	6.387036e-04
9	2.414975e-05	2.693439e-04	1.323868e-04	3.074018e-04	0.000000e+00	1.756345e-05	6.721190e-05	5.665789e-05	1.356964e-04	2.832895e-05	1.994322e-04	2.916376e-04	1.317259e-04	7.667148e-05
10	2.208244e-05	3.386447e-04	1.914165e-04	2.675732e-04	0.000000e+00	3.036335e-05	2.535155e-05	0.000000e+00	2.122227e-04	1.068535e-04	0.000000e+00	3.270339e-04	1.656183e-04	6.426577e-05
11	1.218994e-05	2.492513e-04	2.450219e-05	1.021175e-04	0.000000e+00	5.688638e-05	2.861108e-04	2.621566e-04	1.225109e-04	1.572940e-04	0.000000e+00	3.355289e-04	1.584692e-04	6.149167e-04
12	1.325407e-05	5.284695e-04	5.994255e-05	1.189628e-04	1.939719e-04	4.638925e-05	2.941803e-04	2.565376e-04	9.490904e-05	1.710251e-04	1.003329e-04	3.330959e-04	1.491083e-04	5.014474e-04
13	7.367628e-05	3.846335e-05	5.435285e-05	5.459130e-04	0.000000e+00	1.567580e-05	1.439718e-04	4.450031e-04	1.654217e-04	5.056853e-04	9.493228e-05	1.969789e-04	1.191361e-04	4.014627e-04
14	6.778866e-05	3.080211e-05	4.731163e-05	5.408370e-04	0.000000e+00	2.134087e-05	1.037656e-04	3.401677e-04	1.816767e-04	4.535569e-04	3.801171e-05	1.081674e-04	1.016830e-04	4.968610e-04
15	5.095568e-05	3.274562e-04	7.681698e-05	1.306732e-04	1.420439e-04	7.279383e-05	2.005688e-04	2.191704e-04	1.536340e-04	1.252402e-04	0.000000e+00	8.711545e-06	1.188966e-04	0.000000e+00
16	5.750034e-05	4.441635e-04	1.091567e-04	1.452721e-04	0.000000e+00	6.814856e-05	1.825538e-04	5.496006e-05	1.541035e-04	1.374001e-04	0.000000e+00	1.529180e-05	9.157462e-05	4.958261e-05
17	2.126670e-05	0.000000e+00	4.007511e-06	4.772017e-05	0.000000e+00	2.126670e-05	3.255340e-05	0.000000e+00	1.242328e-04	1.715105e-04	0.000000e+00	2.099687e-04	7.975012e-05	4.022964e-04
18	1.346913e-05	9.639260e-05	1.895139e-04	5.641678e-05	6.570641e-05	4.601954e-05	3.779874e-05	2.027668e-04	1.252145e-04	4.345004e-05	0.000000e+00	3.626793e-05	5.724381e-05	9.146377e-05

Figure 4.53: Matrix constructed from Rutherford [199] keywords.

	Keywords
Matched Pair	204 rows (102 ‘f’ reports) and (102 ‘nf’ reports) by 250 keywords
Peer Set	408 rows (102 ‘f’ reports) and (308 ‘nf’ reports) by 250 keywords

Table 4.9: Dimensions of constructed matrices for keywords

	Keywords - Rutherford
Matched Pair	204 rows (102 ‘f’ reports) and (102 ‘nf’ reports) by 80 keywords
Peer Set	408 rows (102 ‘f’ reports) and (308 ‘nf’ reports) by 80 keywords

Table 4.10: Dimensions of constructed matrices for keywords [199].

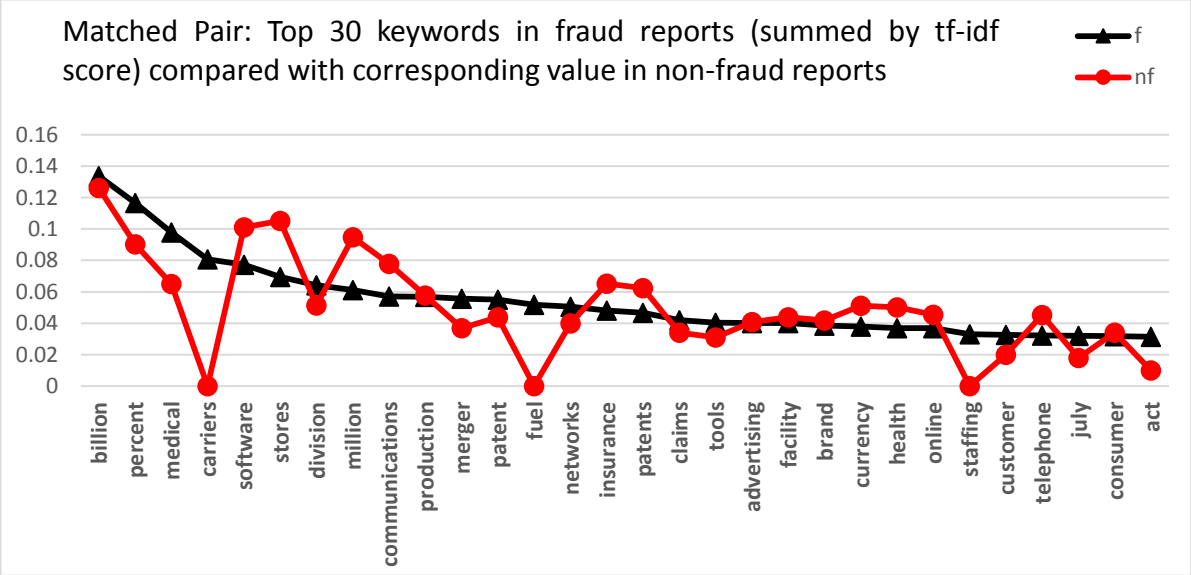


Figure 4.54: Keywords in fraud reports plotted with corresponding tf-idf score for non-fraud reports (matched pair).

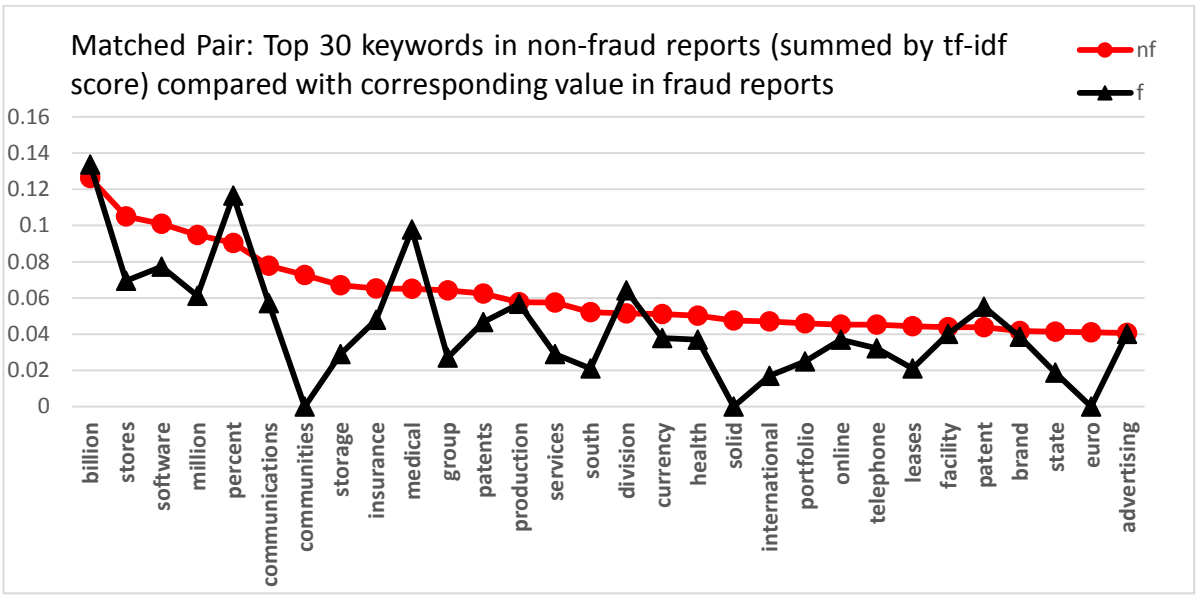


Figure 4.55: Keywords in non-fraud reports plotted with corresponding tf-idf score for fraud reports (matched pair).

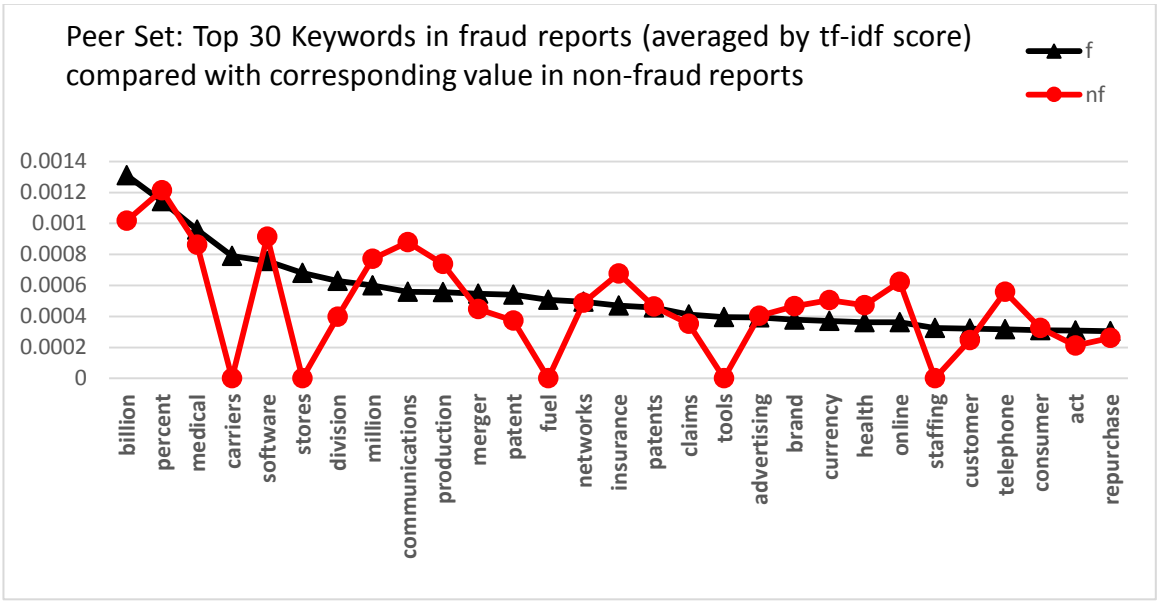


Figure 4.56: Keywords in fraud reports plotted with corresponding tf-idf score for non-fraud reports (peer set).

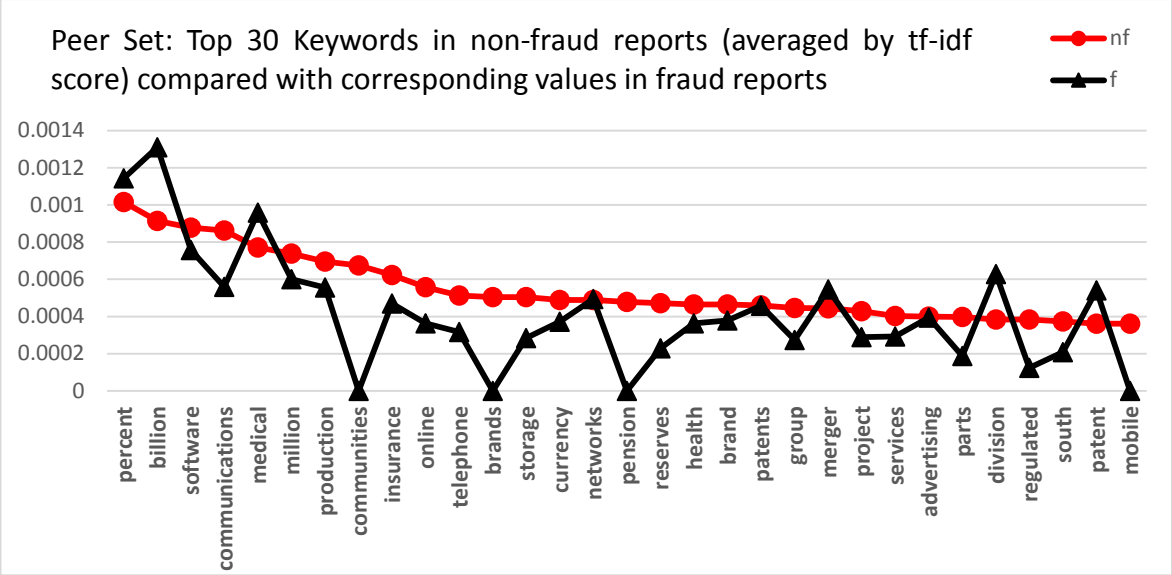


Figure 4.57: Keywords in non-fraud reports plotted with corresponding tf-idf score for fraud reports (peer set).

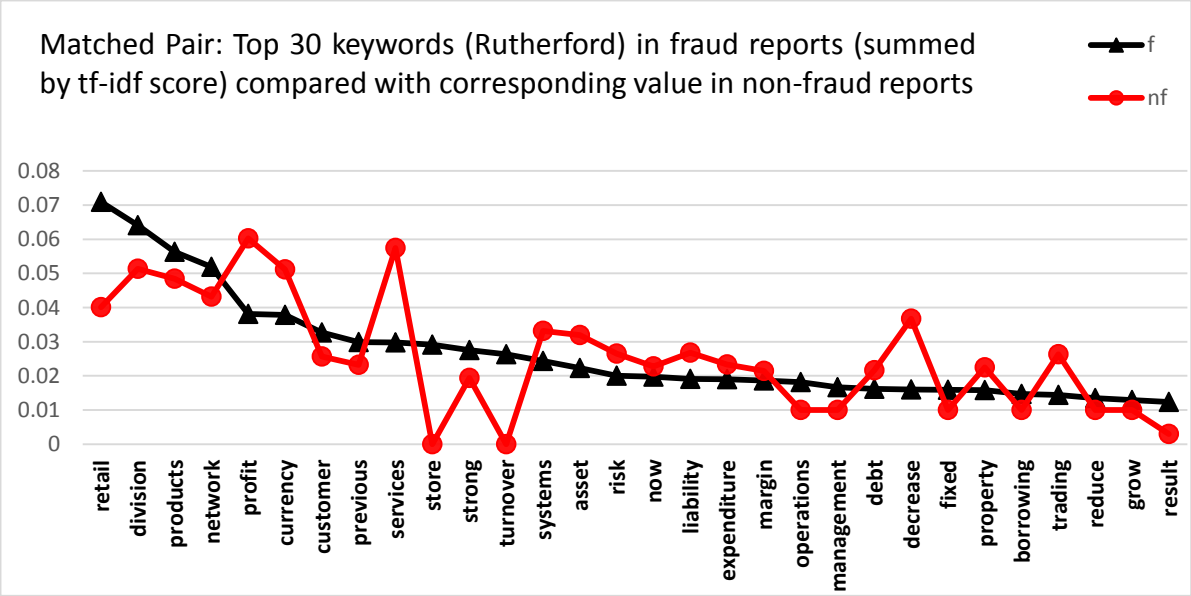


Figure 4.58: Keywords (Rutherford [199]) in fraud reports plotted with corresponding tf-idf score for non-fraud reports (matched pair).

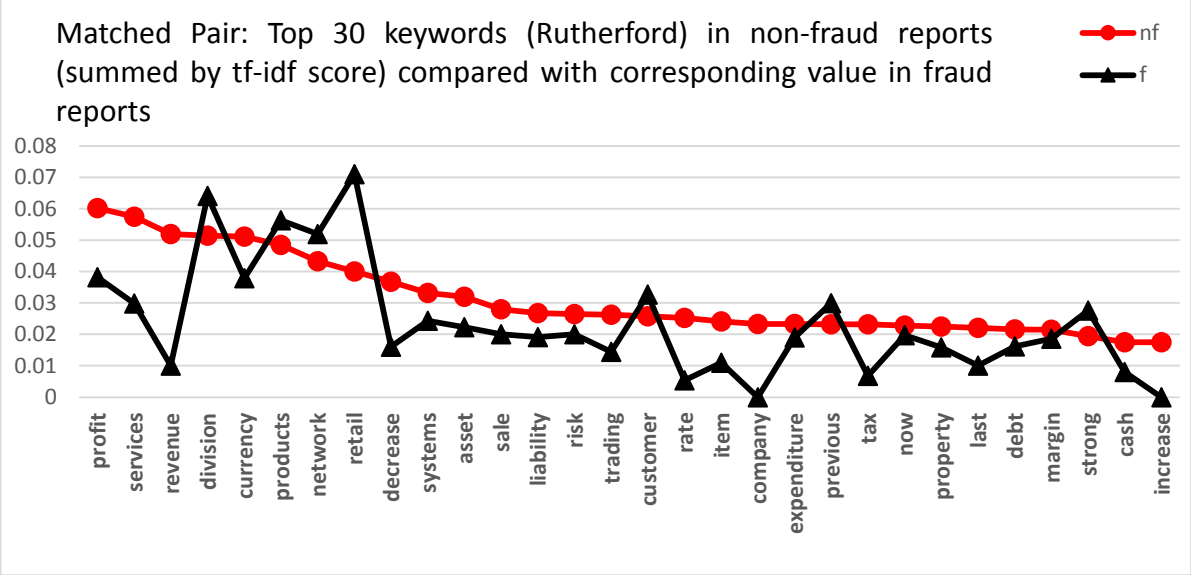


Figure 4.59: Keywords (Rutherford [199]) in non-fraud reports plotted with corresponding tf-idf score for fraud reports (matched pair).

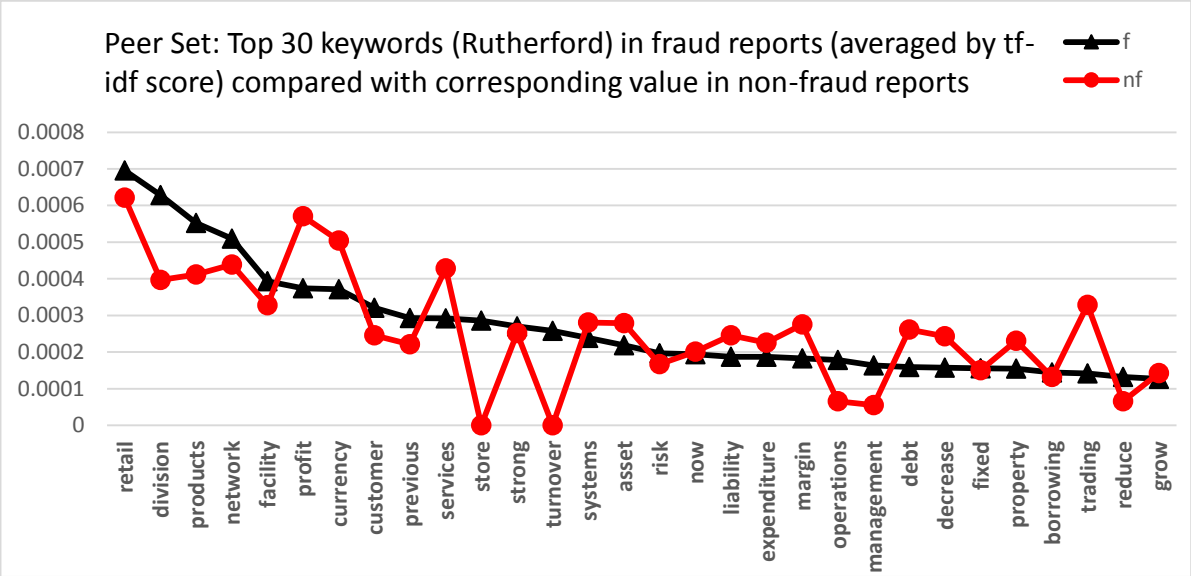


Figure 4.60: Keywords (Rutherford [199]) in fraud reports plotted with corresponding tf-idf score for non-fraud reports (peer set).

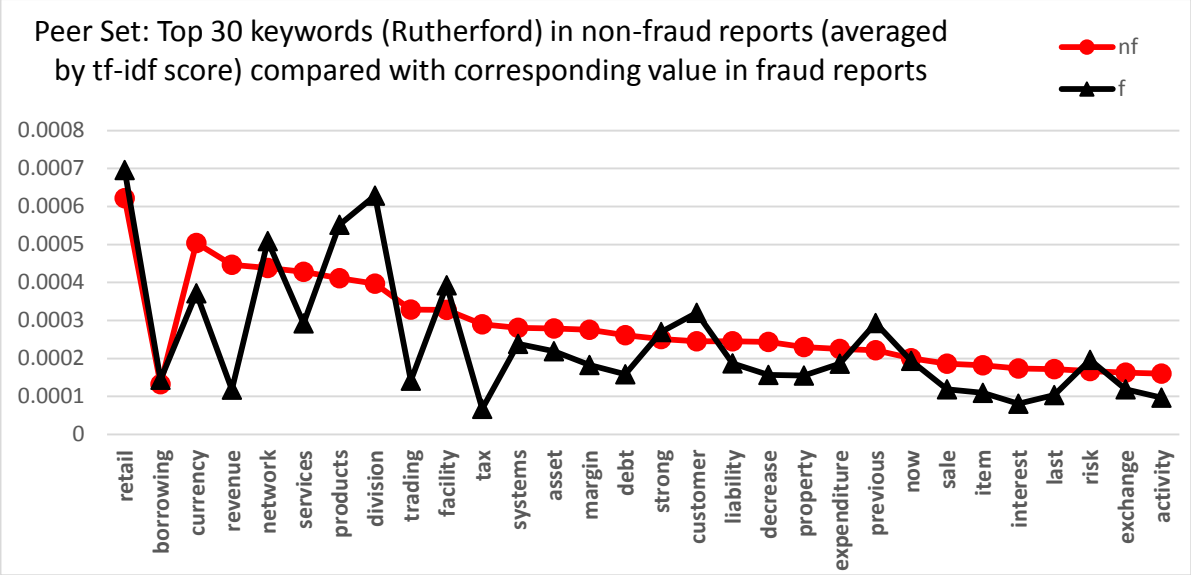


Figure 4.61: Keywords (Rutherford [199]) in non-fraud reports plotted with corresponding tf-idf score for fraud reports (peer set).

Figure 4.54 to Figure 4.61 show the keywords (those shown in figure 3.6 and figure 3.7) and keywords (Rutherford) – shown in figure 3.8 that were the most distinct in the fraud and non-fraud report categories based on both the peer set and matched pair data compositions. It can be seen from the graphs that based on these keywords a clear difference emerges between the two categories.

4.12 Discussion and Conclusion

In order to handle the vast swathes of textual data that abounds in every domain but notably here in the financial domain, ways to reduce this to manageable portions is required. A way forward is to investigate approaches to represent the object of interest, the document or as is the case here the annual reports/10k. The central question addressed is: "*How do we effectively represent annual report/10K to enable a detection of differences, if any in linguistic patterns*?"

The dominant method in information retrieval that based on vector space modelling using the bag of words approach was first examined. Both the query and the document are reduced to bag of words and similarity measures such as Euclidean and cosines are deployed to determine if there are overlaps and ascertain if the document is

relevant to the query. As noted although widely used, it loses much of the structure in language that enables messages to be fully conveyed. The word order is lost and thus different sentences have exactly the same representation as long as the same words are used. As Clark [22] puts it: "*much of the meaning of a document is mediated by* the order of the words, and the syntactic structures of the sentences". However as noted by Clark [22] such a scheme for NLP tasks have worked surprisingly well and attempts to exploit linguistic structure beyond the word level have not usually improved performance. For the document retrieval problem in search engine applications this is unsurprising, since queries are usually a few words and so describing the problem as one of simple word matching between query and document is appropriate. However, in order to move on from this status quo, alternatives that comb the text at a finer level need to be examined. More so in areas such as deception detection in text as is the case here. In this chapter several such alternatives were depicted.

Sinclair [208] maintains that language is 70% formulaic, there is less variability in its use that would be garnered from the term popularised by Chomsky [16] that language is: "*infinite use of finite means*". Written and spoken language composition has been compared to stitching a quilt together, the patches being pre-constructed phrases [122].This would be especially true when examining a particular genre of text such as financial text, where content, structure of discourse and linguistic style would be similar. Therefore, any difference in key constructs of language like multiword expressions could be significant. In this study, multiword expressions such as bigrams and trigrams are picked up from the corpus to aid in fraud detection. Both bigram and trigrams were generated for the corpus, as can be seen from the matrices produced in Figure 4.14 and 4.15 there is high sparsity and high dimensionality. This left unchecked reduces classifier performance and results in overfitting (to be discussed in Chapter 6). If four n-grams had also been used this problem would have been exacerbated. However, as stated n-grams do attempt to pick up more context and meaning than is the case with unigrams.

The application of Coh-Metrix a state of the art, robust and widely used NLP tool to examine text at a deeper level for readability represents a leap forward into examining this construct in the financial domain [50]. Previous studies have used simplistic readability formulas that although useful do not go the distance in examining aspects of cohesion and coherence that affect interpretation and comprehension of the text by the reader (Merkl-Davies [30] show a list of research in this area). Still further studies

that have extracted indicators on readability [103] but have not used tools that are robust at extracting complex linguistic features as those encapsulated by the Coh-Metrix indices. The appropriateness to monitor readability in text that is deemed or suspected to be deceptive has been elaborated by Merkl-Davies [30]. The management obfuscation hypothesis [79] incomplete revelation hypothesis [80] and plethora of other research, shown in chapter 2 indicate that management are given to manipulating readability to hide/deflect poor performance. Therefore it is a central construct that needs to be deeply examined to sift out those who may use it to engage in falsification. This is the first study that looks at readability from the lens of deception. The question addressed is: firms who are known to have committed fraud, are their annual reports/10K less readable than similar non-fraud firms? To answer this question Coh Metrix indices that probe the text deeper, looking at phraseology, lexical diversity, linkages and others as expanded on earlier are extracted for each of the 408 reports and a matrix set up. This is potentially a very revealing matrix as it has penetrated and probed the text deeper to arrive at a deeper appreciation of the composition of the text.

The new release of LIWC (2015) has resulted in this study being the first to use the more updated and expanded dictionaries that are at the heart of its operations. For each text file 73 output variables are written as one line of data to a matrix. Pennebaker et al. [231] argues that the ways people use words in their daily lives can provide rich information about their beliefs, fears and thinking patterns. Words are put into categories which provides clear sight of dominant categories in the report. The matrix once constructed would contain features that seek to provide meaning behind the choice of words used and be revealing once passed the classifiers.

To counteract the general purpose nature of the internal LIWC dictionaries, word lists designed for the financial domain replaced the internal dictionaries in LIWC. They were chosen on the basis of previous studies [88, 89, 251] which maintained that tone in financial disclosure has direct impact investment decisions, for example Tetlock [89] showed that pessimism has a significant downward pressure on prices of the stock indices. Further previous deception research (delineated in chapter 2) also showed that tone is different between truth-tellers and liars. Therefore, it is apt to measure tone using words that were specifically designed for the financial domain. Good financial reporting involves giving information on future plans. In order to enable further differentiation between a fraud (bad) and a non-fraud (good) firm, a forward looking

word list is used to determine if it could aid in the discrimination task at hand. Similarly, it is known from Rezzae and Riley's [2] seminal work on financial statement fraud that financial constraints such as liquidity issues, near bankruptcy push top management to falsification in reporting. In order to measure any traces of such difficulties, risk based word list are deployed to aid in the pick-up of such concerns in the text. Again a matrix is formed that encapsulates rich information garnered from text on the state of a firm.

To directly pick up the cues to deception that have been developed by previous research, epitomised by Zhou et al. [123] the linguistic cues shown in Table 4.6 were extracted. The ratios fit well into the feature engineering outlook encouraged in machine learning (to be covered in chapter 6) that reduces correlation amongst features and enables better generalisation [215]. So the matrix is fully operationalised both from a deception viewpoint and from a machine learning perspective to tackle the classification task of differentiating a fraud from a non-fraud firm.

In an attempt to unlock content of annual reports/10K, in other words to gain a better understanding of what is being said topic modelling techniques such as Latent Dirichlet Allocation is applied over the corpus. This again is new ground that is covered, in terms of the application of this technique over this corpus. As described in this chapter, it is robust and is in widespread use to garner an appreciation of content in huge corpora [242]. At its heart though it is again based on a bag of words approach to natural language. This as pointed out omits a degree of meaning that is intended to be conveyed in the text and again attempts to improve on the basic LDA algorithm as described in this chapter have not led to many gains.

As language is scattered with words that have similar meaning and in attempt to get closer to message that is conveyed in the reports, a concept mining approach was also used over the corpus. This took every word in a report and compared it to every other word with the same part of speech and determined similarity using WordNet. A measure was taken to indicate the strength of its presence in the text through this comparison. In this way a condensed set of concepts that are key in the text was sought. The resultant matrix is large due to sparsity but once through sparsity reduction, it is considerably reduced. This again is a hitherto unused method to attempt to unlock content in annual reports and once through the classifier should prove to be revealing in this discrimination task at hand.

Finally, the keywords from corpus analysis was used as features to represent the reports in the corpus. As indicated, some of the keywords had been dropped through the pre-processing and sparsity reduction steps in using the unigrams bag of words model. This is the model that contains all words in the corpus and now in this step it contained only some of the keywords uncovered using the corpus linguistics toolkit. As these keywords are already known to be markers of differentiation between fraud and non-fraud reports it is anticipated that the classifiers accuracy would be high.

For every matrix that was constructed for the document representation schemes described the most salient features were plotted to show visually the differences in linguistic patterns between the two report categories.

Chapter Five

EXPLORATORY DATA ANALYSIS AND DIMENSIONALITY REDUCTION TECHNIQUES

"We are drowning in information but starved for knowledge"

John Naisbitt, 1982

5.1 Introduction

It has been shown thus far that an apt approach to harness language, would be to focus on a particular domain of interest and gather in documents to form a corpus. From these documents as was shown in chapter 4, features of interest can be extracted. Many candidate features for each document representation scheme were introduced. This can result in the existence of irrelevant/redundant features pertaining to the target concept. However in some cases the features are numerous and in others may be redundant or highly correlated. A feature is: "*relevant if it is neither irrelevant nor redundant to the target concept; an irrelevant feature is not directly associate with the target concept but affect the learning process, and a redundant feature does not add anything new to the target concept*" [252]. Reducing the number of irrelevant/redundant features can drastically reduce the running time of the learning algorithms and yields a more general classifier. This helps in getting a better insight into the underlying concept of deception in text [252].

Therefore, before classification can be executed the feature space has to be investigated to reduce such redundancy, ambiguity and noise: "*A noisy feature is one that, when added to the document representation, increases the classification error*" [49]. Unmanaged, these factors can result in a high dimensionality feature space known as the '*curse of dimensionality'*. This is where increasing dimensions results in increases in the hyperplane with increasing sparseness. Performance of text classifiers hinge on the underlying feature representation. Higher number of features are computationally expensive for many learning algorithms and raises the spectre of the classifiers overfitting the data (further explanation on overfitting in Chapter 6). Once reduced, the representation should correspond to the: "*intrinsic dimensionality*" of the data. This is the minimum number of features needed to represent document

content [253]. Determining relevant features in predictive financial based models is a commonly encountered issue. Common solutions applied are dimensionality reduction techniques as detailed in this chapter [254].

In chapter 4, 10 matrices were formed after feature extraction to represent both fraud and non-fraud documents:-

- Unigrams
- Bigrams
- Trigrams
- Coh-Metrix indices
- LIWC scores
- Custom dictionary scores
- Ratios from Linguistic Based Cues
- Topic modelling
- Concept scores
- Keywords

The features from these matrices need to be checked to ensure they meaningfully represent the documents. There are a number of statistical techniques that can examine the relationship between the observed variables (the features) and the latent or unobserved variable (fraud/non-fraud) in the reports. Factor analysis is commonly used in the social sciences as a way to take a mass of data and shrink it to a smaller data set that is more manageable and more understandable. The primary techniques being Exploratory Factor Analysis and Confirmatory Factor Analysis. Both are used to determine if a relationship between a set of observed variables (also known as manifest variables) and their underlying constructs exists. The former used primarily to explore patterns, whilst the latter is used to perform hypothesis testing. Related to such techniques are dimensionality reduction techniques as proposed by Fodor [255]. They generally fall into two categories (see Table 5.1) known as feature transformation and feature selection [256]. Methods from these two categories will be executed over the documents, as they have been deemed more appropriate in text mining applications. Some methods like Principal Component Analysis straddle both factor analysis and feature transformation approaches. Multi- Dimensional Scaling is primarily exploratory.

Using feature transformation techniques the original high dimensional space is

Dimensionality Reduction			Visualisation
Feature Transformation	**Feature Selection**		*Multidimensional Scaling (MDS)*
Principal Component Analysis (PCA)	**Wrapper**	**Filter**	
Latent Semantic Analysis (LSA)	*Boruta*	*Information Gain*	

Table 5.1: Dimensionality reduction approaches.

projected to a lower dimensionality space. These methods are believed to be very successful in uncovering latent structure in datasets. Techniques within this category include linear methods such as Principal Components Analysis and Latent Semantic Analysis.

In this chapter the following techniques (depicted in Figure 5.1) are applied to the data for reduction and exploratory purposes:-

- Latent Semantic Analysis using singular value decomposition
- Principle Component Analysis for feature section
- Boruta feature selection
- Information gain for feature selection
- Multidimensional Scaling

In feature selection methods, the objective is not extracting new features but rather removing features which seem irrelevant for modelling. This problem is a combinatorial optimization problem [257]. It removes: "*non-informative terms according to corpus statistics and use a term-goodness criterion threshold to eliminate some terms from the full vocabulary of the document corpus*" [258]. Reduction in features results in lower model complexity and better understanding of how these features impact the phenomena under scrutiny. As indicated by Guyon and Elisseeff [259]: *"The objective of variable selection is three-fold: improving the prediction performance of the predictors, providing faster and more cost-effective predictors, and providing a better understanding of the underlying process that generated the data".*

Therefore, the motivation behind feature selection is to remove redundant/irrelevant features from the data set as they can lead to a reduction of the classification accuracy or clustering quality and to an unnecessary increase of computational cost [260, 261]. The advantage of FS is that no information about the importance of single features is lost [261]. Once irrelevant features are dropped, a simpler, parsimonious,

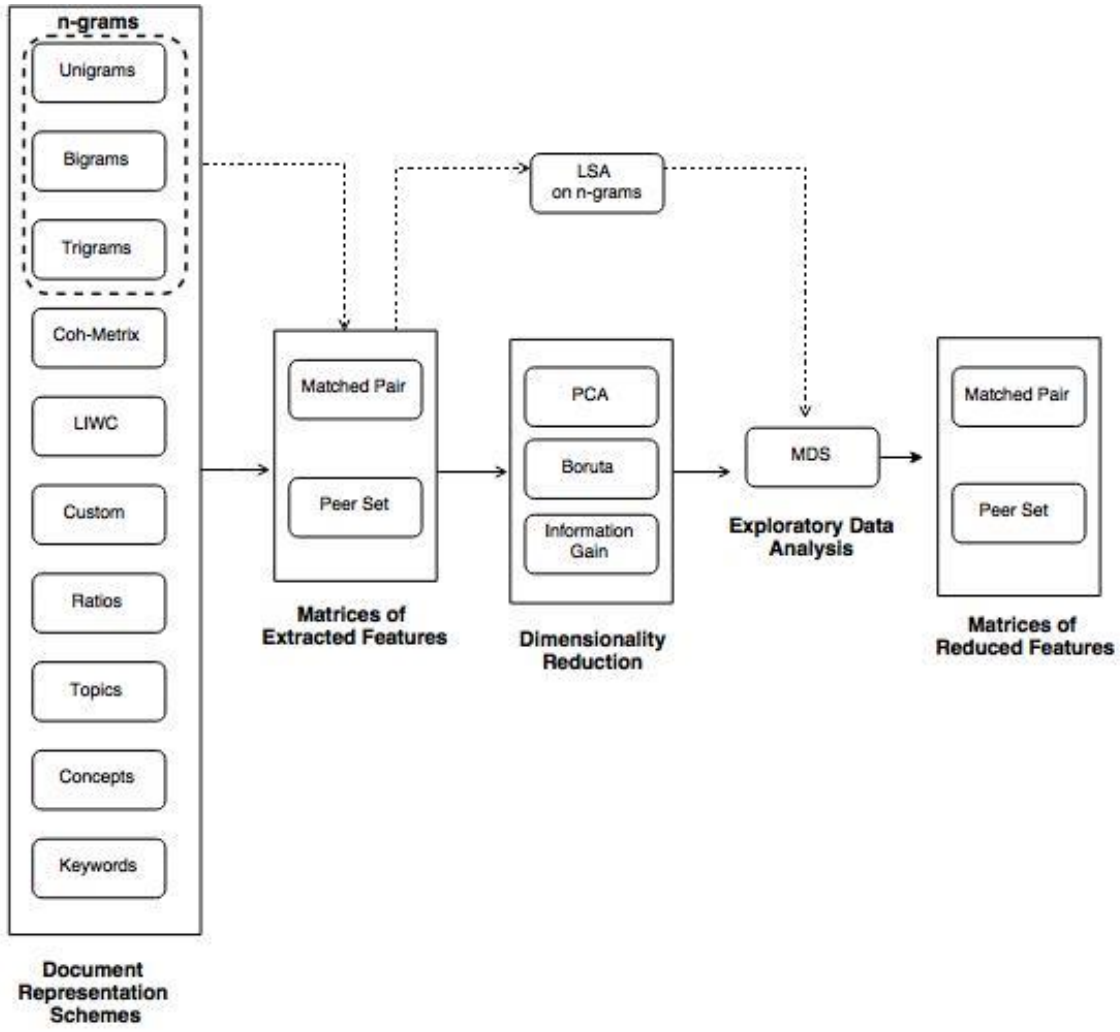


Figure 5.1: Dimensionality reduction techniques applied to matrices constructed in chapter 4.

interpretable model is then passed to the classifiers to enable better prediction. Janacek et al. [273] investigated the relationship between several attribute space reduction techniques (both filter and wrapper) and the resulting classification accuracy for two very different application areas. The experimental results underline the importance of a feature reduction process. The classification accuracy achieved with reduced feature sets was significantly better than with the full feature set. Several extensive surveys of various feature selection and dimensionality reduction approaches can be found in the literature, for example, a recent one was performed by Abdallah [261]. They underline the importance of the feature reduction step in building predictive models.

Feature selection routines from the two main feature selection methods known as filter methods and wrapper methods will be executed over the matrices identified in chapter 4. The former operates by applying a statistical measure to assign a scoring

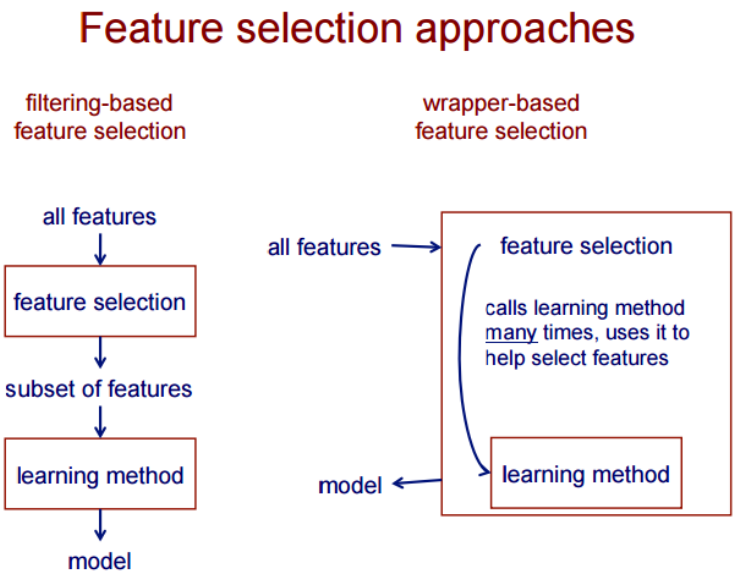


Figure 5.2: Feature selection approaches [273].

to each feature. The features are ranked by the score and either selected to be kept or removed from the dataset. The latter works by considering the selection of a set of features as a search problem, where different combinations are prepared, evaluated and compared to other combinations. The mechanics of the two feature selection approaches described in this chapter are captured in Figure 5.2. Wrapper based feature selection is dependent on the learning model chosen, whereas the alternative ignores feature dependence.

The dimensionality reduction techniques chosen will be executed over the 10 matrices. The output for both the peer set scenario and matched pair design will be shown using MDS. Each of the dimensionality reduction techniques will be elaborated in this chapter. For explanatory purposes one matrix will be chosen to demonstrate how the techniques are performing their operations and how the final data is obtained that will be then sent to the classifiers. The application of the techniques will be the same for all the other matrices. The flow of this chapter is captured in Figure 5.1. Using the framework, Figure 5.1 expands on the exploratory factor analysis and feature selection stage.

5.2 Latent Semantic Analysis (LSA)

This method: "*transforms the original textual data to a smaller semantic space by taking advantage of some of the implicit higher-order structure in associations of words with text objects*" [262]. The transformation is computed by applying truncated singular value decomposition (SVD) to the term-by-document matrix. In SVD, a rectangular matrix is decomposed into the product of three other matrices:

"One component matrix describes the original row entities as vectors of derived orthogonal factor values, another describes the original column entities in the same way, and the third is a diagonal matrix containing scaling values such that when the three components are matrix-multiplied, the original matrix is reconstructed. There is a mathematical proof that any matrix can be so decomposed perfectly, using no more factors than the smallest dimension of the original matrix. When fewer than the necessary number of factors are used, the reconstructed matrix is a least-squares best fit. One can reduce the dimensionality of the solution simply by deleting coefficients in the diagonal matrix, ordinarily starting with the smallest" [263].

Therefore SVD condenses the matrix by factorizing it. Words such as '*sales*' and '*revenue*' have similar context, such rows using SVD would be merged together. Thus, documents using different terminology to talk about the same concept would be positioned near each other in the new space. This handling of polysemy and synonymy results in it being used heavily in information retrieval for document searches based on query input.

LSA is designed specifically for text processing, and works with term-document matrices. Such matrices, however, are often considered too large, so they are reduced to form lower-rank matrices in a way very similar to Principal Component Analysis (both of them use SVD). It is not feature selection that is performed but feature vector transformation. According to Turney and Pantel [218] there are three perspectives on LSA that are notable:-

- It uncovers latent meaning by clustering words along a small number of dimensions. In many classification approaches, a keyword is assumed to be a unique representation of a distinctive concept or semantic unit. However, this is not the case and LSA helps reveal the concepts.

- It performs noise reduction and counters data sparseness through dimensionality reduction.
- Analyses higher order co-occurrence, words are deemed similar when they appear in similar context. As Landauer puts it: "*the representation of any meaningful passage must be composed as a function of the representations of the words it contains*" [148].

LSA was undertaken separately for both the (102) fraud reports and the (306) non fraud reports. The scikit-learn package in Python was chosen to execute LSA over the corpus. The Python interpreter allows greater interaction and manipulation of the data set. It also enabled easier extraction of concepts and the related words. LSA is built on the bag of words model where as indicated meaning of a passage is equal to the sum of the meanings of its words. Therefore, the pre-processing is the same as was conducted for the bag of words model depicted in Chapter 4 but this time stemming was not performed. This was done in order to gain a better appreciation of concepts in the reports.

After pre-processing code shown in Figure 5.3 is executed. The first 4 lines removes stop words and set up a term document matrix, called *X*. The TfidfVectorizer function takes in an argument called the *ngram_range*. This is unique to scikit-learn and is unavailable in other packages. This sets up a TDM that includes unigrams, bigrams and trigrams. Thus when LSA is executed it can perform better dimensionality reduction by determining all n-grams that are related to a concept.

Figure 5.4 shows the dimensions of the matrix *X*, which stores tf-idf scores for the 102 documents and n-grams found in the reports. The matrix is 102 rows for the 102 reports by 999611 columns to denote the n-grams (unigram, bigram, trigrams) found in the fraud reports. However as shown in the Figure 5.4 only 21075 of these terms are relevant once the matrix is stripped of all zeros and infrequent terms.

Tf-idf transformed document matrix is then passed to the LSA function TruncatedSVD. This transformer performs linear dimensionality reduction by means of truncated singular value decomposition (SVD). The *n_components* parameter is set to 50, which denotes the number of concepts to be extracted from the TDM, *n_iter* is the number of iterations undertaken to perform SVD.

This matrix X after SVD is decomposed results in the following:-

```
vectorizer = TfidfVectorizer(stop_words=stopset, use_idf=True, ngram_range=(1,3))
X=vectorizer.fit_transform(final)

lsa = TruncatedSVD(n_components=50, n_iter=100)
lsa.fit(X)


terms = vectorizer.get_feature_names()
with open("c:/temp/pconcepts.txt", "a") as file1:
    for i, comp in enumerate(lsa.components_):
        termsInComp =zip(terms, comp)
        sortedTerms= sorted(termsInComp, key=lambda x: x[1], reverse=True)[:10]
        file1.write("Concept %d:" %i)
        file1.write('\n')
        for term in sortedTerms:
            file1.write(term[0])
            file1.write('\n')
```

Figure 5.3: LSA executed over the reports using Python 3.5 [380].

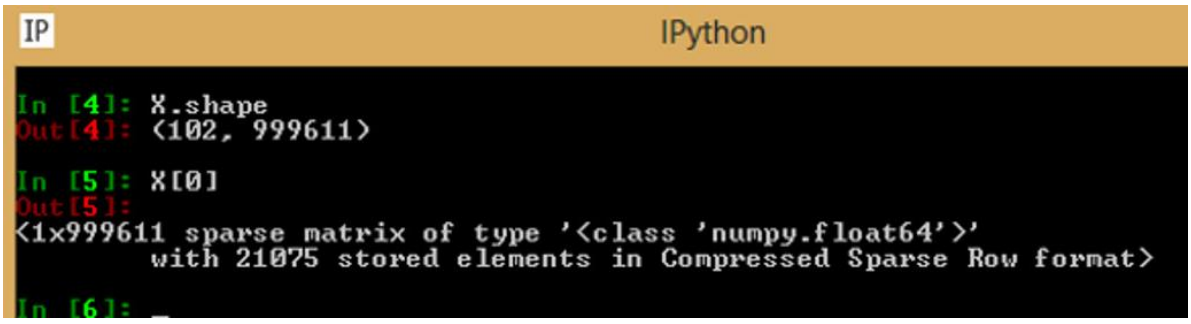


Figure 5.4: Dimension of matrix containing LSA concepts

- a U (m * k) matrix. The rows will be the documents and the columns will be the concepts.
- a S (k * k) diagonal matrix. The elements will be the amount of variation captured from each concept.
- a V (m * k) - transpose matrix. The rows will be terms and the columns will be concepts.

Therefore $M = USV^T$

These matrices when multiplied, give a new matrix M which is the least-squares best fit approximation of M with k (50, in this case) singular values [148].

The dimension reduction step has collapsed the component matrices in such a way: "*that words that occurred in some contexts now appear with greater or lesser estimated frequency, and some that did not appear originally now do appear, at least*

fractionally…. LSA induces similarity relations by changing estimated entries up or down to accommodate mutual constraints in the data" [148].

SVD has estimated what words appear in what context by using only the information extracted. It does this using the following logic: "*the text segment is best described as having so much of abstract concept one and so much of abstract concept two, and this word has so much of concept one and so much of concept two, and combining those two pieces of information (by vector arithmetic), my best guess is that word X actually appeared x times in context Y*" [264].

The last 10 lines of code in Figure 5.3, uses a for loop to go through the V matrix to pick up the terms that are part of each concept and writes this output to file. Appendix I, Table I.1 shows a few concepts/term bunching for the fraud reports. The above process was also run over the 306 non-fraud reports. The concept/term bunching for the non-fraud reports is shown in Appendix I, Table I.2, I.3 and I.4.

Appendix I, Table I.5 and Table I.6 show all the unique terms from concepts found in fraud and non-fraud reports. An analysis of terms in concepts reveal terms that are in fraud and not in non-fraud reports and vice versa. The results are shown in Appendix I, Figure I.1 and Figure I.2. A count is taken of all terms in concepts that are repeated in the reports. For example, the term '*company'* was repeated 12 times in concepts identified in the fraud reports and 35 times in non-fraud reports. The results of counting all terms in concepts are shown graphically in Appendix I, Figure I.1 and Figure I.2.

Based on terms in concepts uncovered by LSA and analysis conducted as shown in Appendix I, 40 term/concepts were identified as showing potential to aid classification of fraud/non fraud reports. These 40 features are shown in Table 5.2. The terms identified in the concepts using LSA were then put through MDS to determine the distances between the reports based on these terms, as shown in Appendix J, Table J.1.

There are some issues that need to be taken into account when using LSI over a corpus. LSI may ignore important features for some documents as they may not be the most important feature for all the document collection [265]. Such features are removed in the dimension reduction step. This can be resultant from the fact that when a global Latent Semantic Space is created for all documents the classification information (ie whether a document class is fraud or non-fraud) is usually not considered. This can be mitigated by including the classification information [265].

Terms from concepts identified by LSA to be used for classification			
acquisitions	costs	interest	insurance
adverse	credit	investment	procedures
amount	currently	loans	production
average	customers	lower	products
bank	debt	management	required
capital	decrease	marketing	result
cash	fiscal	may	revenue
certain	higher	net	securities
communities	impact	operating	total
control	increased	operations	value

Table 5.2: Terms from concepts identified by LSA to be used for classification.

According to Shafiei et al. [266] is that SVD, along with other least squares methods, is really designed for normally distributed data but such a distribution is not representative of the term by document matrix. This can result in negative values when constructed the matrix after SVD, which is inappropriate as the matrix has count data (ie number of times a term occurs in a document). The new dimension with re-calculated frequency values are hoped to be a better representation of underlying concepts in the corpus. However, such clear distillation and interpretation of concepts are difficult because as indicted it is not always possible to attain concrete, physical quantities for all concepts.

It has also been pointed out by [261] that there is no theoretical optimum for the number of dimensions to be kept. In fact as Landauer et al. [263] point out the underlying principle is that the original data should not be perfectly regenerated but: *"rather an optimal dimensionality should be found that will cause correct induction of underlying relations, the customary factor-analytic approach of choosing a dimensionality that most parsimoniously represent the true variance of the original data is not appropriate. Instead some external criterion of validity is sought, such as the performance on a synonym test or prediction of the missing words in passages if some portion are deleted in forming the initial matrix"* [263].

Therefore the optimal dimensionality is highly heavily experimental in nature and therefore classification based on LSI reduced matrix depend upon its reduced dimensions. The authors also add that LSA's "bag of words" method ignores all: "*syntactical, logical and non-linguistic pragmatic entailments which sometimes misses meaning or gets it scrambled*" [263].

Furthermore, Landauer et al. [263] point out that that some words that have more than one contextual meaning receive a sort of average high-dimensional placement that out of context signifies nothing, and that many words are sampled too thinly to get well placed. An example given by Hand et al. [267] illustrates this susceptibility to spurious correlations. *"If our corpus happens to contain lots of documents which mention "farming" and "Kansas", as well as "farming" and "agriculture", latent semantic indexing will not make a big distinction between the relationship between "agriculture" and "farming" (which is genuinely semantic) and that between "Kansas" and "farming" (which is accidental, and probably wouldn't show up in, say, a corpus collected from Europe*" [267]. Similarly for the corpus under study there are such spurious relationship that can be "telecommunications" and "US" (an accidental association) in comparison to say a word "satellite" (genuinely semantic).

5.3 Principle Component Analysis (PCA)

PCA distributes the variation in a multivariate dataset across components in a way that enables pattern observation. Again as with LSA the aim is to reduce the dimensionality of the data: "*consisting of a large number of interrelated variables while retaining as much as possible of the variation present in the data set. This is achieved by transforming a new set of variables the principle components (PCs) which are uncorrelated and which are ordered so that the first few retain most of the variation present in all of the original variables*" [268].

PCA takes the cloud of data points and rotates it such that the maximum variability is visible. An eigenvector is drawn through the perspective that shows up the most variability. An eigenvalue is a number, telling you how much variance there is in the data in that direction, in other words how spread out the data is on the line. The eigenvector with the highest eigenvalue is therefore the principal component. Typically, the amount of eigenvectors/values that exist equals the number of dimensions of the dataset. The eigenvectors have just put the data into a new set of dimensions to show up the greatest variance in the data. The data have been transformed into a new coordinate system. As indicated the first axis corresponds to the first principle component that explains the greatest amount of variance in the data. Eigenvectors drawn must be orthogonal to each other, this enables capture of variance

not caught by previous principal components. In general, most of the variance will be explained by a very small number of principal components.

The matrices from chapter 4 will be taken individually and put through PCA to reduce their dimensionality and to determine variable that cause the most variability in the data set. These are then passed to the classifiers in chapter 6. The concept mining matrix for the matched pair design setup will be used to illustrate how PCA is performed over the data. The PCA routine from the factoMineR and factoextra packages in the R programming language are used to perform the computation. The former is used to execute PCA commands over the matrix, whilst the latter is used for visualisation.

An extract of the matrix that was produced after concept mining as depicted in chapter 4 was performed is shown below. This matrix is for the matched pair design scenario, therefore the rows in the matrix, the 'objects' are the 102 fraud reports and 102 non-fraud reports. The columns are termed 'variables' and comprise the measurements made on the objects. In this case the variables are the concepts identified.

The dimensions of this matrix constitute 204 rows by 5595 columns. All columns (concepts) that added up to zero were removed as were all columns that had less than 5 entries. This considerably reduced the sparsity and likely spurious data.

Now follows snippets of code with explanation.

Line 37 and Line 38 loads in the libraries. Line 39 performs the PCA over the matrix.

It has performed the following steps:-

1. Scale the data: This gives each variable an equal opportunity to contribute to the principal component analysis.
2. Calculate the covariance values between all the different dimensions. Covariance measures how much the dimensions vary from the mean with respect to each other [269]. A positive co-variance value indicates that both variables increase together. Whereas a negative value indicates that as one dimension increases, the other decreases. The formula for calculating covariance is very similar to the formula for variance and is shown below [269].

$$cov(X,Y) = \frac{\sum_{i=1}^{n}(X_i - \bar{X})(Y_i - \bar{Y})}{(n-1)} \qquad \text{(Eq 5.1)}$$

achievements.noun	achieving.verb	acid.noun	acknowledged.verb	acquire.verb	acquired.verb	acquirer.noun	acquirers.noun	acquires.verb	acquiring.verb	acquisition.noun
0.0000000	0.00000000	0.0000	0.00000000	0.3045877	0.5464576	0.0000	0.0000	0.0000	0.1608595	1.8523577
0.0000000	0.00000000	0.0000	0.00000000	0.2580038	0.6880100	0.0000	0.0000	0.0000	0.1958569	0.9036723
0.0000000	0.00000000	0.0000	0.00000000	0.2202000	0.6524000	0.3457	0.0000	0.0000	0.0000000	1.5472000
0.0000000	0.00000000	0.0000	0.00000000	0.1457000	0.5935000	0.0000	0.0000	0.0000	0.0000000	0.8735000
0.0000000	0.00000000	0.0000	0.00000000	0.0000000	0.0000000	0.0000	0.0000	0.0000	0.0000000	0.7661000
0.0000000	0.00000000	0.0000	0.00000000	0.1878000	0.3722000	0.0000	0.0000	0.0000	0.1843000	1.4990000
0.0000000	0.00000000	0.0000	0.00000000	0.3305000	1.2378000	0.0000	0.3023	0.2405	0.2787000	6.4663000
0.0000000	0.00000000	0.0000	0.00000000	0.2563000	1.0734000	0.0000	0.2745	0.2483	0.2870000	3.3174000
0.0000000	0.00000000	0.0000	0.00000000	0.8000000	1.3719000	0.0000	0.0000	0.2585	0.3887000	3.7162000
0.0000000	0.00000000	0.0000	0.00000000	0.6530000	0.9608000	0.0000	0.0000	0.2517	0.5989000	2.8825000
0.0000000	0.00000000	0.0000	0.00000000	0.5706000	0.4815000	0.0000	0.0000	0.3945	0.2687000	2.5719000
0.0000000	0.00000000	0.0000	0.00000000	0.3465000	0.5044000	0.0000	0.0000	0.0000	0.2182000	1.8261000
0.0000000	0.00000000	0.0000	0.00000000	0.5840000	0.2040000	0.0000	0.0000	0.0000	0.1615000	1.2303000
0.0000000	0.00000000	0.0000	0.00000000	0.6767000	0.2533000	0.0000	0.0000	0.0000	0.1691000	1.5482000
0.0000000	0.00000000	0.0000	0.00000000	0.2703000	0.3144000	0.2550	0.0000	0.0000	0.1725000	0.6511000
0.8517000	0.00000000	0.0000	0.00000000	0.5056000	0.7216000	0.0000	0.0000	0.0000	0.5406000	3.2737000
0.0000000	0.00000000	0.0000	0.00000000	0.3860000	1.4200000	0.0000	0.0000	0.0000	0.3295000	9.4172000
0.0000000	0.00000000	0.0000	0.00000000	0.5831000	1.4655000	0.0000	0.0000	0.0000	0.3965000	7.8901000
0.0000000	0.00000000	0.0000	0.00000000	1.3960000	0.4140000	0.5260	0.0000	0.0000	0.5110000	3.1420000
0.0000000	0.00000000	0.0000	0.00000000	0.3280000	0.3000000	0.1390	0.0000	0.0000	0.1760000	0.6100000
0.0000000	0.00000000	0.0000	0.02970000	0.1900000	0.8272000	0.0000	0.0000	0.0000	0.0000000	2.3870000

Figure 5.5: Matrix based on concepts derived using WordNet.

```
37  library("factoextra")
38  library("FactoMineR")
39  res.pca <- PCA(data2, graph = FALSE)
40  eigenvalues <- res.pca$eig
41  head(eigenvalues[, 1:2])
42  fviz_screeplot(res.pca, ncp=10)
43  head(res.pca$var$coord)
44  head(res.pca$var$contrib)
45  fviz_contrib(res.pca, choice = "var", axes = 1, top = 25)
46  dimdesc(res.pca, axes = 1:3, proba = 0.05)
47  res.desc <- dimdesc(res.pca, axes = c(1,2))
48  res.desc$Dim.1
49  a<-res.desc$Dim.1
50  a<- as.data.frame(a)
51  aframe <- a[order(a$quanti.p.value), ]
```

Figure 5.6: PCA executed over selected matrix in R.

	Name	Description
1	$eig	eigenvalues
2	$var	results for the variables
3	varcoord	coord. for the variables
4	varcor	correlations variables - dimensions
5	varcos2	cos2 for the variables
6	varcontrib	contributions of the variables

Table 5.3 Terms used in R for PCA computation.

The formula interpreted means: "*For each data item, multiply the difference between the x value and the mean of x, by the difference between the y value and the mean of y. Add all these up, and divide by (n-1)*" [269]. All the possible covariance values between all the different variables is calculated and put into a matrix.

3. Calculate the eigenvectors and eigenvalues of the covariance matrix. The former is the direction in which the data varies the most and the magnitude of this vector equals the corresponding eigenvalue. The second largest eigenvector is always orthogonal to the largest eigenvector, and points into the direction of the second largest spread of the data. The third eigenvector is the direction of greatest variance among those orthogonal to the first two and so on [269].

Line 40 and 41 in Figure 5.6 executes the command to show the proportion of variation (the eigenvalues) retained by the principal components (PCs). Output shown in Figure 5.7. The first PC corresponds to the direction with the maximum amount of variation in the data set. This can be shown visually in Figure 5.8. As predicted most of the variance is captured by the first PCA. The correlation between a variable and a PC is called *loading*. An extract of the loadings is printed once line 43 in Figure 5.6 is executed and is shown in Figure 5.9.

As shown in Figure 5.8 variables from PC1 and PC2 are the most important in explaining the variability in the data set. Variables that do not correlate with any PC or correlated with the last dimensions are variables with low contribution and can be removed. Line 44 once executed give the contributions of variables in accounting for the variability in a given principal component. An extract of the results is shown in Figure 5.9. The larger the value of the contribution the more the variable contributes to the component. Lines 46-51 extract the top 25 most significant variables for dimension 1. The top 25 variables from dimension 1 or PCA 1 are depicted in Figure 5.10.

The top 50 variables are extracted from PCA 1 for each matrix (both peer set and matched pair) identified in chapter 4. This will then be passed to the classifiers to be used to determine if they aid in discriminating a fraud from a non-fraud firm. Results will be shown in chapter 6. As demonstrated PCA has reduced the information dimensionality that is inherent in the matrix that contained a large number of interrelated variables.

	eigenvalue	percentage of variance
comp 1	293.5560	4.714244
comp 2	145.6603	2.339173
comp 3	130.9907	2.103593
comp 4	123.6058	1.984998
comp 5	108.9590	1.749783
comp 6	102.5448	1.646776

Figure 5.7: Principal components with the highest variance.

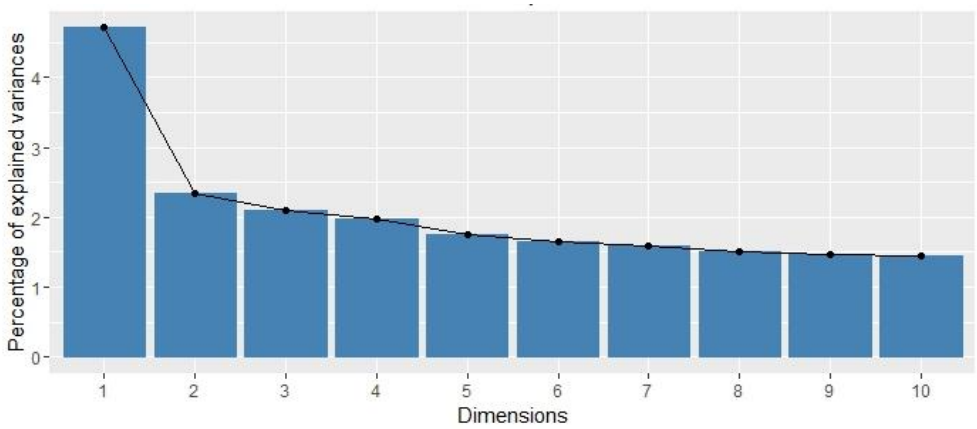


Figure 5.8: Variance of Principle components graphed.

	Dim.1	Dim.2	Dim.3	Dim.4	Dim.5
abandon.verb	0.55916923	-0.08577087	0.216946333	-0.222456984	0.112354754
abandonment.noun	0.40080246	0.06783291	-0.304570534	0.065739545	0.279762073
abilities.noun	-0.06501527	-0.02397113	-0.045497273	-0.005728029	-0.004398321
ability.noun	-0.01789101	0.14854169	0.294367398	0.165779862	0.041938844
absence.noun	0.31652972	-0.08876278	0.108214185	0.126274028	-0.270393934
absolute.noun	0.15782081	0.34071719	0.189288921	-0.297630903	0.030038012
absorb.verb	0.09191586	0.14842276	-0.081134727	-0.162345347	0.008873998
absorbed.verb	0.02685715	0.01533602	-0.067201400	0.069460223	-0.004406291
absorption.noun	0.11463279	-0.15222164	-0.038782104	0.081572016	0.249060720
abuse.noun	0.23455425	0.05578763	-0.345443428	0.231104151	-0.249057387

Figure 5.9: Variance of features captured by principle components.

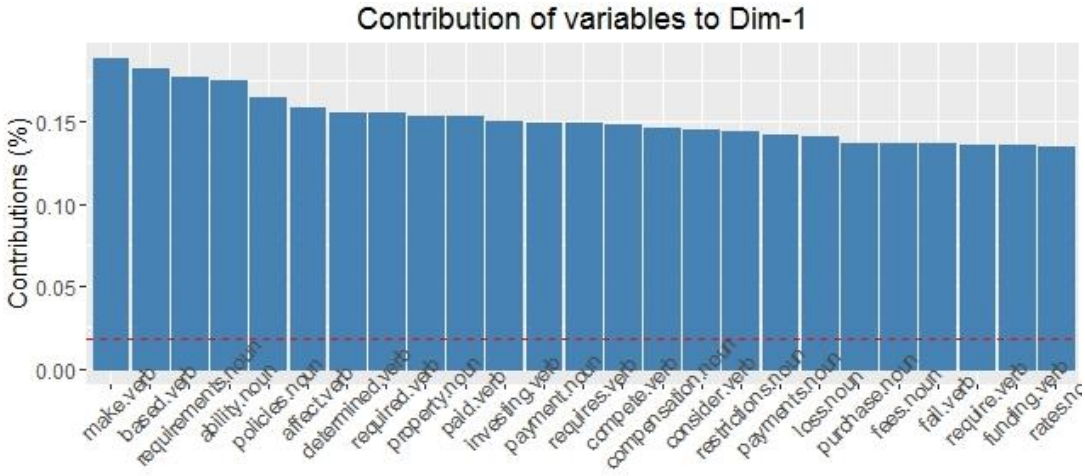


Figure 5.10 Features in first dimension (PCA):

5.4 Boruta Feature Selection

Feature selection is fundamental to the drive to harness data for better prediction and classification. Typically, data sets are characterised by far too many variables, often highly correlated for model building. Most of these variables are irrelevant to classification and their relevance is not known in advance. As Kursa [270] argues using too many variables slows down the classification algorithms and decreases accuracy. According to Engelhardt [271] there are two approaches to selecting the features (variables): *"the minimal-optimal feature selection which identifies a small (ideally minimal) set of variables that gives the best possible classification result (for a class of classification models) and the all-relevant feature selection which identifies all variables that are in some circumstances relevant for the classification"*.

The Boruta package [270] is built on the latter approach of all-relevant feature selection. As Englebert [271] points out this approach is illuminating as it leads to better understanding of the mechanisms related to the subject of interest. This would be beneficial for the corpus under study as it would shed light on all relevant variables that aid in the discrimination between fraud and non-fraud firms. A wrapper method is used in Borurta to determine a variables importance to classification. In this method the classifier - random forest is used as a black box returning a feature ranking [270]. *"It is an ensemble method in which classification is performed by voting of multiple unbiased weak classifiers — decision trees. These trees are independently developed on different bagging samples of the training set. The importance measure of an attribute is obtained as the loss of accuracy of classification caused by the random permutation of attribute values between objects. It is computed separately for all trees in the forest which use a given attribute for classification. Then the average and standard deviation of the accuracy loss are computed. Alternatively, the Z score is computed by dividing the average loss by its standard deviation can be used as the importance measure. In Boruta a Z score as the importance measure since it takes into account the fluctuations of the mean accuracy loss among trees in the forest"* [270].

In order to corroborate the importance of the Z score, it is compared to a random permutation of a selection of variables to test if it is higher than the scores from random variables. Classification is then performed using all attributes. It iteratively compares

importance of attributes with importance of shadow attributes created by shuffling original ones. Attributes that have significantly worst importance than shadow ones are being consecutively dropped. Those attributes that are significantly better than shadows are admitted to be confirmed. Shadows are re-created in each iteration. Algorithm stops when only 'confirmed' attributes are left. Some attributes may be left without a decision. They are claimed 'tentative' [270].

In sum, as Kursa [270] argues Boruta is based like the random forest classifier on the idea that by adding randomness to the system and by collecting results from the ensemble of randomised samples reduces the misleading impact of random fluctuations and correlations. However wrapper methods are known to be slower than the filter methods and have a tendency towards overfitting – discrepancy between the evaluation score and the ultimate performance [259]

The Boruta algorithm is executed over the matrices that represent the documents (covered in chapter 4) both for the peer set and matched pair combinations. To illustrate how Boruta feature selection was executed over the matrices, the peer set LIWC matrix processing using Boruta feature selection is expanded below. All the other matrices were processed in a similar manner and results are shown in Appendix L.

The LIWC matrix, extract shown in Figure 5.11 would be read into a variable in R (line5) Figure 5.12. The Boruta algorithm would then be executed over the matrix and the resultant features chosen held in *bor_features*. These features are then extracted from the original matrix (data3) – line 8. In this matrix the features extracted are shown in Table L.5, Appendix L. This reduced matrix would then be passed to the classifiers.

5.5 Information Gain Feature Selection

Yang and Pederson [258] conducted a comparative study of feature selection methods in statistical learning of text categorisation. Five methods were evaluated. They found that information gain was amongst the most effective as the resultant features that had sifted out yielded an improved classification accuracy. A result affirmed by later studies [272].

Filename	Segment	WC	Analytic	Clout	Authentic	Tone	WPS	Sixltr	Dic	function	pronoun	ppron	i	we	you	shehe
f adelphia 1999.txt	1	22992	98.10	54.03	21.61	60.15	26.13	37.71	71.16	36.30	3.12	0.33	0.05	0.02	0.00	0.01
f Adelphia 2000.txt	1	20745	98.13	53.97	20.93	55.00	27.70	38.17	71.89	36.51	3.24	0.41	0.03	0.05	0.00	0.01
f Anicom Inc 1998.txt	1	7088	98.09	57.45	24.72	58.87	26.35	38.94	75.65	37.18	3.68	0.21	0.04	0.03	0.00	0.01
f Anicom Inc 1999.txt	1	6989	97.31	64.68	23.09	54.60	26.88	37.19	76.69	38.40	5.21	1.67	0.04	1.49	0.06	0.00
f Applied Microsystems 2001.txt	1	10498	97.60	55.07	17.30	37.92	27.41	41.63	76.22	38.79	3.30	0.45	0.00	0.25	0.02	0.00
f Applied Microsystems 2002.txt	1	14294	94.78	77.21	13.61	53.10	30.09	36.73	75.46	42.16	6.97	4.60	0.00	4.39	0.02	0.00
f Assisted Living Concepts 2008.txt	1	26746	96.98	71.91	18.75	53.31	28.30	36.60	77.32	39.58	5.77	3.56	0.01	3.30	0.00	0.01
f Assisted Living Concepts 2009.txt	1	25272	96.95	71.53	20.32	52.81	28.11	35.99	77.21	39.56	5.71	3.51	0.00	3.26	0.00	0.02
f Bally Gaming and Systems 2004.tx	1	22992	97.11	64.43	13.58	70.31	29.07	34.18	76.24	40.77	4.61	2.42	0.08	2.09	0.01	0.03
f Bally Gaming and Systems 2005.tx	1	18230	96.09	75.76	22.11	61.45	27.21	34.02	76.85	39.85	6.18	3.86	0.03	3.58	0.04	0.00
f Beazer Homes 2005.txt	1	12543	94.86	80.73	23.53	58.19	28.57	35.33	78.47	41.73	7.51	5.31	0.00	5.05	0.00	0.00
f Beazer Homes 2006.txt	1	15412	95.59	79.09	23.52	56.53	26.85	35.23	77.02	40.78	6.87	4.76	0.01	4.54	0.00	0.00
f Broadcom 2006.txt	1	37364	90.72	78.18	20.32	50.28	29.49	37.21	75.26	40.24	7.43	4.92	0.05	4.50	0.07	0.00
f Broadcom 2007.txt	1	42131	91.28	77.90	20.69	50.92	30.24	36.86	75.47	40.23	7.37	4.87	0.04	4.48	0.06	0.00
f BROOKE CORPORATION 2006.txt	1	32506	95.33	72.69	13.59	73.51	27.85	34.89	79.39	39.80	6.15	3.46	0.01	3.14	0.02	0.01
f BROOKE CORPORATION 2007.txt	1	39741	96.18	63.79	12.87	58.81	27.02	36.60	77.50	39.02	5.07	2.27	0.02	1.95	0.02	0.01
f Cabletron Systems Inc 2000.txt	1	20697	98.39	53.61	20.23	55.08	27.05	37.96	75.22	37.97	3.00	0.21	0.00	0.00	0.00	0.00
f Cabletron Systems Inc 2001.txt	1	23451	97.15	66.50	19.88	51.97	26.41	38.45	76.09	39.45	5.25	2.46	0.00	2.16	0.01	0.00
f CHINA NATURAL GAS, INC. 2008.tx	1	14191	96.39	73.73	14.55	41.44	26.28	33.56	73.40	39.58	5.67	3.54	0.04	3.30	0.05	0.02
f CHINA NATURAL GAS, INC. 2010.tx	1	31995	95.43	73.14	21.36	41.90	30.44	31.16	73.50	41.26	6.40	4.33	0.03	3.96	0.05	0.05
f ChinaMedia Express Holdings 200	1	20492	91.74	80.60	12.50	81.78	33.70	32.69	80.93	47.71	8.60	5.66	0.02	4.74	0.00	0.16
f ChinaMedia Express Holdings 200	1	44462	97.61	48.32	22.35	48.74	30.75	34.51	73.40	39.21	4.30	0.53	0.02	0.12	0.02	0.04
f Computer Associates 2000.txt	1	9852	98.39	53.00	21.08	60.21	25.07	39.71	74.15	37.30	2.83	0.26	0.01	0.01	0.00	0.00
f Computer Associates 2001.txt	1	12561	98.48	53.38	21.30	61.55	25.43	40.32	74.99	37.79	2.85	0.31	0.00	0.01	0.00	0.00
f Computer Sciences Corporation 2	1	21090	98.16	61.54	19.13	58.88	24.99	36.71	75.81	38.99	3.66	1.50	0.01	1.38	0.00	0.00
f Computer Sciences Corporation 2	1	19415	97.67	61.88	17.18	59.83	25.21	36.55	75.29	39.02	3.94	1.75	0.01	1.61	0.00	0.00

Figure 5.11: An extract of matrix with LIWC features representing the reports.

```
1  library("Boruta")
2  library("mlbench")
3
4
5  data3 <- read.csv(file="c:/data6/new LIWC results/LIWCCSVALL.csv", header=TRUE, sep=",")
6  Boruta.Fraud2 <- Boruta(classT ~ ., data = data3, doTrace = 2,ntree = 500)
7  bor_features <-getSelectedAttributes(Boruta.Fraud2, withTentative = FALSE)
8  data3 <- subset(data3, select = (bor_features))
```

Figure 5.12: The Boruta FS algorithm executed over the matrix shown in Figure 5.11.

Based on such results and the prominence it holds as a feature selection routine in text classification [49] the Information Gain (IG) feature selection program in R from the FSelector library is executed over the matrices of extracted features. This is also a central filter feature selection method and is '*classifier agnostic*' [273]. The aim of IG is to find out how well each single feature separates the given data set into the 'f' and 'nf' categories. The IG of an attribute would indicate how much information with respect to the classification target 'f' and 'nf' the attribute imparts. In other words it measures: "*how much information the presence/absence of a term contributes to making the correct classification decision*" [49]. The: IG of the feature t_k over the class c_i is the reduction in uncertainty about the value c_i when the value t_k is known. The IG of the feature t_k over the class c_i can be calculated as follows [49]:

$$IG\ (t_k, c_i) = \sum_{c \in \{c_i, \bar{c}_i\}} \sum_{t \in \{t_k, \bar{t}_k\}} P(t,c) \log \frac{P(t,c)}{P(t)P(c)},$$

Eq (5.2)

P(c) is the fraction of the documents in category c and covers the total number of documents. P(t,c) is the fraction of documents in the category c that contain the word t over the total number of documents. P(t) is the fraction of the documents containing the term t over the total number of documents [49] . The highest scoring features by IG are kept. Generally, once the information gain has been calculated for all attributes and sorted, the attributes which obtain an information gain over a predetermined threshold will be added to the feature selection subset.

The important thing to note in this context is that the information gain is a purely information-theoretic measure, unlike Boruta feature selection, it makes no use of a classification algorithm.

The main drawback of using the information gain filter described above is that it tests each feature individually thus any correlation between features may be ignored. Thus there could be redundant features in the final model. However as pointed out by [260] filter methods are simple and cheap methods and give good empirical results, they are fast and effective and can be used as pre-processing for more sophisticated methods.

The code that executes information gain feature selection over the matrices unveiled in chapter 4 is depicted in Figure 5.14. The code specifically refers to the keywords-Rutherford matrix, peer set data set up. Excerpt from that matrix shown in Figure 5.13. However all other matrices went through a similar routine for the Information Gain computation. First, from Figure 5.14, FSelector package in R is initialised (line2). This contains algorithms for filtering attributes, including IG. The main keywords that were identified by Rutherford [138] were then extracted from the bag of words (unigram) term document matrix (line 19). The IG function in line 21 is given the class information (‘f’ or ‘nf’) and the matrix data3 (dimension 408, rows - 102 ‘f’ reports and 306 ‘nf’ reports with the Rutherford keywords as columns. The cells being tf-idf scores for each report/keyword). The function finds weights of discrete attributes (the keywords in the matrix) based on their correlation with the class attribute (‘f’ or ‘nf’). Line 23 executes the cutoff.*k* function which picks the top 20 best features. Line 24 and 25 ensures that the attributes are sorted in descending order. This would give the attributes with the highest weights, an extract of the top 15 attributes are shown in Figure 5.15. These attributes are used to form the reduced matrix that is passed to the classifiers. The other matrices of extracted features covered in chapter 4 would go through similar feature selection processing using information gain.

	activity	borrowing	capital	cash	company	completed	continued	cost	currency	customer	debt
1	2.585551e-05	4.467020e-04	0.000000e+00	5.057979e-05	0.000000e+00	1.098137e-04	3.744375e-05	1.685993e-05	0.000000e+00	3.917587e-04	3.177876e-04
2	2.859801e-05	4.940840e-04	0.000000e+00	4.195861e-05	0.000000e+00	1.579001e-04	4.733192e-05	1.631724e-05	0.000000e+00	4.975073e-04	3.180197e-04
3	1.705998e-04	2.105306e-04	0.000000e+00	6.257541e-05	0.000000e+00	2.536003e-04	1.764724e-05	2.085847e-05	0.000000e+00	7.658970e-04	9.984885e-05
4	2.602702e-04	0.000000e+00	0.000000e+00	7.778730e-05	0.000000e+00	7.369475e-05	3.589728e-05	2.828629e-05	1.915956e-04	9.250362e-04	1.269427e-04
5	0.000000e+00	0.000000e+00	0.000000e+00	2.714409e-05	0.000000e+00	1.178649e-04	5.741291e-05	9.048029e-05	6.128632e-04	7.163759e-04	1.624224e-05
6	4.411545e-05	0.000000e+00	0.000000e+00	7.191731e-05	0.000000e+00	2.248409e-04	7.301447e-05	8.989664e-05	1.948512e-04	4.208637e-04	2.581995e-05
7	2.348705e-05	2.898447e-04	0.000000e+00	1.703849e-04	0.000000e+00	1.695824e-04	1.943644e-05	4.977538e-05	0.000000e+00	1.318045e-05	3.299167e-04
8	2.492868e-05	6.152706e-05	0.000000e+00	1.666195e-04	0.000000e+00	8.470178e-05	2.062945e-05	3.860696e-05	0.000000e+00	1.398947e-05	2.772155e-04
9	1.616064e-04	1.994322e-04	0.000000e+00	1.229442e-04	0.000000e+00	2.974293e-04	3.900619e-05	1.756345e-05	3.866365e-04	4.383359e-04	2.916376e-04
10	1.693387e-04	0.000000e+00	0.000000e+00	1.656343e-04	0.000000e+00	2.589179e-04	5.605372e-05	3.036628e-05	2.991772e-04	3.991236e-04	3.270655e-04

Figure 5.13: An excerpt of matrix with Rutherford [199] keywords passed to IG.

```
library("FSelector")
library("caret")


rutherford408 <- c("activity","borrowing","capital","cash","company",
                   "completed","continued","cost","ccurrency",
                   "customer","debt","decrease","development",
                  "division", "due", "end", "exchange", "expenditure",
                  "facility", "financial", "growth", "high", "higher",
                  "include", "increase", "increasingly", "interest",
                  "investment", "item", "last", "liability", "lower",
                  "make", "management", "margin", "net", "network", "new",
                  "now", "number", "operating", "operations", "overall", "performance",
                  "previous", "products", "profit", "property", "reduce", "result", "retail",
                  "revenue", "risk", "sale", "services", "share", "significant", "store",
                  "strong", "systems", "tax", "trading", "turnover", "years", "loss",
                  "asset", "fixed", "rate", "level", "activity", "major", "total", "grow", "class")

data3 <- subset(tdm.stack, select = (rutherford408))

weights <- information.gain(class~., data3)
print(weights)
subset <- cutoff.k(weights, 20)
w2 <-weights[order(-weights$attr_importance), , drop = FALSE]
head(w2,100)
```

Figure 5.14: IG executed over matrix formed using keywords extracted from Rutherford [199].

	attr_importance
years	0.54853647
company	0.53811178
financial	0.52894600
increase	0.52894600
significant	0.52894600
capital	0.50554007
growth	0.48577729
include	0.39160627
result	0.38587831
operations	0.22661227
loss	0.21623062
tax	0.21444118
operating	0.19850922
make	0.18111130
management	0.16313646

Figure 5.15: IG selected Rutherford [199] keywords.

5.6 Multi-Dimensional Scaling (MDS)

The main goal of MDS it is to plot multivariate data points in two dimensions, thus revealing the structure of the dataset by visualizing the relative distance of the observations. The data for MDS analysis are called proximities. This indicates the similarity or dissimilarity of the documents based on the variables under investigation. An MDS program then looks for spatial configuration of the objects so that the distances between the objects match their proximities as closely as possible. The result is a visual representation of the pattern of proximities. Objects have been rearranged so as to arrive at a configuration that approximates the observed distances. In this corpus the aim is to arrange the documents in a space with a number of dimensions (two in this example) to determine if they can fall into two categories (fraud and non-fraud).

The input data for MDS is in the form of a distance matrix representing in this corpus the distance between documents (as measured by the words in each document). As indicated this is then represented by a configuration in a smaller number of dimensions such that the distances on the configuration reproduce approximately the original space. Therefore 2 documents that are closest together according to the distance matrix should be closest together on the configuration. Therefore, these new distances on the map would be in the same metric (scale of measurement) as the original.

Borgatti [274]

There are a few variants of multi-dimensional scaling algorithm in use. The most salient are centred around classical MDS, metric and non-metric MDS [379]. In all variants an input matrix is set up that captures similarities/dissimilarities between pairs using a distance matrix, captured using Euclidean distance calculations of items and: "*outputs a coordinate matrix whose configuration minimizes a loss function called strain or stress* [378]" The algorithm for all three variants are expounded by Wickelmair [377] and to a lesser degree Borgatti [274].

Figure 5.16 shows the R code used to perform MDS. The MASS library that supports MDS is loaded in at line1. Next the matrices derived from documentation

```r
library("MASS")
#Dictionaries
data2 <- read.csv(file="data6/wa/Dictionaries/L&M Sentimentsv8csv.csv", header=TRUE, sep=",")
# coh_metrix
data2 <- read.csv(file="data6/wa/Coh_Metrix/cohwholecsvall.csv", header=TRUE, sep=",")
data2 <- subset(data2, select = (peercohmetrixpca))
# topics
data2 <- read.csv(file="data6/wa/topicmallet/topicscsvall.csv", header=TRUE, sep=",")
data2 <- subset(data2, select = (peersettopicspca))
num <- c(1:408)
data2["hdr"] <- num
#ratios
data2 <- read.csv(file="data6/wa/LBC/ratios_csv_V4.csv", header=TRUE, sep=",")
data2 <- subset(data2, select = (peersetLBCpca))
data2$class<-NULL
row.names(data2) <- data2[, 1]
x <- data2
x[] <- lapply(x, as.numeric)
data2 <- x
d <- dist(data2)
fit <- cmdscale(d, eig = TRUE, k = 2)
fit$points
Dim1 <- fit$points [1:102,]
Dim2 <- fit$points [103:408]
plot(Dim1, col=2)
points(Dim2, col=3)
```

Figure 5.16: R code to execute MDS over matrices from chapter 4.

	X	positivity_Freq	ForwardLooking_Freq	Constraining	Uncert1_Freq	Litiguous	negativity_Freq	Modal.Weak
1	f adelphia 1999.txt	2.27	2.29	0.91	1.36	1.18	1.13	0.40
2	f Adelphia 2000.txt	2.27	2.30	0.92	1.34	0.91	1.24	0.40
3	f Anicom Inc 1998.txt	3.70	2.59	0.35	1.36	0.14	1.46	0.19
4	f Anicom Inc 1999.txt	3.57	3.83	0.44	1.59	0.23	1.79	0.57
5	f Applied Microsystems 2001	2.12	3.99	0.50	2.47	0.33	3.23	0.99
6	f Applied Microsystems 2002	1.91	4.28	0.58	1.92	0.56	3.20	0.74
7	f Assisted Living Concepts 2008.txt	1.80	2.83	0.88	1.60	0.65	2.72	0.73
8	f Assisted Living Concepts 2009.txt	1.82	2.97	0.92	1.64	0.69	2.79	0.76
9	f Bally Gaming and Systems 2004.txt	2.16	3.17	0.94	1.53	0.91	1.75	0.61
10	f Bally Gaming and Systems 2005.txt	2.35	3.34	0.61	1.83	0.75	2.56	0.76
11	f Beazer Homes 2005.txt	2.70	4.13	1.14	2.35	1.04	2.96	0.99
12	f Beazer Homes 2006.txt	2.75	3.42	1.04	2.02	0.93	2.98	0.88
13	f BROOKE CORPORATION 2006.txt	2.37	3.13	0.54	1.86	0.68	2.56	0.75
14	f BROOKE CORPORATION 2007.txt	2.25	2.69	0.62	1.72	0.93	2.62	0.69
15	f Cabletron Systems Inc 2000.txt	2.70	4.27	0.30	1.92	0.18	2.17	0.63
16	f Cabletron Systems Inc 2001.txt	2.76	4.05	0.43	1.86	0.34	2.37	0.77
17	f CHINA NATURAL GAS Inc 2008	1.78	4.02	0.76	2.04	1.14	1.71	0.93
18	f CHINA NATURAL GAS Inc 2010	2.06	4.67	0.84	2.00	1.34	2.63	1.20

Figure 5.17: An excerpt from matrix based on custom dictionaries before MDA computation.

```
          1         2         3         4         5         6         7         8         9
1   0.000000  6.486494 104.27105 155.75031 225.16493 218.99000 142.01211 148.94096  83.27136
2   6.486494  0.000000 102.99498 153.55044 221.21678 214.47228 136.62800 143.61478  79.68721
3 104.271048 102.994984   0.00000  95.83421 226.19836 226.66171 175.54158 178.22166 120.47464
4 155.750309 153.550435  95.83421   0.00000 171.38033 170.43236 167.60813 164.12969 112.62200
5 225.164932 221.216779 226.19836 171.38033   0.00000  50.20082 116.44454 106.18555 152.08778
6 218.989997 214.472276 226.66171 170.43236  50.20082   0.00000 103.34947  92.11484 140.71350
7 142.012109 136.628000 175.54158 167.60813 116.44454 103.34947   0.00000  11.40195  90.81384
8 148.940961 143.614775 178.22166 164.12969 106.18555  92.11484  11.40195   0.00000  92.14732
9  83.271361  79.687207 120.47464 112.62200 152.08778 140.71350  90.81384  92.14732   0.00000
```

Figure 5.18: An excerpt from distance matrix that captures Euclidean distance between reports.

```
> head(fit$points,14)
         [,1]        [,2]
1   -181.8351 -128.044454
2   -181.3330 -125.195715
3   -168.3072 -121.847381
4   -192.3257  -37.489232
5   -232.6936   76.611239
6   -231.6611   69.190522
7   -205.9233  -20.164700
8   -207.4116  -10.338439
9   -195.9542  -50.298068
10  -203.9353   -1.082897
11  -216.4556   60.228510
12  -202.3973   19.458053
13  -197.6089  -10.839252
14  -191.5492  -30.680575
```

Figure 5.19: MDA computation that has captures the proximities between documents to 2D.

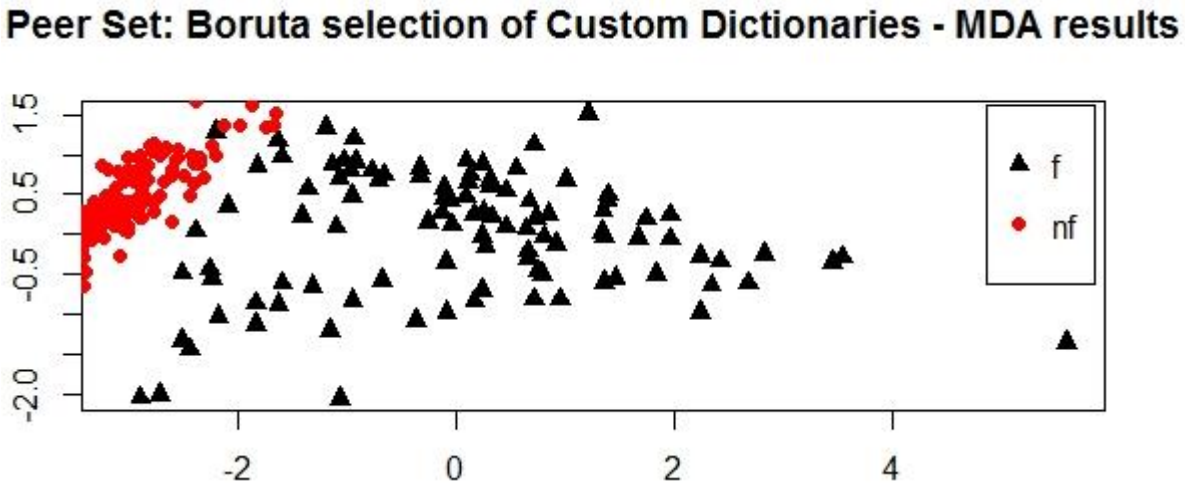


Figure 5.20: Distances between the 2 report categories as determined by MDA.

representation schemes as depicted in Chapter 4 are read in. After each matrix is read in, the variables that were deemed significant were extracted using the subset command. Only 4 shown in extract (custom dictionaries - line 3, Coh-Metrix indices - line 5, topics - line 8, linguistic based ratios (LBC) - line 13). Importantly, only one matrix read in at a time (more than one shown for illustrative purposes only). All class labels ('f' and nf' removed - line 15). Column one designated as header information (this contains firm names). Lines 17 to 19 ensures that all entries in matrix are numeric. An extract of the data frame, data2 in line19 is shown in Figure 5.17 for the values extracted from the custom dictionaries.

Line 20 then takes the matrix in data2 (this could be any of the matrices depicted in chapter 4) and generates the proximities table or the distance matrix, shown in Figure 5.18. This distance matrix computation is calculated using the Euclidean distance between points. This is done for all points in a matrix. Line 21 (Figure 5.16) then takes the distance matrix, d and runs the MDS algorithm, encapsulated in the cmdscale function. The reduced matrix generated that captures the distance between points and significantly enables better visualisation and representation of the distances is captured in the matrix 'fit' shown in Figure 5.19.

Rows 1 to 102 identify the fraud reports. Rows 103 to 408 identify the non-fraud reports. This separation of the data is performed in line 23 (Dim1) and line 24 (Dim2) of Figure 5.16. Line 25 onwards plots the data. The plot shown in Figure 5.20 is the peer set setup for the custom dictionaries document representation scheme.

As can be seen from the plot there is a possibility to separate out the fraud from the non-fraud reports. This will be tested in Chapter 6 when the classifiers are executed over the matrices.

5.7 Results from feature selection and MDS

The matrices devised from the document representation schemes described in chapter 4 were put through: PCA, Boruta and Information Gain to derive the optimal mix of features before classification. LSA was applied only to the n-grams, the features chosen and the results of MDA computation are shown in Appendix J.

The features selected by PCA for all the matrix combinations from chapter 4 are shown in Appendix K, Table K.1 to K.22. The tables in Appendix K also display the graphs that shows up the proximities between the two 'f' and 'nf' category of reports, as a result of MDA. Similarly the Boruta selected features are shown in Appendix L for all the matrices (Table L.1 to L.11). MDA is applied to the reduced features. Graphical displays are provided to show up the proximities between the two report categories.

Lastly Information Gain selected features are shown in Appendix M for all matrices (Table M.1 to M.11). The distances between the two report categories based on the features chosen by IG are also displayed.

5.8 Discussion and Conclusion

The thrust of this chapter has been to select those features that are rich in discriminatory information with respect to the classification problem at hand. This is a crucial step in the design of any classification system, as a poor choice of features results in poor classification. According to Chandrashekar and Sahin [260] selecting highly informative features is an attempt:-

- to place classes in the feature space far apart from each other (large between-class distance).
- to position the data points within each class close to each other (small within-class variance).

As indicated in the introduction, the dimensionality of the data involved in machine learning and data mining tasks has increased explosively. Such high dimensional data known as the curse of dimensionality is difficult to harness for predictive analysis. Unchecked such dimensionality results in the machine learning based classifiers overfitting the model with a degenerate performance on the prediction task [252]. This emphasises again the need for the feature reduction process undertaken in this chapter. The aim was to choose a small subset of the relevant features from the original ones according to certain relevance evaluation criterion, which usually leads to better learning performance (for example higher learning accuracy for classification), lower computational cost, and better model interpretability [252].

The first feature reduction techniques applied was Latent Semantic Analysis (LSA) as described earlier involved analyzing documents to find the underlying meaning or concepts. If each word only meant one concept and each concept was only described by one word, then LSA would be easy since there is a simple mapping from words to concepts. However, English has different words that mean the same thing (synonyms), words with multiple meanings, and all sorts of ambiguities that obscure the concepts. LSA is a technique as described that seeks to get to the root meaning from a choice of words. It was executed using scikit-learn package in python over an amalgam of the n-gram matrices (unigrams, bigram, trigrams). An extract of concepts identified in non-fraud reports are shown in Appendix I (Table I.2, I.3 and I.4). Concepts identified in fraud reports are shown in Appendix I (Table I.1). To get a clearer picture of the terms in concepts that were present in the non-fraud but not in the fraud reports, Table

I.6 and Figure I.2 were put together. The terms that appear to be significant in their non-identification in the fraud reports were for example: '*higher*', '*lower*', '*investment*', '*risk*', '*decrease*', '*operating income'*, '*compared*'. In particular, the term '*communities*' seem to be a key differentiator in the non-fraud reports. Conversely Table I.5 and Figure I.1 shows terms that were present in fraud and not in non-fraud reports. The terms '*acquisitions*', '*accounting*', '*control*', '*debt*', '*procedures*', '*required*', '*securities*' seem to be significant identifiers by LSA in fraud reports. In particular when comparing tf-idf scores terms such as: '*acquisitions*', '*procedures*', '*required*', '*securities*' seem to have variation between the fraud and non-fraud reports. Figure I.1 shows the number of times the terms graphed appeared in concepts mentioned in fraud reports (all term/concept pairing for fraud reports identified by LSA shown in Table I.1). Terms such as '*may*', '*development*', '*value*', '*interest*', '*products*', '*capital*', '*loans*' seem to be meaningful in their variability between the fraud and non-fraud reports. Conversely Figure I.2 shows the number of times the terms identified in concepts by LSA appeared in non-fraud reports and compared their frequency with fraud reports. Terms such as '*products*', '*operations*', '*year'*, '*financial*', '*increased*' and '*management*' seem to show variability. However once their tf-idf scores from these terms identified in the fraud/non fraud reports by LSA only '*may*', '*development*', '*loans*' are the meaningful terms that retain a difference between the two category of reports.

Tf-idf scores were extracted for terms that have been identified above: '*communities*', '*acquisitions*', '*procedures*', '*required*', '*securities*', '*may*', '*development*', '*loans*' along with others that were deemed to be possible differentiators through LSA analysis for all the 408 reports (list shown in Table 5.2). Both a peer set and matched pair matrix was set up for these scores and these matrices were then put through MDS. The results are shown in Appendix J. As can be seen from the graphs produced that the terms do not seem to result in a good separation in distance between the fraud and non-fraud reports. However, some classifiers through their routines to separate the classes may be able to attain a reasonable separation, discussed in chapter 6.

Another dimensionality reduction technique, Principal Component Analysis (PCA) was also executed over the matrices derived from chapter 4. The main aim of this technique is to identify the strongest patterns in the data, it finds attributes (Principal Components) which meets certain criteria. The PCs are:-

- linear combinations of the original attributes.

- orthogonal to each other.
- capture the maximum amount of variation in the data.

The variability of the data is often captured by a relatively small number of PCs [273]. As indicated for all the document representation matrices identified in chapter 4 most of the variance was captured by the first PC. Most of the matrices had the graph shown in Figure 5.10 where the first PC captured most of the variance. Appendix K shows the variables identified by the first PC for all the matrices. These variables shown were then used to form new reduced matrices and put through MDS with the results shown below each graph that depicts the PC captured.

For n-grams the best separation of the classes seems to be attained by unigrams, although both as can be clearly seen by the peer set data setup there seems to be enough separation in distance for both the bigrams and trigram to enable a good classification. Linguistic Based Cues (peer set), concepts (peer set), keywords (peer set) and Rutherford-keywords as can be seen from Appendix K are showing up a clear difference in distance using the PCA identified variables shown. All the other matrices have a visible separation as well. The classifiers in chapter 6 will provide the necessary corroboration of these separations visible using MDS.

As indicated in this chapter the two main feature selection types covered are wrappers and filters. The former are feedback methods which incorporate the ML algorithm in the FS process, they rely on the performance of a specific classifier to evaluate the quality of a set of features (Janacek et al, 2014). Boruta [270] designed as a wrapper around a Random Forest classification algorithm was used over the matrices. It iteratively removed the features which were proved by a statistical test to be less relevant than random probes [270]. The features identified by Boruta for each matrix from chapter 4 are shown in Appendix L. These features are then used to form new reduced matrices for both the peer set and match pair data set up. These reduced matrices are then put through MDS to gauge distance between the fraud and non-fraud reports. From the results shown in Appendix L, unigrams (peer set), custom dictionaries (peer set) and key terms (peer set) are showing a clear separation between the fraud and non-fraud reports. All the other matrices as can be seen again are showing up the fraud and non- fraud reports as apart with varying distances. Again the final proof will be with the classifiers, the results from which will inform how apart and how separable the two classes are for these Boruta reduced matrices.

Finally, a filter based FS method Information Gain is utilised. Information Gain based methods are among the most representative algorithms of the filter models. Appendix M gives the results of applying IG from the FSelector package in R. MDS shows up that the clearest distinction is attained by the features selected for the custom dictionaries, keywords and keywords (Rutherford [138]) peer set matrices. All the others show less separation but still there could be a pattern that is separate and distinct enough that could be picked up by the classifiers in chapter 6.

Thus so far in the framework, the fraud firms and matching non-fraud firms were identified and their 10-K/annual reports collated and cleaned. This formed the corpus. Its contents investigated in chapter 3, resulting in identification of some salient features. The next process in the framework was feature extraction shown in chapter 4. These feature were identified as potential linguistic correlates of deception from literature review conducted in chapter 2. Document representation schemes shown in chapter 4 transformed the data into a state that could be a starting point for a machine learning model. However, given the need for dimensionality reduction outlined in this chapter feature selection was performed on the matrices identified in chapter 4. Now at the end of this chapter all these matrices have been reduced as illustrated in Appendix K (using PCA), Appendix J (only n-grams using LSA), Appendix L (using Boruta), Appendix M (using Information Gain). All the results from feature selection outlined in this chapter indicate that there is a separation in distance as measured by MDS that could results in successful classification. All these permutations of the original matrices will now be passed in chapter 6 to the final stage of actual classification.

Chapter Six

THE CLASSIFERS

"For nothing ought to be posited without a reason given, unless it is self-evident or known by experience or proved by the authority of sacred scripture"

William of Occam c.1320

6.0 Introduction

Thus far in the framework, the corpus has been composed, cleaned, pre-processed and features extracted. These features are then further funnelled to drop those that do not relate well to the outcome label (fraud or non-fraud). However, the main engine, the classifier that would separate out the fraud from the non-fraud reports is still missing. This chapter delineates the shape and form of a potential classifier.

As can be gathered thus far fraud is an intractable and complex problem, potential fraudsters can employ any number of linguistic techniques to hide their deceit. This was covered in Chapter 2 and summarised in Table 2.1. They invariably morph to alter their deception strategy to prevent detection [173]. Given the scale of rising textual data with the concomitant rise in text that can be laced in deceit, there needs to be an automated way to aid in such fraud detection in an accurate manner. This problem domain is ideally suited for a machine learning (ML) based approach for combat against this misconduct. This involves the use of algorithms that discover patterns in text. These patterns are then used to make prediction on new data. The algorithms are designed so that they learn from mistakes and learn new patterns without being explicitly programmed. Inferences are made from data based on a function determined by the ML based algorithms. From the financial fraud perspective, the salient feature is that these algorithms iteratively learns from new data, they independently adapt, therefore directly closing in on a key ploy of fraudsters, the ability to alter their deception. In the absence of a machine learning approach this would be a hugely challenging task [275].

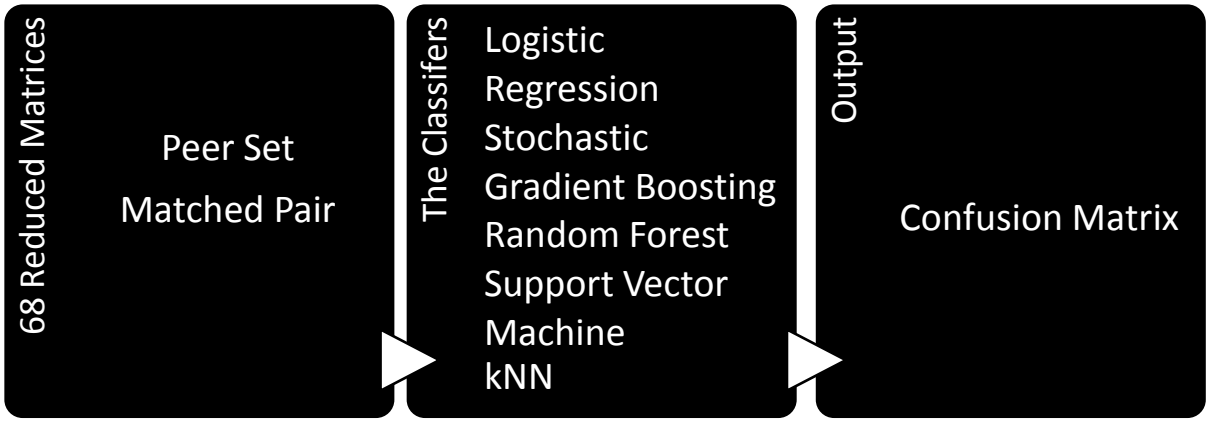


Figure 6.1: The Classification process as covered in this chapter.

Further as discussed in chapter 2, a number of researchers have proposed statistical and machine learning methods to detect financial fraud effectively. As indicated previously in chapter 2, FSF from linguistic analysis is a less researched field.

The main contents of this chapter at a high level of abstraction is shown in Figure 6.1. The 68 matrices have been derived from the process delineated in chapter 4 and 5. The 10 document representation schemes produced data set-ups of 2 types, matched pair and peer set, resulting in 12 matrices. These matrices were then put through 3 feature selection routines. This resulted in a total of 66 matrices. Additionally, for the n-grams (unigrams, bi-grams and tri-grams combined) Latent Semantic Analysis was executed, giving an additional 2 matrices. In total therefore there are 68 matrices (34 matched pair and 34 peer set). They are then put through the classifier models set up and outlined in this chapter. The end results would be a confusion matrix that output all the main performance indicators on the models.

Additionally, for each document representation scheme the matrices for the peer set scenario will also be put through the k-means clustering algorithm to test performance of this technique in separating the fraud from the non-fraud reports.

The structure of this chapter is as follows. First an overview of the machine learning process will be mapped out, with emphasis on the pitfalls and the main performance indicators. Thereafter the 5 learning algorithms used to build the classifier models will be delineated. The penultimate section will outline the k-means technique and discuss results. The chapter will close with a discussion and a conclusion section.

6.1 General Overview of Machine Learning

Essentially machine learning is rooted in statistical learning. James et al. [276] provide a motivating example, applied in a simplistic way to the problem domain under study is described below.

Given an annual report narrative from a company deigned to be fraudulent denoted by *Y* with X_i = (X_{i1},..., X_{ip}) i=1…p. X_i being the observed linguistic features such as denoted by the matrices in chapter 4 and 5. If it is believed that there is a relationship between Y and at least one of the X's, the relationship could be modelled as:-

$$Y_i = f\,(X_i\,) + \varepsilon_i \qquad \text{Eq (6.1)}$$

Where f is an unknown function and ε is a random error with mean zero. The relationship between the predictor variable (X_i) and dependent variable Y_i could be approximated to be as depicted in Figure 6.2. The difficulty of estimating f will depend on the standard deviation of the ε's. The machine learning algorithms that are covered in this chapter attempt to do just that. The goal of such inductive machine learning is to take some training data and use it to induce a function f. Once f is estimated it can be used for prediction and inference. In the former case, a good estimate for f with low variance of ε would enable accurate predictions for the response variable *Y* based on a new value of *X*. The function should generalizes well to new data. In the latter case the relationship between the *Y* and *X*s should be be scrutinised. For example: "*which particular predictors actually affect the response? Is the relationship positive or negative? Is the relationship a simple linear one or is it more complicated?"* [276]. Figure 6.3 concretely shows the above reasoning in a machine learning context. A task (red box) requires an appropriate mapping – a model – from data described by features to outputs. Obtaining such a mapping from training data is what constitutes a learning problem (blue box) [215]. The learning algorithm aids in determining a function f that could approximate to the relationship observed in the training data. The function once constituted is known as the model or sometimes referred to as the final hypothesis. Choosing a representation for a model is tantamount to choosing the hypothesis space that it can possibly learn [277].

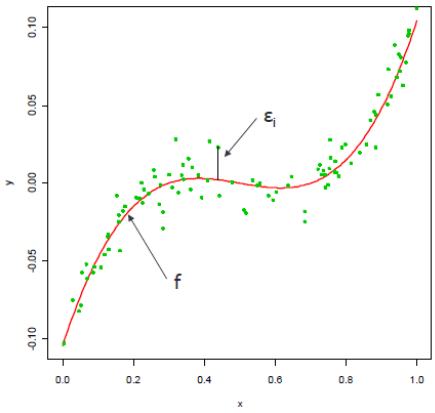


Figure 6.2: Fit function to data points.

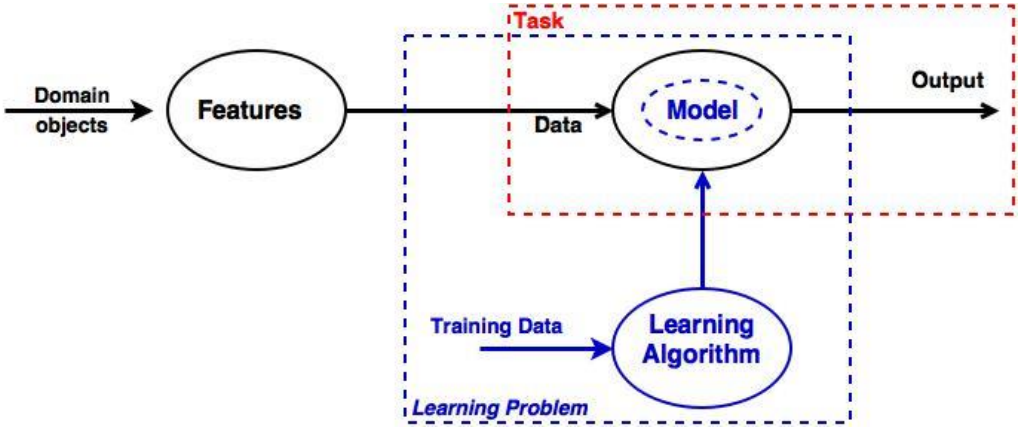


Figure 6.3: The machine learning process [215].

Typically a machine learning algorithm reduces the problem of estimating f down to one of estimating a set of parameters. Assumptions are made about the functional form of f typically chosen from the hypothesis space. The training data is then used to fit the model in other words to estimate f and the unknown parameters. The coefficients are adjusted to lower the ε's. This process is repeated over and over until the system has converged on the best values for the coefficients of the function. In this way, the predictor becomes trained, and is ready to do prediction. The iterative process of lowering the errors is often undertaken by using a "*measurement of wrongness*" [215]. The wrongness measure is known as the cost function or loss function. The choice of the cost function is another important piece of a machine learning process. In different contexts, being '*wrong*' can mean very different things.

In the function shown in Figure 6.3 the well-established standard is the linear least squares function. With least squares: "*the penalty for a bad guess goes up quadratically with the difference between the guess and the correct answer, so it acts as a very "strict" measurement of wrongness. The cost function computes an average penalty over all of the training examples*" [278]. According to Domingos [247] and given the above explanation for the model building process as outlined by Ng [278] all learning algorithms consist of three basic components, which are:

- Representation: A classifier (discussed below) must be represented in some formal language. The model consists of a set of classifiers that it can learn (hypothesis space). If a classifier is not in the hypothesis space it cannot be learned. Examples of representations include logistic regression, SVM, decision trees which will be covered later in the chapter.
- Evaluation: An evaluation function or an objective function is also needed to distinguish good classifiers from bad ones. The precision, recall and accuracy rates are what is used to measure performance of classifiers in this chapter.
- Optimization: is the process of minimising the cost function.

Classifiers typically come equipped with optimization techniques that perform such an operation. It determines the efficiency of the learner. Figure 6.4 shows an optimisation technique called gradient ascent attempting to minimise a loss function. It is trying to this by finding values for coefficients of the model that would produce a line that goes through the data that performs a good separation between the fraud and non-fraud data points.

A motivating factor in choosing the models used for classification was interpretability. Complicated models such as neural nets could have been used as they are known to fit a wide range of possible shapes of f [279]. However simpler approaches are easier to interpret and a good fit does not always lead to good prediction [276, 277]. The determining factor is how well the chosen model generalises. In other words how well does the function perform on new unexposed data, known as the test set, as opposed to the training set used to approximate f [215].

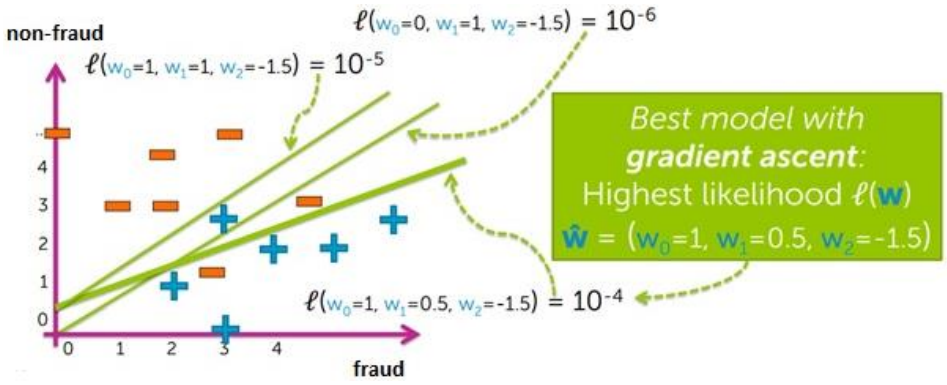


Figure 6.4: Fit function to data using a cost function to minimise errors [280].

6.1.1 Supervised Versus Unsupervised

As described above when f is induced from a portion of the data known as a training set. This is known as supervised learning. In the training set both the predictors X_i, (the column in the matrices shown in chapter 4) and the response, Y_i, (or labels – 'f' or 'nf') are observed. This function f will then be evaluated on the test data. The test data has no response variable which the induced f should predict. It has succeeded if it correctly predicts the labels. Figure 6.3 iIlustrates this process. A learning algorithm reads in training data and computes a learned function f. This function can then automatically label future text examples [281]. Supervised learning excels in applications where historical data predicts likely future events. Recent successful implementations in the area of financial fraud are detailed in chapter 2.

According to Liu [282] the foremost advantage of supervised learning is interpretability, the output is meaningful to humans. Disadvantages cited include the difficulty of labelling each record with class information, especially when there is a huge volume of input data. Also there is the difficulty of giving each record a distinctive label often there are uncertainties as to what class a record belongs to. However in the narratives under study there is no such ambiguity as firms that have been deemed fraudulent have been through a judicial process. Further as indicated in chapter 4, success in the supervised machine learning task hinge on choosing representative features [215]. In this thesis as detailed in chapter 4 a range of features were chosen in a bid to find the most representative of the reports.

Unsupervised learning is where the model has no class information during the

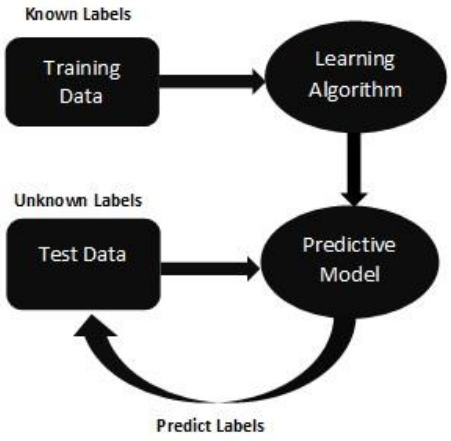


Figure 6.5: Supervised machine learning [283].

training. Only the data points or the vector of features are known without the accompanying class labels. However, the data points can be used to cluster the input data into classes on the basis of their statistical properties only. These inputs will be able to find the structure or relationships between different inputs. Valpolla [284] argues that in an unsupervised setting it is possible to learn larger and more complex models than with supervised learning. This is because in supervised learning there is a constraint in trying to find the connection between two sets of observations. The difficulty of the learning task increases exponentially in the number of steps between the two sets and that is why supervised learning cannot, in practice, learn models with deep hierarchies.

The most used unsupervised learning routine is clustering, which will create different clusters of inputs and will be able to put any new input in the appropriate cluster [276]. Clustering, is the main unsupervised learning techniques used over the corpus in this thesis and will be discussed and executed over the data in section 6.9. Figure 6.5 illustrated a general unsupervised based model [276]. As is the case for supervised learning, feature selection plays an important role for effective clustering, this reduces computational complexity and simplifies the subsequent process. According to Ahmed et al. [285] selection of a clustering algorithm is a vital step to cluster the underlying data. Similarity/dissimilarity measure or else known as proximity measure quantify how similar two data points are. A good clustering criterion leads to a partition that fits the data well. Therefore a: "*proximity measure and clustering criterion play a vital role in determining the accuracy of a clustering algorithm*" [285].

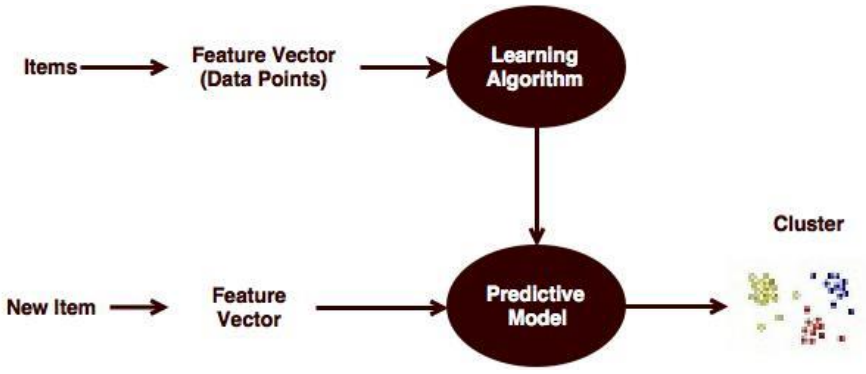


Figure 6.6: Unsupervised machine learning - clustering.

6.2 The Classification Task

Based on explanatory work outlined by James et al. [276] supervised learning problems can be further divided into regression and classification. The latter covers situations where *Y* is continuous/numerical, for example predicting value of stocks. Whereas the former covers situations where *Y* is categorical as is the case under study. The question addressed is based on a set of features from narrative sections of annual reports/10-K, is it possible to separate out a fraud firm from a non-fraud report?

Classification is one of the most widely used techniques in machine learning. It will be deployed in this chapter over the 68 matrices using 5 learning algorithms to determine its success at the discrimination task. The task of discriminating between a fraud and non-fraud firm boils down to a text classification task, which is a mapping process (as shown in Figure 6.7 and Figure 6.8, extracted from [280]). It is defined as: "*the task of automatically detecting one or more predefined categories that are relevant to a specific document*" [217]. In this case text is assigned to class fraud (f) or non-fraud (nf).

X = the firm narratives

C = f, nf of the possible classes

Classifier Y maps inputs to classes

Y: X → C

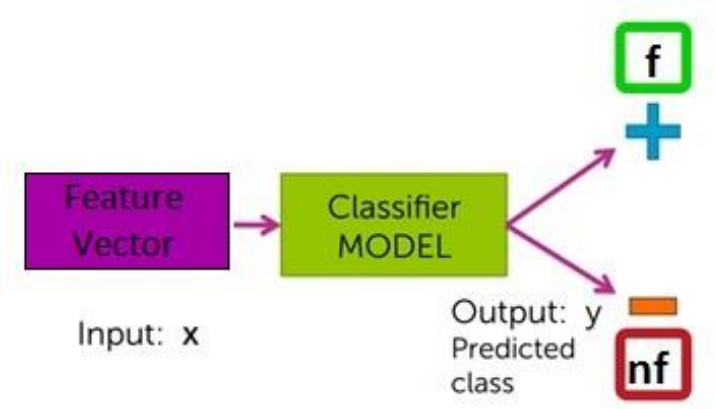


Figure 6.7: The classification task performed on matrices [280].

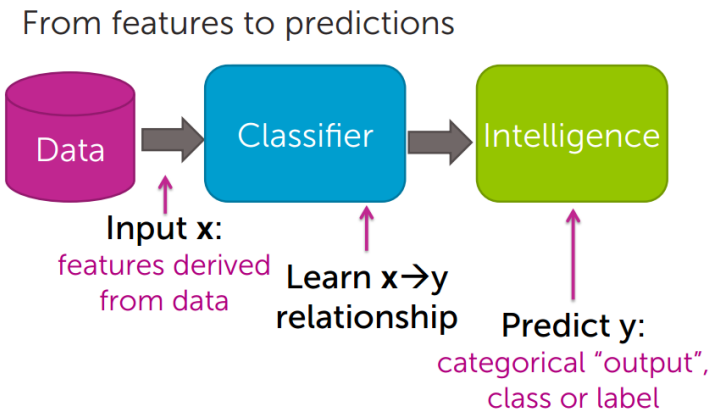


Figure 6.8: The classification task performed on matrices with more detail [280].

Goal:

Determine the function for this classifier with a training set D composed of n examples [217]

Thereafter, when confronting new text (which constitutes the test set, the classifier would identify the correct category (fraud or non-fraud). The classifier is typically developed using supervised machine learning techniques. A classification algorithm: "*is trained over a corpus of labelled documents in order to capture the most distinguishing category patterns that will be used to classify the new unlabelled instances*" [217].

The following process delineates how a class (fraud or non-fraud) is determined for a document.

Input:

a document *d*

a fixed set of classes *C = {c1, c2,…, cJ }* in this case there are only 2 classes

C= {f,nf}

a training set of m of labelled documents *(d1,f),(d2,nf)....,(d_m,c_m)*

Output:

A learned classifier: $y: x \rightarrow c$

Given a new observation x - a report the classification task is to find a classification function $y: x \rightarrow c$, which can predict the unknown class label *Y* of this new observation using available training data as accurately as possible [280]. The test of the learned classifier is whether it produces the class label ('f' or 'nf') for future examples. How well does the fraud classifier perform on previously unseen feature vectors extracted from new annual reports/10-K? This whole process with more detail than Figure 6.7 is shown in Figure 6.8 diagram.

According to Domingos [247] to access the accuracy of classification, a loss function is needed. This is defined as: "*price paid for inaccuracy of predictions in classification problems*" [286]. A commonly used loss function for classification is the zero-one loss is shown below. This notation means that the loss is zero if the prediction is correct and is one otherwise.

Binary classification:

zero/one loss $l(y, \hat{y}) = \{0\ if\ y = \hat{y}\ or\ 1\ otherwise$

As explained by Daumee [283] the job of l is to monitor how '*bad*' a system's prediction is in comparison to the truth. In particular, if y is the truth and $\hat{y}$ is the system's prediction, then $l(y, \hat{y})$ is a measure of error. This captures the notion of what is important to learn. Once the loss function is defined, Daumee [283] using a probabilistic model as an example further elucidates on the learning task. In this model it is assumed that there is a probability distribution $\mathcal{D}$ over input/output pairs, $\mathcal{D}$ is a distribution over (*x,y*) pairs. No assumption is made about what the distribution looks like. It simply defines what sort of data is expected. The training sample is a random sample of input/output pairs drawn from $\mathcal{D}$. Based on this training data a function f is induced that maps new inputs to corresponding predictions $\hat{y}$. The: "*key property of f is that it should do well (as measured by l) on future example that are also drawn from*

$\mathcal{D}$" [283]. Formally, it's expected loss ε over $\mathcal{D}$ with respect to l should be as small as possible:

$$\varepsilon \triangleq \mathbb{E}_{(x,y)\sim\mathcal{D}}\big[l\big(y, f(x)\big)\big] = \sum_{(x,y)} \mathcal{D}(x,y) l(y, f(x)) \quad \text{Eq (6.2)}$$

The difficulty in calculating the expected loss is that $\mathcal{D}$ is unknown, only a portion (the training set) is known. Therefore given a learned function f the training error or the average error over the training data can be computed. This can be reduced to zero as the data is known but it would be a poor predictor for new data. As Daumme [283] emphasises this is the main challenge faced in machine learning, the training error is known but the main drive is to lower the expected error. To keep that low the learned function must generalise beyond the training data to future data that is unknown. According to Daumme [283] amalgamating the explanation given so far, a formal definition of inductive machine learning could be posed as such:

"*Given (i) a loss function l and (ii) a sample $\mathcal{D}$ from some unknown distribution $\mathcal{D}$, a function f has to be computed that has low expected error ε over $\mathcal{D}$ with respect to l*".

6.3 Generalization and Overfitting

The main determinant of success at the machine learning task is to generalise beyond the examples in the training set. The established way to investigate success at the learning task is to set aside some of the available data as '*test'* data and check performance indicators (discussed in section 6.6). If all the available data is used to train the model then it will be able to identify all the relevant information in the training data, but will fail when presented with the new data. Training error (the ε's) will be low, the model is however incapable of generalizing, it is overfitting the training data.

In the Figure 6.9, top three diagrams we have data (x's) and models (dashed curves). From left to right the models have been trained longer and longer on the training data. The training error curve in the bottom box shows that the training error gets better and better as data is trained longer (increasing model complexity). The top right box shows a complex model that hits all the data points. This is a classic case of overfitting. This model performs well on the training data, but when presented with new data (examine the prediction error curve in the bottom box) then the model

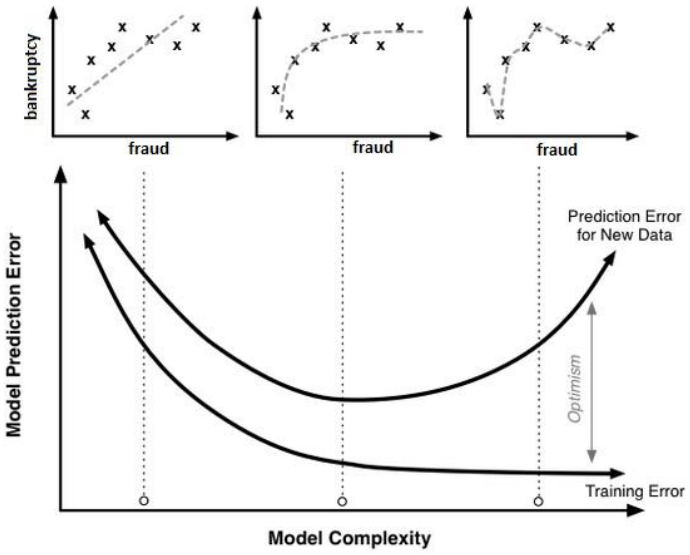


Figure 6.9: Relationship between training and testing error and model complexity [287].

shows its poor predictive ability [276]. Therefore to create good predictive models in machine learning that are capable of generalizing, the model should be trained to a degree that inhibits overfitting. Less common is under-fitting this refers to a model that can neither model the training data nor generalize to new data and can be detected easily as it will have poor performance on the training data.

There are two important techniques that can be used when evaluating machine learning algorithms to limit overfitting [288]:

- Use a resampling technique to estimate model accuracy.
- Hold back a test dataset.

The most popular resampling technique is *k*-fold cross validation. It allows you to train and test your model k-times on different subsets of training data and build up an estimate of the performance of a machine learning model on unseen data.

In the test set method randomly choose 30% of the data to be in a test set. The remainder is the training set. Perform classification on the training set then estimate classifier performance with the test set.

From the literature some recent attempts to improve generalisation/reduce overfitting include that of Janpuangtong and Shell [289]. They propose a framework that allows a novice to create a model from data easily by helping structure the model building process and capturing extended aspects of domain knowledge. The capture of domain knowledge would be undertaken by an ontology that enables selection of relevant

features. Their results indicate making use of the ontology, helps to improve model generalization.

Closely related to the concept of overfitting and under-fitting in a model are the twin concepts of bias and variance. According to Hastie el al. [276] bias refers to the error that is introduced by modelling a real life problem (that is usually extremely complicated) by a much simpler solution. For example, linear regression assumes that there is a linear relationship between *Y* and *X*. It is unlikely that, in real life, the relationship is exactly linear so some bias will be present. The more flexible/complex a method is the less bias it will generally have. Therefore, the error due to squared bias is the amount by which the expected model prediction differs from the true value or target, over the training data. Through repeated model building the average prediction values can be calculated. If these average prediction values are substantially different than the true value, bias will be high [276].

In contrast, variance refers to how much your estimate for f would change by with a different training data set. Generally, the more flexible a method is the more variance it has. As explained by Manning et al. [49], variance measures how inconsistent are the predictions from one another, over different training sets, not whether they are accurate or not.

Models that exhibit small variance and high bias underfit the truth target. Models that exhibit high variance and low bias overfit the truth target. In reality if the data is non-linear but a linear model is chosen to induce a function then bias is introduced resulting from the linear model's inability to capture nonlinearity. The linear model is underfitting the nonlinear target function over the training set. Similarly, if data is linear but a nonlinear model to chosen to approximate it, then bias is introduced from the nonlinear model's inability to be linear where required. In fact, the nonlinear model is overfitting the linear target function over the training set [276].

The "tradeoff" between bias and variance can be framed as such – a learning algorithm with low bias must be 'flexible' so that it can fit the data well. But if the learning algorithm is too flexible it will fit each training data set differently, and hence have high variance. A key characteristic of many supervised learning methods is a built-in way to control the bias-variance tradeoff either automatically or by providing a special parameter that can be adjusted [290].

Figure 6.10 plots the model's performance using prediction capability on the vertical axis as a function of model complexity on the horizontal axis. Gutierrez [290] depicts

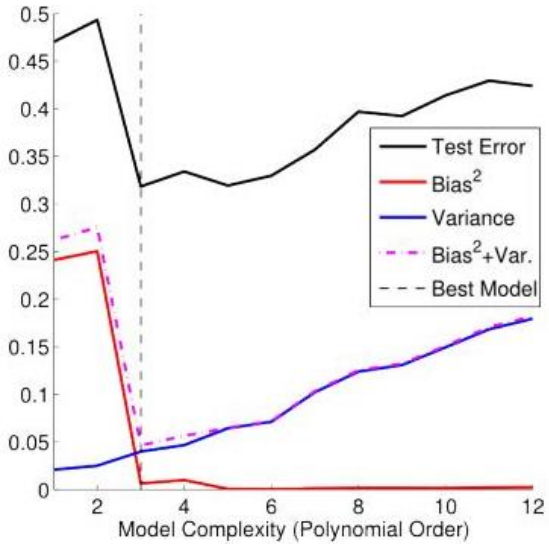


Figure 6.10: Bias, variance trade-off [290].

a case where a number of different orders of polynomial functions are used to approximate the target function. Shown in Figure 6.10 are the calculated square bias, variance, and error on the test set for each of the estimator functions.

It can be seen that as the model complexity increases, the variance slowly increases and the squared bias decreases. This points to the trade-off between bias and variance due to model complexity, i.e. models that are too complex tend to have high variance and low bias, while models that are too simple will tend to have high bias and low variance. The best model will have both low bias and low variance [290].

6.4 The curse of dimensionality

Adding more features to a classifier does not improve its performance. Increasing the dimensionality of the problem by adding new features would actually degrade the performance of the classifier. This is illustrated by Figure 6.11, and is often referred to as '*The Curse of Dimensionality'*. The term was first introduced by Bellman [291] and researchers are still working to control its detrimental effects on predictive models today [292].

As can be seen from Figure 6.11 as the dimensionality increases, the classifier's performance increases until the optimal number of features is reached. Further

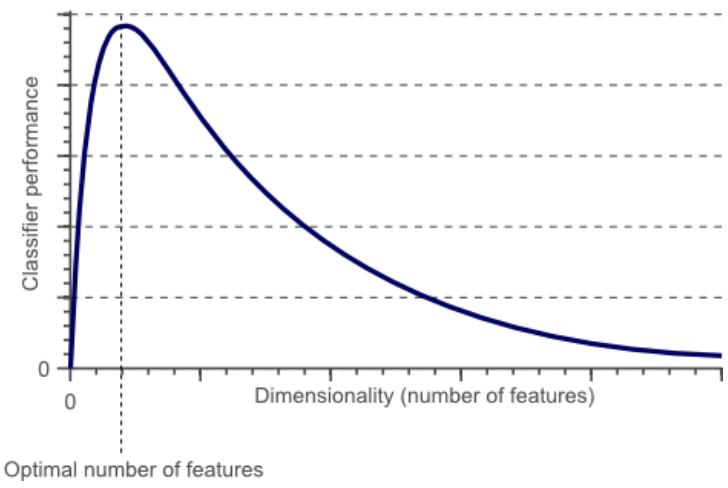


Figure 6.11: Trade-off between classifier performance and dimensionality [293].

increasing the dimensionality without increasing the number of training samples results in a decrease in classifier performance. As more features are added the dimensionality of the feature space grows and becomes increasingly more sparse [293, 294]. This facilitates the identification of a separable hyperplane. According to Spruyt [293] when viewed in 2D a different picture emerges the classifier seems to learn specific instances and exceptions. It has over-fitted. If more dimensions are added the amount of training data also needs to grow to maintain the same coverage and avoid overfitting.

Further, Bishop [295], Spruyt [293] maintain that the variance of a parameter estimate increases if the number of parameters to be estimated increases as would happen with increasing number of features. This means that the quality of the parameter estimates decreases if the dimensionality goes up, due to the increase of variance. An increase of classifier variance corresponds to overfitting.

Additionally, distance measures are also rendered less meaningful to measure dissimilarity in highly dimensional spaces. Classifiers depend on these distance measures (e.g. Euclidean distance, Manhattan distance) for successful discrimination between the classes. Spruyt [293] maintains that there is no fixed rule that defines how many features should be used in a classification problem. It is dependent on amount of training data, the decision boundaries and the type of classifier used. Therefore number of features that mitigates overfitting as a consequence of increases in dimensionality is context dependent, however as rule of thumb as more features are

added more data should follow. The smaller the size of the training data, the less features should be used [293]. However this has to be balanced with the type of classifier used. Spruyt [293] maintains that if a classifier is used that generalizes easily for example naive bayes or linear classifier, then the number of used features can be higher since the classifier itself is less expressive.

Other ways that are commonly used in the battle against increased dimensionality is to partake in feature selection to weed out those features that do not contribute to the classification task. Another way would be to engage in feature engineering and combine features. Finally, cross validation approaches that split the original training data into one or more training subsets is another armoury against overfitting. During classifier training, one subset is used to test the accuracy and precision of the resulting classifier, while the others are used for parameter estimation. If the classification results on the subsets used for training greatly differ from the results on the subset used for testing, overfitting is in play [276, 293].

6.5 The caret package and resampling

The caret package in R: "*contains functions to streamline the model training process for complex regression and classification problems*" [296]. It has a unified interface for modelling, prediction and tuning using resampling. The general flow of model building is as follows [296]:-

1. Create the model using the train function.
2. Assess the properties of the model.
3. Predict outcomes for samples using the predict function.
4. Check performance using confusion matrix function

The main functions that are used for model building are outlined in Appendix Y, Figure Y.2.

The function createDataPartition is used to create balanced splits of the data. The *y* argument is a factor with '*f*' or '*nf*' value. The random sampling is enacted within each class and preserves the overall class distribution of the data. For all matrices used split of 75%\25% is deployed to keep to a standard that is used within for model building for classification tasks [296, 297].

The output of the function *train* is an object of class '*train*'. This is the fitted model with the tuning parameter values selected by resampling. It chooses an optimal model from the parameter tuning process performed through resampling. It also estimates model performance from a training set [296]. This process is described below extracted from Kuhn [296].

Define sets of model parameter values to evaluate;
for each parameter set ***do***
    for each resampling iteration ***do***
        Hold–out specific samples ;
        Fit the model on the remainder;
        Predict the hold–out samples;
    end
    Calculate the average performance across hold–out predictions
end
Determine the optimal parameter set;

The *train* function chooses the model with the largest performance value by adjusting the cost function. The *train* function allows the user to specify alternate rules for selecting the final model. The argument *selectionFunction* can be used to supply a function to algorithmically determine the final model. Breiman et al. [298] suggested the "*one standard error rule*" for simple tree-based models. According to Kuhn [296] in this case, the model with the best performance value is identified and using resampling, the standard error of performance can be estimated: "*the final model used was the simplest model within one standard error of the (empirically) best model. With simple trees this makes sense, since these models will start to over-fit as they become more and more specific to the training data*" [296].

The *train* function can also take in a parameter *preProcess* that pre-process the data in various ways prior to model fitting. For the model building conducted in this study mostly this parameter is given the value *center* and *scale*. The *center* command subtracts mean from values and *scale* divides values by standard deviation [296].

The *trainControl()* function is used to create a set of configuration options known as a control object, which can be used with the *train()* function. These options allow for the management of model evaluation criteria such as the resampling strategy and the measure used for choosing the best model. Two important parameters typically set using this function are: *method* and *selectionFunction*. The *method* parameter is used to set the resampling method, such as holdout sampling or k-fold cross-validation.

The selectionFunction can be used to choose a function that selects the optimal model among the various candidates. Three such functions are included. The *best* function simply chooses the candidate with the best value on the specified performance measure. This is used by default. The other two functions are used to choose the most parsimonious (that is, simplest) model that is within a certain threshold of the best model's performance. The *oneSE* function chooses the simplest candidate within one standard error of the best performance, and *tolerance* uses the simplest candidate within a user-specified percentage. An example of how the *trainControl()* used is shown in Appendix Y, Figure Y.1.

This train object can then be used in the traditional way to generate predictions for new samples, using that model's *predict* function [296]. To predict the class of new samples, *predict* function is used. The predict function with the train object will generate predictions. For classification models, the aim is to calculate the predicted class.

For the classification task, 75% of the data was used for training and building the classifier. The remaining 25% was used to test the accuracy of the classifier. The training set is used: "*to estimate model parameters*" [299]. Whilst the test set is: "*used to get an independent assessment of model efficacy*" [299]. The test set is not used during model training.

To combat overfitting, resampling of the data used for training is undertaken. Resampling methods: "*tries to 'inject variation' in the system to approximate the model's performance on future samples*" [296]. Through this method it can be gauged when poor choices are made for parameter values for the determined function. When calling the *trainControl* function the type of resampling used has to be specified. In all models, the widely used repeated *k*-fold cross validation is deployed (3 separate 10 fold cross-validations is set). In this approach, the samples are randomly partitioned into *k* sets (called folds) of roughly equal size. A model is fit using all the samples except the first subset. Then, the prediction error of the fitted model is calculated using the first held-out samples. The same operation is repeated for each fold and the model's performance is calculated by averaging the errors across the different test sets [276]. This enables, in the absence of a large test set, an estimate of the test set prediction error.

This process as used in Caret is described by Kuhn [296] is as follows:-

1. Randomly split the data into *k* distinct blocks of roughly equal size.

2. Leave out the first block of data and fit a model.
3. This model is used to predict the held-out block
4. Continue this process until all *k* held–out blocks predicted.

The final performance is based on the hold-out predictions. *K* is usually taken to be 5 or 10 and leave one out cross–validation has each sample as a block. Repeated *k*–fold CV creates multiple versions of the folds and aggregates the results.

In sum, the data is: "*divided into k equal parts, one k part is left out, the model is fitted to the other k-1 parts (combined) and predictions obtained for the left out the kth part, this is done for each part*" [299]. An overall accuracy estimate is provided. This approach shakes up the data, however, each *k* is only as big as the original training set and prediction error could be biased upwards. Cross-validation is one of the most widely used method for model selection, and for choosing tuning parameter values [300]. This is confirmed by a recent study by Wong [301] on resampling and closely allied technique leave-one-out cross validation. The author considers factors to investigate the usage of k-fold cross validation. The factors include the number of folds, the number of instances in a fold, the level of averaging, and the repetition of cross validation.

6.6 Performance Indicators

A battery of measures are taken to provide a comprehensive outlook on classifier performance. The best overall performing classifiers in both the peer set and matched pair data sets are in Appendix X, Tables X.1 and X.2. The results shown are generated from the confusion matrix command in R: "*a 'confusion matrix' is a cross–tabulation of the observed and predicted classes*" [296]. A basic definition of these metrics is outlined below and are also described in Kuhn [296].

Accuracy (ACC)

The number of correct predictions from all predictions made.

True positive

A fraud report correctly classified as a fraud report.

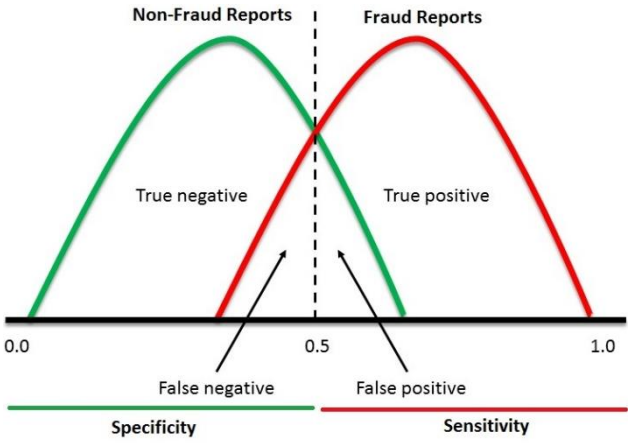


Figure 6.12: Relationship between classifier performance criteria [280].

True negative

A report is not fraudulent and is correctly classified as a non-fraud report.

False positive

A report is a non-fraud report but is incorrectly classified as a fraud report.

False negative

A report is a fraud report but is incorrectly classified as non-fraud report.

Sensitivity

Given that a result is truly a fraud report, what is the probability that the model will predict a fraud report? In other words the proportion of fraud reports, correctly identified. This is also known as recall.

$$Sensitivity = \frac{\#\text{ f reports predicted to be f}}{\#\text{ true f reports}}$$

Specificity

Given that a result is truly not a fraud report, what is the probability that the model will predict a negative results? In other words, the proportion of non-fraud reports, correctly identified.

$$Specificity = \frac{\#\text{ true nf reports to be nf}}{\#\text{ true nf reports}}$$

These conditional probabilities are directly related to the false positive and false negative rate of a method.

The relationship between sensitivity, specificity, true positive, true negative, false positive, false negative is captured in Figure 6.12.

No Information Rate (NIR)

Largest proportion of the observed classes. In the peer set scenario there were more non-fraud reports than fraud reports in the corpus and therefore more non-fraud in the test cases. Whereas in the matched pair design, there were equal numbers of fraud and non-fraud reports.

P Value (ACC > NIR)

A hypothesis test is computed to evaluate whether the overall accuracy rate is greater than the rate of the largest class. P values lower than 0.05 indicate a significant difference.

Pos Pred Value (PPV)

The percent of predicted positives (fraud) that are actually positive. In other words, it is the probability that a report designated as fraudulent is truly fraudulent. This is also known as precision.

Neg Pred Value (NPV)

The percent of negative positives (non-fraud) that are actually negative. Again, it can be expressed as the probability that a report designated as non-fraudulent is truly non-fraudulent.

Balanced Accuracy

Arithmetic mean of sensitivity and specificity values.

Kappa

A metric that compares observed accuracy with expected accuracy (random chance). Therefore a measure of prediction performance of classifiers.

It takes into account the expected error rate:

$$k = \frac{O - E}{1 - E}$$

where O is the observed accuracy and E is the expected accuracy.

The area under a ROC curve quantifies the overall ability of the classifier to discriminate between those reports that are fraudulent and those that are not. A poor classifier (one no better at identifying true positives than flipping a coin) has an area

of 0.5. A perfect classifier (one that has zero false positives and zero false negatives) has an area of 1.00. Most classifiers have an area between those two values.

Given the above definitions, a well performing classifier would have higher kappa values, higher sensitivity and (PPV) scores as they are complimentary, higher specificity and NPV scores (also complimentary), higher accuracy (ACC) and balanced accuracy score (again complimentary) and low p values (ACC>NIR). Figure 6.12 depicts the relationship between sensitivity, specificity, true positives values and true negatives. Given a binary problem like fraud, there are four potential classification outcomes: (1) true positive, a fraud firm is correctly classified as a fraud firm; (2) false negative, a fraud firm is incorrectly classified as a non-fraud firm; (3) true negative, a non-fraud firm is correctly classified as a non-fraud firm; and (4) false positive, a non-fraud firm is incorrectly classified as a fraud firm. False negative and false positive classifications are associated with different misclassification costs [180]. The false positives plus false negatives divided by the total number of examples constitutes the classification error. As can be seen from Figure 6.12 false positives and false negatives adversely impact the sensitivity and specificity values.

Results are only shown for the classifiers used from the caret package in Appendix N to W. For the peer set scenario, the classifiers were trained on 307 reports. The 101 remaining reports were used as test cases against the trained classifier to predict report class (fraud or non-fraud). For the matched pair design scenario, 153 reports were used to train the classifier, leaving 51 reports to be used as test cases. This prediction is based on the learning functions derived using a training set by the classifiers shown in the tables (Appendix N to W).

6.7 Imbalanced data sets

According to He and Garcia [302] given the data explosion it is a call of the time to advance: "*the fundamental understanding of knowledge discovery and analysis from raw data to support decision-making processes*". The thrust of this thesis is to show that such techniques can be used successfully in real world applications. However, a fundamental assumption underlying these techniques is that data should be balanced but in most real word application this does not apply. Most real world application including in the financial reporting domain, the data is more likely to be imbalanced

[303, 304]. Specifically, with reference to classification, imbalanced data refers to a problem where the classes are not represented equally. This is the case with the corpus under study, there are 102 fraud reports matched with 306 non fraud reports, resulting in a 3:1 ratio. This is reflective of a real world scenario as most reports are not likely to be fraudulent. Most classification data sets do not have equal number of instances in each class [302].

Typically, the first performance measure examined in a classification model is classification accuracy which is the number of correct predictions from all predictions made. In an unbalanced data set this measure can be very misleading. For example if 90% of the data belongs to class 1 and 90 % classification accuracy is attained then this result should be subject to further examination. The accuracy is likely to be reflecting the underlying class distribution. The model is likely once calculating out this distribution has just predicted the one class. This is known as the accuracy paradox [305].

A number of tactics are recommended for use when dealing with unbalanced data sets in a classification task [302]. The most obvious and simplest to implement is to look at a number of metrics that give more insight into the accuracy of the model not just the classification accuracy. In particular closer examination of sensitivity, specificity, balanced accuracy, kappa and ROC curves can indicate how the lower number of fraud reports are handled by the classifier. Other techniques commonly applied include manipulation of the resampling by oversampling by adding more instances of the fraud set or under-sampling by deleting instances of the non-fraud set. Brownlee [306] also recommends using a number of different algorithms to gain greater insight into model performance on an imbalanced data set.

The approach employed in this study to deal with the imbalanced data is to establish two types of data setups. The peer set which is imbalanced at a ratio of 1 fraud report to 3 non-fraud reports is established and the learning algorithms executed over this composition. The model performance on this data set is compared to a balanced setup known as matched pair. These two types of setup provide an indication of the effect of imbalanced data on model performance. This would be equivalent to down-sampling. The outcome of this is that the class frequencies match the least prevalent class. This is the case in the matched pair scenario 2/3 of the non-fraud companies are not included in model building. A number of learning algorithms will be executed

over the two types of data set up and model performance will be measured using a number of metrics outlined in section 6.6.

As indicated imbalanced sets reflect the unequal composition of the data in real world applications. In the area of FSF detection using linguistic features Goel [37], Throckmorton et al. [169], Purda and Skillicorn [168] accounted for this imbalance by setting up a peer set (see Appendix A, Table A.2 and A.3). Classifier performance was judged through an overall balanced accuracy score. There are a number of financial based predictive modelling applications set up that account for imbalanced data sets. Some recent work outlined below.

Danenas and Garsva [307] describes an approach for credit risk evaluation based on linear Support Vector Machines classifiers. Financial ratios are extracted from 10-K reports from the EDGAR database. They deal with the imbalanced data sets through judicious manipulations of the cost function.

Sanz et al. [308] propose a system that allows obtaining good prediction accuracies using a small set of short fuzzy rules and according to the authors implying a high degree of interpretability. These rules also effectively deal with issues surrounding imbalanced datasets with no need for any pre-processing or sampling method and, thus, avoiding the accidental introduction of noise in the data used in the learning process.

Kim et al. [177] develop multi-class financial misstatement detection models to detect misstatements with fraud intention. They extract financial ratios from a variety of sources on company data. They use three classifiers as predictive tools to detect and classify misstatements according to the presence of fraud intention. They deal with class imbalance by again manipulating the cost function.

Lima and Pereira [309] apply machine learning based classification approaches to fraud detection in web transactions. They confirm that fraud detection work is characterised by large imbalance between the classes (fraud or non-fraud). They find that imbalance between the classes reduces the effectiveness of feature selection and deploy an under-sampling strategy to improve final classification results. They find that under-sampling does indeed lead to improvements in classifier performance.

6.8 The Classifier Models

Extensive results on the classifiers are shown in Appendices N to W. Each appendix shows the results of the trained model on the test set as given by a confusion matrix (this summarizes the results of a classification model) on the document representation schemes delineated in chapter 4. This is shown for features chosen by each of the features selection routines (Principal Component Analysis, Boruta and Information Gain) for both the peer set and matched pair set up. The caret package also includes functions to characterize the differences between models generated using their sampling distributions. Recall that resampling is based on repeatedly drawing samples from a training set of observations and refitting a model on each sample in order to obtain additional insights into that model. In each appendix, each time a model is trained on a feature set/document representation scheme/data set up a box plot is drawn to show the results of the training process using cross validation. These plots further show up and reinforce any performance differences in the model brought forth from the confusion matrix results from test set. Appendix X shows the best performing classifiers/feature selection routines for each document representation scheme.

Another aspect of the models described below and often used to better classification results are boosting and bagging techniques. Such techniques create ensemble classifiers. Boosting involves consecutively for a set number of times randomly selecting a subset of training samples without replacement and training learners that are weak. These learners are combined to create a model that should make better predictions. Instances that were misclassified by the previous learners are given more weight so that subsequent learners give more focus to them during training. Bagging is based on bootstrapping which is random sample with replacement. A number of such samples are generated. The algorithm chosen is trained on each of these samples separately. The predictions made are averaged at the end. Bagging is recommended for reducing variance, whereas boosting is used to reduce both bias and variance [295].

A notable result is the near perfect/perfect accuracy attained by all models using unigrams as features. This finding affirms previous text mining research that has used unigrams and in some cases combined with bigrams and trigrams. This was the case by a study conducted by Jarvis et al. [310] on a corpus constructed of speech

transcripts to determine if speech was from a native or a non-native speaker. They attained classification accuracy of 90-100 %. In the financial fraud domain Goel [37] attained accuracy of 89 % using a bag of words (unigram) model (Appendix A, Table A.3). Fitzgerald et al. [311] also find using a bag of words (n-grams) to correctly identify source of file fragments that classification accuracy can be as high as 99 %. In the corpus under study this has resulted from a judicious selection of unigrams. The features were identified through the corpus linguistic methodology identified in chapter 3. From that process it was clear that there was a divergence in the use of some words between fraud and non-fraud reports. Therefore, it is unsurprising that they attained perfect results.

6.8.1 Classifier Tuning

The machine learning models built using the document representation schemes were fine-tuned. Versotek [312] argues that rather than choosing arbitrary values for each of the model parameters, it is better to conduct a search through many possible values to find the best combination. The caret package provides tools to assist with automatic parameter tuning. Each model comes with unique parameters to adjust. The data and consecutive results produced determine how extensively they should be tuned to find the optimal settings. The classifier models used and the parameters that can be tuned is shown in Table 6.1. Only those parameters listed in the table below are supported by caret for automatic tuning.

The goal of automatic tuning is to search a set of candidate models comprising a matrix, or grid, of possible combinations of parameters. Versotek [312] recommends that as it is impractical to search every conceivable parameter value, only a subset of possibilities is used to construct the grid. By default, caret searches at most three values for each of p parameters, which means that 3^p candidate models will be tested.

Model tuning is often undertaken in caret using a grid of parameters to optimize. The grid must include a column for each parameter in the desired model, prefixed by a period. For example for the random forest decision tree, this means only one column with the names *.mtry* will be needed. If there are more than one parameter then the

Model	Method Name	Parameters
Support Vector Machines	*svmRadial*	*sigma* (Sigma)
		C (Cost)
Random Forest	*rf*	*mtry*: which is the number of variables randomly sampled as candidates at each split.
Stochastic Gradient Boosting (SGB)	*gbm*	*n.trees* (# Boosting Iterations)
		interaction.depth (Max Tree Depth)
		shrinkage (shrinkage)
		n.minobsinnode (Min. Terminal Node Size)
k-Nearest Neighbours (kNN)	*kknn*	kmax (Max. #Neighbors)
		distance (Distance)
nIter	*LogitBoost*	# Boosting Iterations

Table 6.1:Tuning parameters for classifiers [297].

function *expand.grid()* function could be used which creates data frames from the combinations of all values supplied. Optimizing by adjusting tuning parameters encourages simple models. Simpler models tends to have smaller variance in future predictions, making prediction stable [313].

In the model building process for the classifiers described below, parameters were set using the grid facility and also manually in an attempt to find the optimal model. The results noted down in Appendix N to W were the typical results obtained after running the model a number of times with a set of parameter values that gave optimal results.

6.8.2 Logistic Regression

A parsimonious (avoids overfitting) and interpretable model that has excellent performance is a driving aim of the model building process [314]. Logistic Regression is a long established technique that approximates well to this aim. The interpretability is rooted in premise that the data can be divided using a linear boundary. It is similar to linear regression but with a binomial response variable or Y variable (in the case here '*f*' or '*nf*'). It: "*analyzes the relationship between multiple independent variables and a categorical dependent variable, and estimates the probability of occurrence of an event by fitting data to a logistic curve*" [315]. The X variables are used to build a mathematical equation that predicts the probability that the Y variable takes on a value

of ‘f’ or ‘nf’. Thus: “*logistic regression is used when it is plausible that whether or not the Y variable is ‘f’ or ‘nf’ is like a flip of a coin where the probability of getting ‘head’ depends on the X variables. That is unlike a regular coin the probability of getting’ head is not always 50/50 but rather depends on the values taken by the X variables*” [316]. It is used to obtain odds ratio in the presence of more than one explanatory variable (the features in the document representation schemes. The result: “*is the impact of each variable on the odds ratio of the observed event of interest. The main advantage is to avoid confounding effects by analysing the association of all variables together*” [316]. The odds ratio represents the odds that an outcome (‘*f*’ or ‘*nf*’) will occur given a particular exposure (eg Coh-Metrix indices) compared to the odds of the outcome occurring in the absence of that exposure.

These odds ratio are calculated using a logistic or sigmoid function. This logistic function operates through:-

$$f(z) = \frac{1}{1 + e^{-z}}$$

Eq 6.3

where:

$$z = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \beta_3 x_3 + \ldots + \beta_k x_k$$

Eq 6.4

β_0 is the intercept and $\beta_1 + \beta_2 + \beta_3 + \ldots + \beta_k$ are the coefficients of $x_1 + x_2 + x_3 + \ldots + x_k$ (the Coh-Metrix indices for example) respectively. The value of the z measures the total contribution of all the predictor variables used in the model. Logistic curve is an S-shaped or sigmoid curve (shown in Figure 6.13).

Typically using the sigmoid function when the probability of y is greater than 0.5 the model predict fraud else predict non-fraud. Logistic regression has been well used in financial fraud detection, as shown in chapter 2, Appendix A, Figure A.2 and A.3. In particular it has been used in the insurance and credit card fraud [93] with good resultant classification accuracies. In financial statement fraud detection using non-textual features it has been used a number of times [35, 174] and has performed well. Perols [180] compared the performance of 6 machine learning models in detecting financial statement fraud. The results showed that logistic regression and support vector machines perform well relative to an artificial neural network in detection and identification of financial statement fraud. From Appendix A, Table A.3, it was found

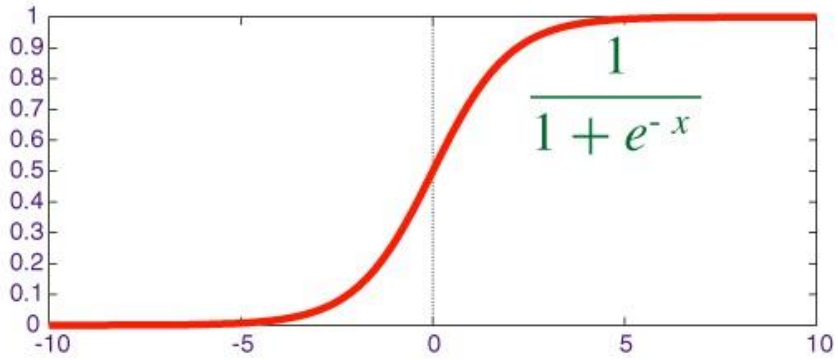


Figure 6.13: The logistic regression function [317].

that only Humpherys [103] had employed logistic regression on two models they had set up attaining a 58 to 63 percent classification accuracy. From the literature review it seems that this work is the first to apply logistic regression in a wide ranging manner on a number of feature sets extracted (the document representation) schemes.

Logistic Regression (LR) was executed over the 12 categories of document representation schemes for both a peer set and matched pair set up. The data was split into training and testing and parameters set to initiate a 10-fold cross validation methodology. Appendices N to W shows the results along with all the other algorithms. Apart from keywords and unigrams, bigrams obtained the highest sensitivity score of 60 per cent. Similarly for the matched pair set up the highest scores were attained by bigrams with a balanced accuracy of 80 per cent and kappa of 0.6. LR is given to high bias due to the simpler nature of the S curve (the function shown in Figure 6.13). The caret version of LR used allows for boosting using the *niter* parameter to boost performance and to deal with the high bias issue.

An oft reported issue with LR is multi-collinerarity. This is where there are high correlations among predictor variables, leading to unreliable and unstable estimates of regression [317]. However to a degree in this framework this was mitigated through the use of 3 distinct feature selection routines which should reduce dimensionality and include only value add features.

6.8.3 Random Forest

Random Forest are a combination decision tree-based machine learning algorithm [318]. As described by Breiman [322] a Decision Tree (DT) is a tree structure, where

each node represents a test on an attribute and each branch represents an outcome of the test. In this way, the tree attempts to divide observations into mutually exclusive subgroups. The goodness of a split is based on the selection of the attribute that best separates the sample. The sample is successively divided into subsets, until either no further splitting can produce statistically significant differences or the subgroups are too small to undergo similar meaningful division. At each branch status probabilities are marked. Expectancy values of each plan is calculated and marked on the corresponding status node of that plan. The branches trimmed and expectancy values compared to identify the best plan. There are several proposed splitting algorithms. The successive division of the sample may produce a large tree. Some of the tree's branches may reflect anomalies in the training set, like false values or outliers. For that reason tree pruning is required. Tree pruning involves the removal of splitting nodes in a way that does not significantly affect the model's accuracy rate [95, 319, 322]. A decision tree concept diagram is depicted in Figure 6.14 (on the left) and shows the branches and decision points (leaf nodes) of the process described above. Figure 6.14 also shows the mechanism of tree building process applied to the Coh-Metrix indices, Figure 6.14 (on the right).

Random Forests grows many classification trees as described above and depicted in Figure 6.14. They are also referred to as an ensemble method. Ensembles are a divide-and-conquer approach used to improve performance. The main principle behind ensemble methods is that a group of "*weak learners*" can come together to form a "*strong learner*" [320]. The random forest in Figure 6.15 using this notion combines trees to form an ensemble. Thus, in ensemble terms, the trees are weak learners and the random forest is a strong learner. Each tree gives a classification, or 'votes' for a class. The forest chooses the classification having the most votes (over all the trees in the forest).

Each tree is grown as follows [321, 322]:

- If the number of cases in the training set is N, sample N cases at random - but with replacement, from the original data. This sample will be the training set for growing the tree.
- If there are M input variables, a number m<M is specified such that at each node, m variables are selected at random out of the M and the best split on these m is used to split the node. The value of m is held constant during the forest growing.

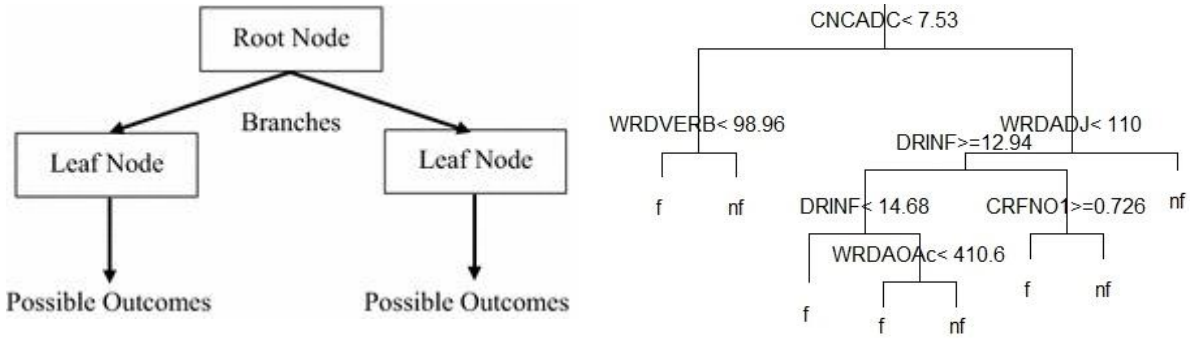


Figure 6.14: The tree building process.

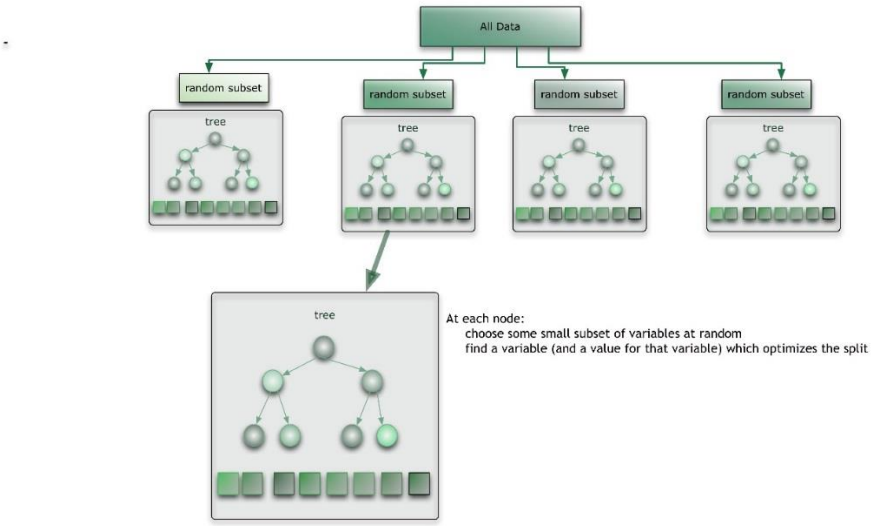


Figure 6.15: Random forest generation [320].

- Each tree is grown to the largest extent possible. There is no pruning.

When the training set for the current tree is drawn by sampling with replacement, about one-third of the cases are left out of the sample. This oob (out-of-bag) data is used to get a running unbiased estimate of the classification error as trees are added to the forest. It is also used to get estimates of variable importance [321, 322]

Advantages often cited for random forest include better generalization capability, robustness, feature pruning ability and simplicity. Breiman [322] cites the following main advantages:-

- It generates an internal unbiased estimate of the generalization error as the forest building progresses.

- It has methods for balancing error in class population unbalanced data sets.
- Prototypes are computed that give information about the relation between the variables and the classification.

Fernandez-Delgado et al. [323] in a wide ranging study based executed a wide family of classifiers on multiple data sets, found that: "*The classifiers most likely to be the bests are the random forest (RF) versions, the best of which (implemented in R and accessed via caret) achieves 94.1% of the maximum accuracy overcoming 90% in the 84.3% of the data sets*" [323].

However random forest are thought to be low on comprehensibility given the number of trees generated. They may also overfit in smaller data sets and are often slower than other models when applying test set data to attain a classification [324].

West et al. [93] conducted a survey on financial fraud and showed that decision tree classifiers in the financial fraud domain are used in credit card fraud detection and FSF. For FSF they mention Humpherys et al. [103], Kirkos et al. [325] Bose and Wang [326]. Classification accuracies in using decision tree based classifiers were 67%, 73% and 72% respectively. Sensitivity averaged around 70%. Appendix A, Table A.2 and A.3 shows that decision tress have been primarily used over non-textual data for classification. Using linguistic data it can be seen that only Humpherys [103] and Dong et al. [163] have used decision trees. This study is therefore amongst the very select few that have attempted to use this classifier-random forest for FSF selection using textual data.

In this study random forest attained the best results in the peer set for LIWC and custom word lists. For the balanced data sets, it attained best results for the bigrams, trigrams, Coh-Metrix, LBCs and topics document representation schemes. From these results it confirms its position as a state of the art classifier.

The code used to run the classifier in R is approximately the same for all classifiers and is shown in Appendix Y, Figure Y.1. The *mtry* parameter available for tuning was set between 10 and 15. This parameter sets the number of weak learners to generate. The optimal results were tabulated (in Appendix N to W).

6.8.4 Support Vector Machine (SVM)

Support Vector Machine are widely used in classification and prediction tasks with consistently good generalization performance [278, 288] . SVM learning has been among the best "*off-the shell*" supervised learning algorithms [278].

SVM produces a binary classifier with the distinctive hyperplane through a non-linear mapping of the input vectors into a high dimensional feature space. In the matrices shown in chapter 4, each row in a matrix constitutes a p-dimensional vector of features representative of a firms report with an associated class label 'f' or 'nf'. A select number 'n' of such examples from a matrix constitutes the training set. The training examples that are closest to the maximum margin hyperplane are known as support vectors. All other training examples are irrelevant with regard to defining the binary class boundaries. Good separation is achieved by the hyperplane with the greatest distance to the nearest training data point of any class, as in general the larger the margin, the lower the generalization error of the classifier [327]. This description of SVMs is depicted in Figure 6.16.

The training examples that are above or below the separating hyperplane or decision function, with the associated class label will be '*nf*' or '*f*' are shown in Figure 6.16. This produces a simple classification process. A test observation is assigned to a class depending upon which side of the hyperplane it is located on. However, the most optimal hyperplane needs to be determined. This would be one with the maximum margin in other words one that is furthest away from any training examples and is thus 'optimal'. SVM typically uses quadratic programming construction of optimal hyperplane to classify the data points into respective classes.

Specifically, from the description given by Yeh [328] to find the hyper plane: $H: y = w.x + b = 0$ and two hyper planes parallel to it and with equal distances to it, $H_1: y = w.x + b = +1$ and $H_2: \mathrm{y} = \mathrm{w.x} + \mathrm{b} = -1$ with the condition that there are no data points between H_1 and H_2, and the distance or margin M between H_1 and H_2 is maximized (depicted in Figure 6.16). The distance between H_1 to H is $\frac{|wx+b|}{||x||} = \frac{1}{\|w\|}$ and thus between H_1 and H_2 is $\frac{2}{||w||}$. Therefore to maximize the margin, we need to minimize $|w| = w^T w$ = with the condition that no data points between H_1 and H_2 satisfy: $w.x + b = >= +1$ for positive examples $y_i = +1$ and $w.x + b = <= -1$ for

negative examples y_i=-1. The two conditions can be combined into $y_i(w.x + b) \geq 1$. So, our problem can be formulated as $\min(w,b)\frac{1}{2}w^T w$ subject to $y_i(w.x + b) \geq 1$ for I =1 1…N. This can then be formulated and solved as a quadratic optimization problem. If no decision function is capable of linearly separating the data a kernel transformation function can be used to map the data into a different dimensional space so that it can be linearly separated using standard SVM decision function techniques [278]. An example of this process shown in Figure 6.17.

Appendix Y, Figure Y.1 again shows the caret code used to execute SVM over the matrices shown in chapter 4. A tuneGrid is used to adjust the tuning parameters sigma (Sigma) and C (Cost). The latter has to be tuned to better fit the hyperplane to the data. It is responsible for the linearity degree of the hyperplane (it is not present when using linear kernels). The smaller sigma, the more the hyperplane is going to look like a straight line. If sigma is too great, the hyperplane will be more curved and might separate the data too well and lead to overfitting. The C parameter is responsible for the 'soft margin' of SVM. The soft margin is the area around the hyperplane (distance between H2 and H1) in Figure 6.16.

In the FSF domain, Appendix A, Table A.2 and Table A.3 shows that it has been used considerably with ratios as features. These results are in unison with a survey paper done by West et al. [93]. From Table A.3 it can be seen that it is one of the most popular classification techniques used by researchers working with linguistic data as features to detect FSF. Classification accuracy is typically above 70%.

6.8.5 Stochastic Gradient Boosting (SGB)

Gradient Boosting entails '*boosting*' many weak predictive models into a strong one, in the form of an ensemble of weak models. This is achieved by applying the function of the model repeatedly in a series and combining the output of each function with weighting so that the total error of the prediction is minimized [276].

SGB derives from the methods of applying boosting to decision trees. The strategy is to compute a sequence of relatively simple trees, where each successive tree is built from the prediction residuals of the preceding tree. For example, if the complexity of trees is limited to only three nodes: a root node and two child nodes - a single split.

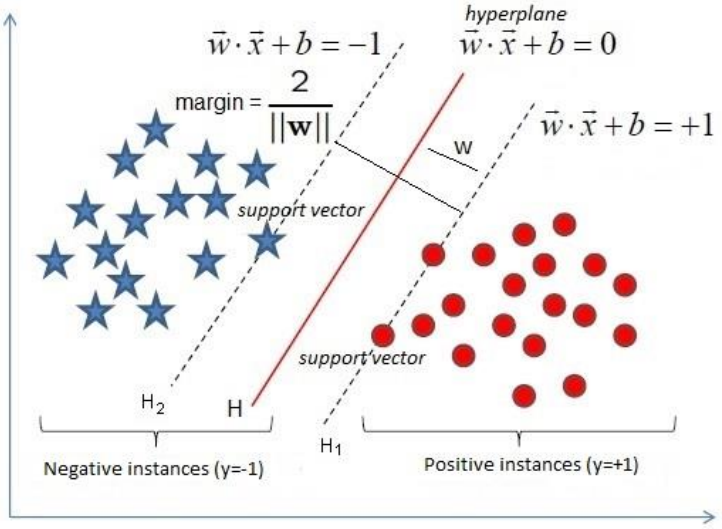


Figure 6.16: Support vector machines [328].

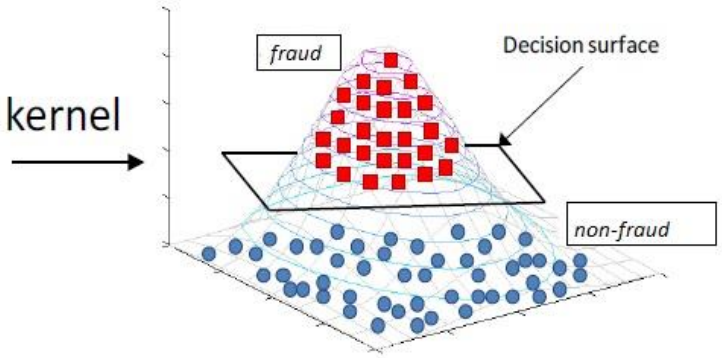


Figure 6.17: Support vector machines with kernel [328].

At each step of SGB, a simple (best) partitioning of the data is determined, and the deviations of the observed values from the respective means (residuals for each partition) are computed. The next three-node tree will then be fit to those residuals to find another partition that will further reduce the residual variance for the data given the preceding sequence of trees. Each tree developed during the process is summed, and each observation is classified according to the most common classification among the trees. The combined effect is to reduce SGB's sensitivity to inaccurate training data, outliers, and unbalanced datasets. This process is shown in Figure 6.18, as the errors are amalgamated and built into the training model the residual errors are reduced as the process continues until finally the resultant function maps the data more accurately [329].

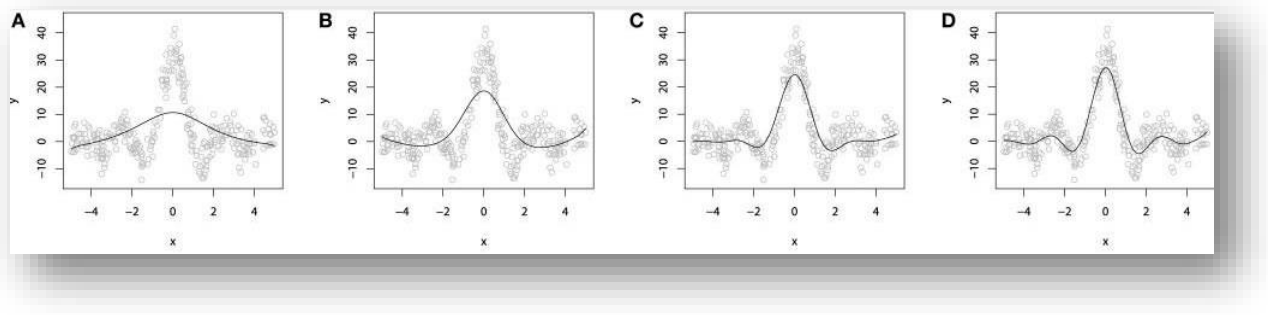


Figure 6.18: Stochastic gradient boosting [329].

The typical ensemble techniques like random forests rely on simple averaging of models in the ensemble. The family of boosting methods is based on a different, constructive strategy of ensemble formation. The main idea as shown of boosting is to add new models to the ensemble sequentially. At each particular iteration, a new weak, base-learner model is trained with respect to the error of the whole ensemble learnt so far [329]. Classification accuracy has been pushed upward with the use of SGB in a number of reported cases [330, 331].

The caret code used to execute SGB is similar to the previous models and is shown in Appendix A. The tuning parameters, the *n.trees* parameter was set between 150 to 200. This is the total number of trees to fit. Increasing n reduces the error on the training set but setting it too high may lead to over-fitting [332]. The *interaction.depth* was set between 5 and 10. This is the maximum depth of variable interactions in other words number of splits that it has to perform on a tree (starting from a single node). The *minobsinnode* minimum number of observations in the trees terminal nodes was also set between 5 and 10. The *shrinkage* parameter is applied to each tree in the expansion was set at 0.1 as recommended by Ridgeway [332]. It is used for reducing, or shrinking, the impact of each additional fitted base-learner (tree). It reduces the size of incremental steps and thus penalizes the importance of each consecutive iteration [332].

There are very few studies in financial fraud detection using intelligent techniques that deploy SGB. It is unclear if any boosting of any type have been applied to the base learners. West et al. [93] indicate use of decision trees in financial fraud, as pointed out in section 6.8.3. Their study shows that decision trees are routinely used. This can also be confirmed from Tables A.2 and A.3 in Appendix A. The results obtained are

competitive with respect to other models. However variants of decision trees like SGB was only used by Whiting et al. [183] (see Appendix A).

From Appendix Y it can be seen that SGB has demonstrated good predictive ability using the document representation schemes of chapter 4. It has been the top performer based on a peer set data setup using bigrams, trigrams, LBC's and Topics. For the matched pair design it has outperformed other models when using features for LIWC, concepts and LSA chosen concepts.

SGB is a powerful technique against overfitting since each consecutive tree is built for a different sample of observations, and yield models (additive weighted expansions of simple trees) that generalize well to new observations, i.e exhibit good predictive validity. However, the overfitting can only be overcome with the correct tuning of the parameters shown. Given that there are 4 parameters to choose this can be a trying balancing act. Therefore, it makes it hard to get a good fit to the data. The boosting process may also render models that are low in interpretability.

6.8.6 k Nearest Neighbour (kNN)

The kNN technique is widely used as a classifier because of its simplicity and high efficiency [333]. It is also known as a '*lazy*' classifier as no work is done to train the model. All that is needed are input points with their labels. Therefore, given a test document *x*, the goal is to find the *k* nearest neighbours of x among all the training documents. The test document would be classified in the most popular class among its *k* nearest neighbours. '*k*' is chosen before any assignments take place. A visual of this process is shown in Figure 6.19. If *k* = 3 the document would be assigned to the red class, if *k* = 5 it would be assigned to the green class. Assignment are based on the contiguity hypothesis: "*a test document d would have the same label as the training documents located in the local region surrounding d*" [49]. Distance is computed using either Cosine similarity or Euclidean measures.

According to Jiang et al. [334] if several of the *k* nearest neighbor documents belong to the same category, then the sum of the score of that category is the similarity score of the category in regard to the test document *x*. By sorting the scores of the candidate categories, the system assigns the candidate category with the highest score to the test document *x*. The authors assert that the decision rule of kNN can be written as:

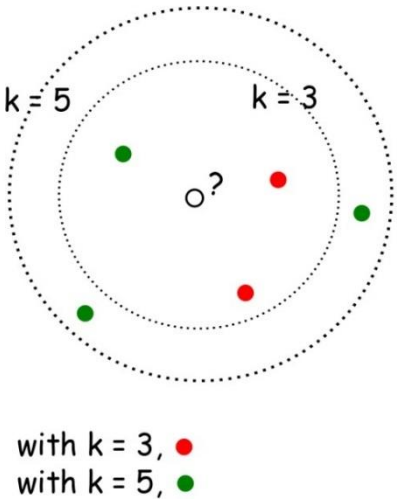


Figure 6.19: The kNN operation [333].

$$f(x) = \arg\max Score\ (x, C_j) = \sum_{d_j \in kNN} sim(x, d_i) y\left(d_i, C_j\right),$$

Eq 6.5

$f(x)$ is the label to the test document x.

$Score\ (x, C_j)$ is the score of the candidate category C_j with respect to x.
$sim(x, d_i)$ is the similarity between x and the training document d_i.
$y\left(d_i, C_j\right) \in \{0,1\}$ is the binary category value of the training document d_i with respect to C_j (y=1 indicates document d_i is part of category C_j or y=0).

For the matrices shown in chapter 4, 75% of the documents (vectors) would be mapped out in Euclidean space. The remaining 25 % would be used to be tested out as outlined above. The caret code used is the same as used for previous classifiers. The *kmax* the number of neighbours is varied between 3 and 5. The distance used is based on Euclidean distance.

From the recent survey done by West et al. [93] and the results of literature search shown in Appendix A, it is clear that kNN has been used sparingly for financial fraud detection. This is likely due to the fact that its performance is degraded when dealing with high dimensional data and unbalanced data sets. In the former case, nearest neighbour might be some distance away and determining a nearest neighbour can become problematic and in the latter case it would be harder to harness discriminatory power based on the kNN algorithm for unbalanced sets as most test cases would likely

be classified into the majority class. Pre-processing through feature selection is necessary when using this algorithm as otherwise irrelevant attributes would render the results less meaningful [335, 336]. From the results shown in Appendix N to X it can be observed that kNN is one of the poorest performing classifiers, especially in the peer set set-up. This is in line with expectations as the data despite feature selection is still high dimensional which degrades the classifier accuracy. This classifier was included in the model building exercise to highlight that the performance of the models are in line with expectations. Not all perform well. Some do better than others.

6.9 Clustering – k-means

Clustering defines a group of data exploration techniques that seeks to unearth the 'natural' grouping of multidimensional observations based on their degree of similarity or distance [337]. As can be deduced clustering is distinct from classification. As the latter pertains to a known number of classes and the objective is to assign new observations to one of these groups. In the latter case nothing is known on the number and nature of the groupings. The objective of clustering is to discover the natural groupings of the observations such that the observations in a given cluster tend to be similar in some sense to other observations in the same cluster and dissimilar to observations in other clusters [337]. For example, for the corpus under study groupings are sought so that the fraud documents are clustered into a group and non-frauds into another. The k-means algorithm is deployed to attain this separation in the data. This is known as partitional clustering as opposed to hierarchical clustering. The latter is a division of the set of data objects into non-overlapping clusters so that each data object is in exactly one subset. In the former the clusters have sub-clusters organised into trees. The k-means algorithm as described by Ahmed et al. [285] is described below:-

1. Select K points as initial centroids.
2. Repeat
3. Form K clusters by assigning each point to its closest centroid.
4. Recompute the centroid of each cluster.
5. Until Centroids do not change.

Often a mean value is used to initialise a centroid. After points are assigned to a centroid the centroid is then updated. To assign a point to the closest centroid, a proximity measure (Euclidean distance) is typically used. The algorithm repeatedly calculates the similarity of each point to each centroid.

The end goal for the fraud/non-fraud reports: "*is to minimize the average squared Euclidean distance of documents from their cluster centers where a cluster center is defined as the mean or centroid*" [49]. How well the: "*centroids represent the members of their cluster is the residual sum of squares (RSS) - the squared distance of each vector from its centroid summed over all vectors*" [49]. So k (chosen) clusters are assigned centroid coordinates: "*the algorithm then moves the centroids around in space in order to minimize distance of documents to their nearest center*" [49]. This movement in the centroids results in variability in results every time the algorithm is run. In k-means the ideal clusters is a: "*sphere with the centroid as its center of gravity*" [49]. If the data does not have this spherical structure then clustering accuracy drops considerably.

Appendix Z, Tables Z.1 to Z.22 show the results of applying the k-means algorithm to the matrices outlined in chapter 4. Only the peer set data set up was considered as the intention is to show how well the fraud and non-fraud reports are separated using the k-means algorithm. Its performance on peer set data set up is adequate for this purpose. All matrices that resulted from feature selection using the 3 feature selection routines was passed to k-means.

Intra-cluster similarity is shown by the data3Cluster$withinss values and inter-cluster similarity is shown by the data3Cluster$betweenss values. A good clustering result would be high intra-cluster similarity and low inter-cluster similarity. The table(data3Cluster$cluster) command displays the number of data points allocated to a cluster.

From Appendix Z it can be seen that for all the document representation schemes combined with feature selection that data3Cluster$betweenss has attained a reasonable separation. However the data3Cluster$withinss is spread out so that the separation is not clear and distinct (with the exception of unigrams and keywords). In all cases the Boruta selected keywords have produced better clusters. The table(data3Cluster$cluster, data2$class) command shows the clusters with the labels revealed. The correctly clustered values that can then be deduced. It can be seen that

in some cases as with the n-grams the clustering results tie in approximately with the classification results. However in most other cases classifier performance is better, as shown in Appendix N to X for the peer set data set-up. It is also clear from the results that fraud reports have again been mostly assigned to non-fraud reports due to the unbalanced nature of the data. This, as has been seen, in the classifier results can be mitigated by correcting the imbalance.

Clustering has been used by previous researchers for financial fraud detection. Ahmed et al. [285] perform a state of the art survey in this area. They identified the use of k-means in a handful of occasions. For example, areas examined include refund transactions. Issa and Vasarhelyi [338] using k-means identify anomalies by pinpointing any transactions that are furthest away from the centroid. Thiprungsri and Vasarhelyi [339] used k-means to aid auditors to identify life insurance claims and similarly looked for anomalies. Le Khac et al (2016) used it for examining data relating to money laundering.

Results of detecting of Financial Statement Fraud using clustering over the corpus used in this study was shown in Appendix A, Table A.2 and A.3. It can be seen that Li et al. [178] research is the only recent identified case that use k-means. Ahmed et al. [285] also cite Deng and Mei [340] use k-means combined with self-organising map technique to derive the clusters. These clusters are then measured using a silhouette index to determine potential fraud. Data used were 100 financial statements of listed Chinese companies (half of these statements were fraudulent). From Appendix A, Table A.3 it can be seen that Chen [162], Wang and Wang [172] and Glancy and Yadav [165] used a clustering technique to identify fraud reports.

Sabau [337] also conducted a survey on all clustering techniques as applied in the finance domain to detect fraud. Further Albashrawi [341] conducts a more recent review of detecting financial fraud using data mining techniques.

It can be seen that there is a paucity of research that uses clustering for FSF detection. Zhou and Kapoor [173] point out that fraud has an ability to morph and evolve. Signature based techniques like classifier models may be too rigid to catch new tactics used by fraudsters. Clustering based techniques that make few assumptions perhaps could pick up anomalies better and should be used more extensively. From the results shown by running k-means over the corpus and from the results indicated by previous research it is clear that clustering holds potential to aid in financial fraud detection.

6.10 Discussion on Classifier Performance

Appendix O, Table and Figures O.1 to O.12 show the bigram classification results. For bigrams/pca/peer set LR attained a balanced accuracy of 62% and sensitivity of 40% which compares well against the other classifiers. On the cross validation performed during training the ROC is showing up near 0.62 with sensitivity and specificity the same as was attained in the test set results. As can be seen from the comparative results in the box plots all other classifiers are similar in their performance. The resampling results, overall classification results improve with the more balanced data set in the matched pair data set ups. The results for bigrams/boruta improve for the peer set data with an overall balanced accuracy of 73% and improved sensitivity measure at 52%. Though confidence interval are lower than other intervals for classifiers. The results also show up that SGB and SVM have attained higher sensitivity value. The matched pair results (bigrams/boruta) are better all round for all classifiers. LR for bigrams/IG peer set attained a slightly higher overall classification accuracy with importantly a higher sensitivity score and ROC value of nearly 0.8 but compared to the other classifiers registering lower confidence intervals. Bigrams/IG matched pair all classifiers attained a classification accuracy of about 80% and the resampling results show little difference amongst the classifiers.

Appendix P, Table and Figures P.1 to P.12 show the trigram results. For trigrams/pca/peer set all classifiers attained classification accuracies of around 52%. This poor performance is apparent from the cross validation resampling results shown in the box plots, significantly the sensitivity results are low. PCA/matched pair results are better with LR attaining 62% accuracy with RF, SGB and SVM performing better. Results from Boruta/peer set are better for trigrams, with LR attaining 72% accuracy with a sensitivity score of 48%. It has an ROC score of about 0.7, due mainly to a higher specificity score. As can be seen from the box plots the classifier performance is comparable to the others with SVM doing better. The matched pair results for Boruta are better with LR at 70% classification accuracy, its performance as can be seen from the box plots is comparable to the other classifiers with only SVM performing better. Results have dipped for IG/peer set with LR attaining 60% balanced accuracy with a low sensitivity score of 28%. As can be seen from the box plots this low score is borne out by all the classifiers. Results have improved slightly for IG/matched pair with LR at

62% classification accuracy with ROC value of about 0.8. Performance of LR is again comparable to others with GBM doing better.

Appendix Q, Tables Q.1 to Q.12 show the Coh-Metrix results. For Coh-Metrix/peer set/PCA, LR attained a classification accuracy of 61% with low sensitivity score, only SVM performed better though it had the lowest confidence intervals. Performance on matched pair/PCA was lower and compared unfavourably with the other classifiers. LR performance with Boruta selected features for peer set was improved at 72% classification accuracy with 52% sensitivity. This is comparable to the other classifiers as can be seen from the box plots. Again classifier performance for LR fell to 66% for matched pair/Boruta ranking it lower than the others. For Coh-Metrix IG/peer set had a balanced classifier accuracy of 66% with a sensitivity score of 56%, ROC value of about 0.68. This again is in line with the performance of all the other classifiers. For matched LR stays at the 66% level with an improved sensitivity score and kappa.

Appendix R, Table and Figures R.1 to R.9 show result for LIWC. In the peer set the best results by LR was attained using the Boruta selected variables with a low sensitivity score of 42 %. The results show that for all feature selection routines it had a ROC value in the region 0.6 for the peer set data set up. It can also be seen from the box plots the SVM and SGB performed in comparison LR and had higher confidence values. For the peer set, Information Gain selected features for LIWC resulted in an overall lower classification success primarily resultant from poorer sensitivity score. The matched pair performance of LR on LIWC chosen variables was better all round. PCA chosen variables attained the best results with a classification accuracy of 86%. For all feature selection routines, SVM and SGB performance were above the others. From the boxplots its can be seen that the performance for all other classifiers is comparable.

Appendix V, Figures and Tables V.1 to V.12 show results for custom dictionaries developed for the financial domain. For the peer set, LR attained the highest accuracy with Boruta selected variables at 64 % with sensitivity low at 32 %. It can be seen that for all peer set/all classifiers have produced similar results, the sensitivity score is low, lowering the ROC value. For the matched pair set up LR produced a better sensitivity score but the specificity has dropped. However as can be seen for all feature selection routines and for all classifiers the classification performance has improved with improved ROC values though kappa is still low.

Appendix T, Figures and Tables T.1 to T.12 show results for topic modelling. The best results in this category for peer set was attained by SGB using Boruta chosen features. For PCA generated features again for peer set, SGB attained the highest accuracy with 68% balanced accuracy and 48% sensitivity score. The other classifiers all had similar performance as can be viewed from the box plots, with the lower sensitivity score contributing to lower kappa rating. The high ROC values are due to the higher specificity score. For the peer set/boruta/topics performance again did not improve much as the sensitivity score remained low for all classifiers, reaching a high of 44 % with SGB. The box plot show up the slightly better performance of LR and SGB. For peer set/IG/topics the sensitivity score is lowered further rendering a lowered overall balanced classification accuracy score. For the matched pair/topics scenario the best results was attained by PCA/RF and Boruta/SGB. Overall the results are comparable for all the classifiers with improved sensitivity/specificity scores. The box plots show up the superior performance in terms of ROC values, specificity, and sensitivity for the SGM and RF classifiers in this category for all feature selection routines.

Appendix S, Tables and Figure S.1 to S.12 show results for Linguistic Based Cues. Best results for peer set/LBCs was attained by PCA/SGB at a balanced accuracy score of 70 % and a sensitivity score of 44%. Results for PCA selected features/peer set are comparable with an averaged balanced accuracy of around 65% for all classifiers and an average sensitivity score of around 42%. The ROC values are all above 0.60 due to high specificity values. SVM and kNN performance drops for the Boruta selected feature, with LR, SGB and RF maintaining the same performance shown for PCA selected features. This difference in performance is visible in the box plots. Performance for peer set drops for all classifiers using IG selected features with a significant drop in the sensitivity score.

In the matched pair/LBCs category the best performing classifier was RF using Boruta selected features (Appendix S, Table S.9) with a classification accuracy of 74%. For matched pair/PCA selected features, only kNN performed poorly with the others attaining a classification accuracy of over 60%. This performance difference is reflected in the box plots that show up differences in the ROC, sensitivity, specificity scores. Classifier performance on matched pair/Boruta/LBC remains similar to matched pair/IG/LBCs, however this time LR has a poorer performance than the others. This is again reflected in the box plots that shows up the performance

differences. IG routine returned 0 features from the LBCs matrix as significant so no further analysis was taken.

Appendix U, Tables and Figures U.1 to U.12 show results for concepts. Appendix U show that the best performing classifier in this category was SVM using Boruta selected features at a balanced accuracy of 77% with a high sensitivity score of 60% and a high kappa score of 0.59. This shows that features used are of good discriminatory value. For PCA/concepts/peer set the sensitivity score reaches a high of 64% with LR though p values are high lowering the statistical significance of the result. SGB attained 48% sensitivity with better confidence intervals. The better performance of SVM and SGB in this category can be ascertained from the box plots. ROC values are over 0.6 due to higher specificity values. For the peer set/Boruta/concepts category the results are good for SGB, RF and SVM with sensitivity score exceeding 50%, higher confidence intervals and p-values < 0.05 (indicating results are statistically significant). Both LR and kNN performed poorer on these scores. This can be verified by the box plot that compares performance.

For the peer set/IG/concepts category the best performing classifiers were RF, SGB, LR with RF and SGB reaching a sensitivity score of 60% with p values were less than 0.05. The resampling results bear out the better performance of these classifiers, though as can be seen from the box plots there is some variability on the sensitivity results.

Highest results for concepts on the matched pair category was attained by SGB at a classification accuracy of 78%, sensitivity 80% and specificity 76%. Matched pair/concepts with all feature selection routines performed well apart from kNN/Boruta (sensitivity 57%), kNN/IG (kappa 0.3), kNN/PCA (kappa, 0.08, sensitivity 48%) and SVM/IG (kappa 0.3) all other classifiers attained a score around the 0.40-0.50 interval with classification accuracy, sensitivity and specificity values all exceeding 60%.

For the remaining document classification schemes (unigrams, keywords, keywords-Rutherford and LSA keywords in Appendix N, O, P and W (all tables and figures displayed) for both the peer set and matched pair all the classifiers produce good performance. For example for keywords – Rutherford all performance measures shown in Appendix W (for example kappa, sensitivity, specificity result in a score 1,1,1 respectively) are showing that the features chosen are able to discriminate between the two classes of reports. However it should be borne in mind that in such cases, as

depicted in Figure 6.9 a certain amount of overfitting may be taking place. The models could be showing high variance and low bias thus overfitting the truth target. More data to perform further testing may indicate that this is the case. The results obtained using these features did not shift much, despite tuning and shifts in proportions between training/testing data and tuning parameters specific to the classifiers.

6.11 Discussion and Conclusion

For each document representation scheme/data set-up/feature selection routine, the variable importance is computed based on the corresponding reduction of predictive accuracy when the predictor of interest is removed.

For unigrams, as shown in Appendix N similar variables have aided in the model building for all the classifiers. Predominantly, the following lemmas occur: *'addit', 'requir', 'expens', 'primarili', 'achiev', 'increas', 'share', 'expect', 'signific', 'financi', 'futur', 'capit', 'growth', 'loss', 'may'*. This suggest that there is difference in the use of connectives between fraud and non-fraud firms. A preoccupation with performance is also apparent through the use of terms such as '*capit*', 'growth', 'loss', '*increas*', *'share', 'expect'.*

For bigrams as shown in Appendix O, the following analysis could be made:-

For PCA/peer set: *'ability to', 'be able', 'may be', 'our revenues', 'we believe', 'subject to', 'unable to', 'to obtain'*, *'and could'* dominate for all classifiers. Additionally for the PCA/matched pair scenario '*interest rate*' is also prominent. Boruta/peer set selected features attained higher classification accuracies with SGB reaching a high of 85% with 72% sensitivity. Bigrams that dominate include: *'acquisition of', 'the acquisition', 'purchase price', 'borrowings under', 'of approximately', 'year ended', 'to obtain', 'accounted for', 'necessary to', 'continued to', 'the fiscal', 'state of', 'volume of'*. For matched pair additionally: '*primarily due', 'operating income'* are also prominent. For IG/peer set, *'acquisition of', 'purchase price', 'to date', 'accounting and', 'customers with', 'event that', 'year ended', 'of approximately', 'between the', 'necessary to', 'continued to', 'primarliy due'* have aided in the discrimination task. For the IG/matched pair set up bigrams *'purchase price', 'failure to', 'in compared', 'contributed to', 'may be'* are also prominent. Overall, bigrams that contain the term *'acquisition', 'may', 'purchase' , 'ability', 'obtain', 'borrowings', 'approximately', 'primarily', 'able',* dominate.

This again suggests that concerns in financial performance in particular liquidity concerns.

Trigrams have generally performed poorer with low balanced classification results for all peer set data set up. SGB with Boruta chosen variable produced the highest balanced classification accuracy at 78% and a sensitivity score of 60%. The trigrams deigned important by SGB were: *'the results of', 'the impact of', 'the acquisition of', 'provided by financing', 'at the time', 'use of the', 'in the event', 'may be required', 'for the year', 'million in cash', 'primarily due to', 'and sale of'*. Random forest using IG features attained the highest classification accuracy with trigrams such as *'based on the', 'the end of', 'primarily due to', 'in the event', 'one or more', 'entered into a', 'the event of', 'in compared to', 'comply with the', 'of shares of', 'ability to provide',* deigned important in building the training model. Again the continued theme of liquidity, borrowing, performance were able to discriminate a fraud from a non fraud firm.

For Coh-Metrix the best overall performance in the peer set was attained by Boruta/SVM (Appendix Q). It ranked: *'SMCAUSwn', 'WRDFRQa', 'WRDAOAc', 'SYNSTRUTt', 'PCCNCz', 'DRPVAL', 'SYNSTRUTa', 'CNCTempx', 'WRDIMGc', 'WRDADJ', 'WRDMEAc', 'PCCONNz'* as significant discriminators. The importance of these indices are repeated in differing ranking of importance for all the classifiers using Boruta. For matched pair the best performance was attained by Boruta/RF. It ranked: *'SMCAUSwn', 'WRDFRQa', 'WRDAOAc', 'SYNSTRUTt', 'PCCNCz', 'DRPVAL', 'SYNSTRUTa', 'CNCTempx', 'WRDIMGc', 'WRDADJ', 'WRDMEAc', 'PCCONNz',* as significant. Meaning of Coh-Metrix indices are outlined in Appendix B. Tables B.2 to B.12.

Coh-Metrix indices (prefixed by 'DR' and 'SY') indicate that there is a difference in the syntactic structure between fraud and non-fraud firms. *'WRDFRQa', 'WRDIMGc', 'WRDMEAc'* which relate to concreteness and meaningfulness of words are similarly indicative of text that may be less understood by a reader. *'SYNSTRUTa', 'SYNSTRUTt'* also point to less clarity in text. This correlates with the view that deception in text is manifested by dense syntactic structure to reduce readability and comprehension.

There is also a difference in the use of adverbs and adjectives and as mentioned, this can qualify the meaning of statements. Further, there is a difference in the use of connectives which can again lead to poor cohesion if used sparingly. Referential

cohesion measures (prefixed by 'CR') are also showing up as discriminators. This could again be the case that fraud firms are attempting to obfuscate the narratives through poor co-referencing

LIWC variables were extracted using the latest version of the software (2015) which had updated dictionaries to section words in text into categories shown in Appendix H, Table H.2 and Figure H.1. In the peer set the best performance was attained by RF using Boruta. It deigned variables such as: *'adj' (adjectives), 'cogproc' (cognitive processes), 'relative' (relativity – motion, space, time), 'Authentic function'* (summary variable as described by Newman, 2003), 'article' (such as a, an, the),*' focusfutur*e' (words such as may, will, soon), *'focuspresent'* (words such today, is, now), *'interrog'* (words such as how, when, what), *'verb'* (common verbs), *'compare'* (comparisons such as greater, best, after), *'auxverb'* (auxillary verbs such as am, will, have). Full details of LIWC categories and variables are shown in Appendix H, Table H.2. Such results accord with deception research outlined in chapter 2 where linguistic ploys using adjectives and increased cognitive processes, manipulation/omission of details such as time are part of the linguistic arsenal used by liars to create distance from actions/events, to deflect blame. The matched pair results for LIWC were good all round, suggesting that further tuning/management of the imbalance in the peer set if managed differently could improve the classification results. The highest results was attained by SGB using PCA features (see Appendix R). It ranked: *'ipron*' (personal pronouns), '*Tone*' (emotional tone), '*Authentic', 'Analytic'*, (summary variables) '*function'* (function words such as it, to, no, very) , '*negate*' (words such as no, not, never), '*adj'* (adjectives), '*WPS'* (words per sentence), '*focuspresent'* (words such today, is, now), *'compare'* (comparisons such as greater, best, after) as significant. Again such findings overlap with finding from deception research as expounded in chapter 2. The results for matched pair/LIWC are similar for kNN, LR, RF with SGB and SVM performing better. This is visible from the boxplots, confidence levels shown after classification results on test set for each feature selection routine.

The best performing classifier for the peer set/custom dictionaries was RF using boruta selected features. Words in word list of '*positivity_Freq', 'negativity_Freq'*, '*Uncert1_Freq'* (Loughran and Macdonald's positive, negative and uncertainty word list). It can been seen that overall for the results for all classifier is poor as the sensitivity score is low. This improves in a matched pair set up but the kappa values

have not picked up. This suggest that the custom dictionaries have poor discrimination capability in the classification task.

Document representation via topics garnered via LDA to perform classification using weights attached to the topics did not perform strongly, especially in the peer set data set up. Given the research work in the area of fraud detection from linguistic analysis that concentrates on '*how*' disclosure is made, this was a new attempt to identify '*what*' was being disclosed in annual reports/10-K. However the results indicate that perhaps the weights given by LDA (mallet) attached to the topics used to perform the classification are not strong enough to aid in classification, in other words they have poor predictive power. An approached that further probes the text for '*what*' is being said may prove to be more discriminatory. However, for all the classifiers used in both the peer set and matched pair set up '*Topic 24*' dominates as a feature prominent in aiding classification. Appendix T shows topic 24 to contain contains words/phrases: '*december', 'year ended', 'operations', 'approximately', 'million', 'cash', 'business', 'credit', 'capital', 'interest', 'agreement', 'increase', 'due', 'facility', 'company', 'rate', 'state', 'acquisition'.* Again terms relating to liquidity, monetary concerns, performance seep through as being of discriminatory value.

Results from LBC confirm previous work conducted by Humpherys [103] for the matched pair setup. However this work takes a step further through the construction of a peer set data set-up, which is more reflective of a real world situation. There are more non-fraud than fraud firms and a true test for a good predictive model is to pick out the fraud firm from their narrative content. In this case the sensitivity score gets to a high of 44% (Appendix S). This clearly needs to improve, as otherwise fraud firms will not be identified in a real world situation. Other classifiers/ feature selection routines could be attempted to determine if the sensitivity score improves. However looking through the results the LBC that have aided in the discrimination task are: '*Temporal.Imm.Ratio', 'Modal.Verb.Ratio', 'Content.Word.Diversity', 'Avg.Word.Length', 'Avg.Sent.Length', 'Imagery'.* This all accords with linguistic cues of deception research expanded in chapter 2. Invariably liars manipulate text to create distancing effects and introduce obfuscation to detract from the truth and deflect blame.

Cecchini et al. [161] also attempted to bunch tokens used in narrative sections of fraud/non-fraud firms into concepts through the use of WordNet. Their total corpus

size was 122 reports (61 fraud and 61 non-fraud). They attained a classification accuracy of 55 to 72%. This study takes a step using a larger corpus size and setting up of both a balanced and unbalanced data setup. It also shows in a much wider sense how 5 classification algorithms using the two type of data set up perform. It further shows in a greater light the concepts that are aiding the classification task. The best classifier output as significant: *'employee.noun', 'performed.verb', 'ended.verb', 'acquisition.noun', 'obtained.verb', 'relates.verb', 'acquired.verb', 'put.verb', 'event.noun', 'continued.verb', 'improve.verb', 'companies.noun', 'purchase.noun', 'accounting.noun', 'ability.noun', 'provision.noun', 'payments.noun', 'changes.noun', 'regulations.noun', 'restrictions.noun', 'rate.noun', 'based.verb', 'required.verb', 'credit.noun', 'fail.verb', 'compete.verb'.* There is an emphasis on words that relate to, '*acquisition*', '*fail*', '*obtain*', '*require*'. The results of peer set/concepts substantiate previous findings that emphasise the liquidity/monetary concerns that are proving to aid in the classification task. For matched pair/concepts the concepts that were deemed to aid in classification were: *'acquisition.noun', 'standards.noun', 'improve.verb', 'results.noun', 'acquired.verb', 'obtaining.verb', 'division.noun', 'discounts.noun', 'obtained.verb', 'improving.verb', 'association.noun', 'payment.noun', 'compete.verb', 'required.verb', 'compensation.noun', 'agreements.noun', 'action.noun', 'ability.noun', 'continued.verb', 'entered.verb', 'consisting.verb', 'ended.verb', 'improve.verb'*. Again themes around acquisition, obtain, require strong along with improve, results, consisting, payment are featuring strong.

The remaining document representation schemes (keywords, keywords-Rutherford, and LSA-concepts) the classification accuracies were high to near perfect primarily due to the strong underlying patterns produced by the features in separating the fraud from the non fraud reports. Strong features for keywords emerged as: *'capital', 'result', 'management', 'stockholder', 'tax', 'certain', 'net', 'accounting', 'rate', 'results', 'losses', 'primarily', 'capital', 'expected', 'stockholder', 'acquisition', 'compared'.* Strong features for Rutherford-keywords emerged as: *'significant', 'years', 'financial', 'company', 'capital', 'interest', 'risk', 'loss', 'rate'*. For words in LSA-concepts: *'result', 'operating', 'may', 'certain', 'management', 'operations', 'costs', 'net', 'cash', 'required'* emerged as significant. Again themes around capital, stockholder, acquisition, risk, loss all tie in with previous findings that indicate differences in performance and monetary concerns.

Overall this chapter has outlined mechanics of the machine learning process at a high level of abstraction. As can be appreciated the task under study: "*Can linguistic features separate narratives of fraud firms from non-fraud firms*?" From chapter 4 and 5 the feature extraction and feature selection was undertaken as depicted in the framework. For each firm's report in the corpus a vector was produced with a class label 'f' or 'nf'. This from the description given in the chapter is a natural classification task ideally suited for machine learning based modelling. In a two data set-up (matched pair and peer set) approach these vectors, combined into matrices were then fed to the classifiers described in this chapter. The pitfalls in machine learning models such as high dimensionality was managed through feature selection. This was thoroughly investigated using three distinct feature selection routines, the output of each was passed to the classifiers for model building. Overfitting was managed through use of a distinct test set. The parameters were also tuned (as given in Table 6.1) to ensure results were robust. Using the Caret package further streamline's and standardises the process of model building ensuring transparency and giving greater credibility to the results [296]. Given that the performance of classifiers would be adversely affected by an unbalanced data set, a balanced data set was also used to gauge how classifier performance varied. Five classifiers were chosen. SVM was chosen because as can be determined from the description in 6.8.4 that it is a natural binary classifier. Its overall performance as can be viewed from the results in generally high. However extensive tuning needs to be conducted to ensure that the decision boundary and the separating margin are optimal. If not done thoroughly classifier performance can show up as poor. Random Forest and Stochastic Gradient Boosting through their boosting mechanism have reduced classification error and have consistently performed well in the classification task under study (see Appendix X). For SGB it is necessary again to fine-tune the parameters to attain optimal performance. Logistic Regression was included as it is a simple model built on an shaped function. This results in high bias but ensures that no overfitting occurs. As can be determined from the results it generally does not fit the data well, despite the boosting used with the classifier. Similarly, kNN performs poorly. This is likely to be the result of the data that is still more high dimensional that can be managed with the proximity measures used in kNN. Performance of both LR and kNN follow expected

performance. They were included to reinforce the validity of the data and the results in general.

Chapter Seven

CONCLUSION

"If falsehood, like truth, had only one face, we would be in a better shape. For we would take as certain the opposite of what the liar said. But the reverse of truth has a hundred thousand shapes and a limitless field."

Montaigne c.1572

7.1 Summary and Contributions reviewed

From the literature review it was determined that using a corpus was a valid approach to the study of language. The theoretical underpinnings that give weight to this assertion were reviewed. Further predisposing factors that lead to FSF were examined and the scale of the problem reviewed. It was shown that as compared to research using computational techniques that investigates FSF detection with quantitative features, FSF detection using linguistic features was of smaller scale. This study makes an attempt to rectify this deficiency through examining an extant range of linguistic features that could aid in uncovering FSF.

As far as can be determined this is the first study that utilizes the corpus linguistics methodology over a mixture of 10-K/annual reports, 25% of which were produced by firms formally indicted for FSF. The results show clearly that certain keywords, outlined in chapter 3 provide strong discriminatory ability in separating the two types of documents. This was verified when these features were used as input to the classification process. Further keywords taken from previous studies in financial narratives were examined and found to have excellent discriminatory ability in the classification task. A new framework was introduced that captures the process of linguistic feature extraction, selection and classification for deception detection in financial text. This framework is new to the domain under study.

Chapter 4 details a large portion of the significant contributions made in this thesis. N-grams were extracted, hitherto not covered by previous research into FSF detection using '*computational*' techniques. As can be seen from the overall results in Appendix X, these features pick up patterns in text that can separate out deceptive narrative.

LIWC 2015, released recently is the first time that the new significantly updated dictionaries of the tool are used to pick out various language constructs as delineated in chapter 4 over this new corpus. Again from Appendix R, it can be seen that the features extracted using this tool perform well in the classification task, especially in the balanced data set-up scenario.

From the literature review and the readability measures examined it is apparent that measures such as Gunning Fog and Flesch that are still used are limited and narrow in their assessment of readability in text. As far as can be determined this is the first study that takes in other measures that gauge coherence and cohesion in text that as outlined in chapter 2 influence reader comprehension. These new measure were extracted using the Coh-Metrix tool and as can be seen from the overall results in Appendix Q that they aid in separating narratives of fraud from a non-fraud firm. Again performance is better using the matched pair set-up.

A few choice variables extracted using Coh-Metrix and LIWC were then branded into ratios. These were first introduced by Zhou et al. [123] as ways to computationally derive linguistic cues of deception from text. Differently from a previous study [103] this study uses a number of different classifiers and a larger corpus to investigate success at the classification task using these cues. This as shown in Appendix S that the use of LBCs has resulted on a comparative basis in lower performing classifiers. Again this thesis has shown this to be the case using comparative evidence.

For the first time over key 10-K/annual reports this study attempted to examine '*what*' was said as opposed to '*how*' using a topic modelling technique over a corpus of composition described. The tool used Mallet output weights for each topic identified. These weight were then used to drive the downstream classifiers. The best results obtained are shown in Appendix T. It is clear that performance on the topic based classifier models could be improved. There is a need to better capture '*what*' was said. Using this new corpus, a new program was written with the WordNet tool deployed to pick up all synonyms in the text and bunch into concepts. This aided in distilling out key ideas mentioned in the reports. The results as shown in Appendix U indicate that there is a difference in concepts between the two reports. This is the first study that has illustrated this in the manner described in chapter 4 using 2 distinct data set-ups to enable a better appreciation of success at the classification task.

Doc Rep Scheme	Kappa	Sensi	Speci	ACC	95% CI	NIR	P Value [ACC> NIR]	Pos Pred Value	Neg Pred Value	Bal Acc	Feature Selection	Classifier
Unigrams	1	1	1	1	0.96,1	0.75	3.35e-13	1	1	1	PCA, Boruta	All
Bigrams	**0.76**	**0.72**	**0.98**	**0.92**	**0.84, 0.96**	**0.75**	**1.23e-05**	**0.94**	**0.91**	**0.85**	**Boruta**	**SGB**
Trigrams	0.64	0.60	0.97	0.88	0.80 0.93	0.75	0.001	0.88	0.88	0.78	Boruta	SGB
Coh-Metrix	**0.55**	**0.60**	**0.92**	**0.84**	**0.75 0.90**	**0.75**	**0.02**	**0.71**	**0.87**	**0.76**	**Boruta**	**SVM**
LIWC	**0.54**	**0.44**	**1**	**0.86**	**0.76 0.91**	**0.75**	**0.01**	**1**	**0.84**	**0.72**	**Boruta**	**SVM**
Custom Word List	0.36	0.32	0.97	0.81	0.72 0.88	0.75	0.09	0.80	0.81	0.64	Boruta	RF
LBCs	0.46	0.44	0.96	0.83	0.74 0.89	0.75	0.03	0.78	0.83	0.70	PCA	SGB
Topics	0.42	0.44	0.93	0.81	0.72 0.88	0.75	0.09	0.68	0.83	0.68	Boruta	SGB
Concepts	**0.59**	**0.60**	**0.94**	**0.85**	**0.78 0.91**	**0.75**	**0.002**	**0.78**	**0.87**	**0.77**	**Boruta**	**SVM**
LSA concepts	0.92	1	0.96	0.97	0.91 0.99	0.75	2.18e-09	0.89	1	0.98	No feature selectio	LR
Keywords	1	1	1	1	0.96 1	0.75	3.355e-13	1	1	1	All	All
Keywords:- Rutherford	1	1	1	1	0.96 1	0.75	3.355e-13	1	1	1	All	All

Table 7.1: Best Performing Classifiers on Document Representation Schemes – Peer Set

Doc Rep Scheme	Kappa	Sensi	Speci	ACC	95% CI	NIR	P Value [ACC> NIR]	Pos Pred Value	Neg Pred Value	Bal Acc	Feature Selection	Classifier
Unigrams	1	1	1	1	0.96,1	0.75	3.35e-13	1	1	1	PCA, Boruta, IG	All
Bigrams	**0.76**	**1**	**0.76**	**0.88**	**0.75 0.95**	**0.5**	**1.622e-08**	**0.80**	**1**	**0.88**	**IG**	**RF**
Trigrams	0.68	0.84	0.84	0.84	0.70 0.92	0.5	5.818e-07	0.84	0.84	0.94	IG	RF
Coh-Metrix	**0.68**	**0.84**	**0.84**	**0.84**	**0.70 0.92**	**0.5**	**5.818e-07**	**0.84**	**0.84**	**0.84**	**Boruta**	**RF**
LIWC	**0.72**	**0.92**	**0.80**	**0.86**	**0.73 0.94**	**0.5**	**1.049e-07**	**0.82**	**0.90**	**0.86**	**PCA**	**SGB**
Custom Word List	0.6	0.84	0.76	0.80	0.66 0.89	0.5	1.193e-05	0.77	0.82	0.80	Boruta	SVM
LBCs	0.48	0.68	0.80	0.74	0.59 0.85	0.5	0.0004	0.77	0.71	0.74	Boruta	RF
Topics	0.48	0.80	0.68	0.74	0.59 0.85	0.5	0.0004	0.71	0.77	0.74	PCA/ Boruta	RF/SGB
Concepts	**0.56**	**0.80**	**0.76**	**0.78**	**0.64 0.88**	**0.5**	**4.511e-05**	**0.76**	**0.79**	**0.78**	**IG**	**SGB**
LSA concepts	0.96	0.96	1	0.98	0.89 0.99	0.5	4.53e-14	1	0.96	0.98	No feature selection	SGB
Keywords	1	1	1	1	0.92 1	0.5	8.8822-16	1	1	1	All	All (except kNN)
Keywords: Rutherford	1	1	1	1	0.92 1	0.5	8.882e-16	1	1	1	All	All (except kNN)

Table 7.2: Best Performing Classifiers on Document Representation Schemes – Matched Pair

The potential for concepts to separate fraud from non-fraud firms based on their narrative was further illustrated by using a LSA to extract concepts. This as can be seen from the results in Appendix U (Table U.13) performed better. This is the first study that rigorously investigated ‘concepts’ in a corpus as described and produced the results as given.

Another computational technique, clustering using the k-means algorithm was also executed over the corpus, as an alternative to classification. Again from the results shown in Appendix Z this technique has potential to perform distinct clusters dependent upon the features used. This again as far as can be determined is the first study that showed clustering based on a wide range of features to hold promise in separating out a fraud from a non-fraud firm based on their narrative.

The best classifier results for each document representation scheme/feature selection routine is shown both in Appendix X and in tables 7.1 and 7.2. The best results have been slightly enlarged and are in bold. However this is not taking into account the results from the unigrams, LSA concepts, keywords and keywords-Rutherford as these models have attained a kappa of 1 (with few to none false positives and false negatives). They have been excluded to take account of the overfitting that could be producing such performance metrics. A larger test set that could be used to test further the discriminatory potential of these features would provide greater confidence in the results.

For both the peer set and matched pair data set up the chosen Coh-Metrix indices, LIWC variables, Concepts and Bigrams using the kappa values as an indicator produced the most robustly, discriminatory models. The results are overall better for the matched pair models, indicating that unbalanced data does negatively impact classifier performance. The Bigrams model indicates that picking up more context is beneficial and related to that the Concepts model by gathering all related terms has potential to be a differentiator between fraud and non-fraud reports. The theme of ‘*acquisition*’ seems to be key and is also picked up by the topic models. The Coh-Metrix indices and LIWC variables as features can be deigned as linguistic correlates of deception. It was shown amongst others that readability and pronoun use are key markers of deception in text. As indicated, Coh-Metrix and LIWC together would pick up such markers. The results shown in tables 7.1 and 7.2 and discussed in chapter 2, 4 and 6 show that such a claim would be valid.

7.1 Limitations and Future work

A greater sample of fraud reports would lend greater credence to the findings in this thesis. It is a truth self-evident that more '*ground truth*' of the observation under study would lead to better analysis and understanding. From a snapshot of the literature reviewed given in Appendix A, A.2 and A.3 this corpus and the fraud reports are about average size. Some researchers have managed to gather a lot more evidence, some a lot less. However, from the results given in this thesis and from the results garnered from the literature review examining the linguistic correlates of deception, such linguistic cues when used to drive downstream machine learning models have the potential to uncover FSF.

Notwithstanding that the feature gathering and extraction was extant, in a bid to represent the salient aspects of the documents, the examination of the text could still have been deeper. A probabilistic parser specifically trained on financial text would enable a detection of grammatical structure of sentences. This would allow not only a deeper understanding of the surface form of the language ie '*how*' ideas are conveyed but also a deeper insight into '*what*' was said. The latter aim could be improved through the use of a light weight ontology that captures prominent entities and relationships with respect to what constitutes valuable information. In other words "*Do the narratives convey quality information valuable to the larger stakeholder community or is it just clutter, boilerplate, irrelevant?*" *Is information material*? [342].

Material information is: "if *its omission or misrepresentation in the strategic report might reasonably be expected to influence the economic decisions shareholders make on the basis of the annual report as a whole*" [343]. Material information should relate to the areas shown in Figure 7.1.

The UK governement updated the Companies Act 2006 in 2013 in an attempt to direct firms to produce the information shown in Figure 7.1 into their annual reports. In particularly a new strategic report and directors' report has now become mandatory (small companies excluded). Some of the information that is required to be disclosed

Figure 7.1 Expected quality narrative [7].

and the manner of its disclousre is depicted in Figure 7.2. These requirements, plus information relating to factors shown in Figure 7.1 can be mapped onto an ontology and with the aid of a parser could delve further into ascertaining the quality of the information imparted in financial statements such as the AR. A set of rules would need to be defined to pick up material information conveyed.

Such an analysis as depicted in Figure 7.2 could also promote transparency in financial disclosure, a central aim of regulatory bodies. An automated quality check would ensure more firms meet their legal requirements. Consequently it could also highlight potential anomolies in a firms financial reporting such as FSF.

Further language markers of deception could possibly have been picked up from other sources apart from annual reports/10-K. Researchers such as Burgoon et al.[124], Larcker and Zakolyukina [160], examine vocal markers from conference calls made by top management to determine if deception is present. Other narratives that pertain to a firms financial reporting could also be amalgamated in a bid to reinforce any threads of deception.

Non-linguistic data has also been shown to indicate likely FSF. This can be ascertained from previous research as mapped out in Appendix A, Table A.2. This could be amalgamated with linguistic data to strenghten the fraud detection model building process. This was attempted by previous researchers [161, 169] as outlined in Appendix A, Table A.3. However the linguistic features extracted are much wider in this thesis than attempted by any previous researcher.

The vector space modelling of the documents, as described in chapter 4 is the classic modeling technique used in the Information retrieval [22, 48, 218]. Such an approach assumes: "*the meanings of words to be given and effectively atomic, without*

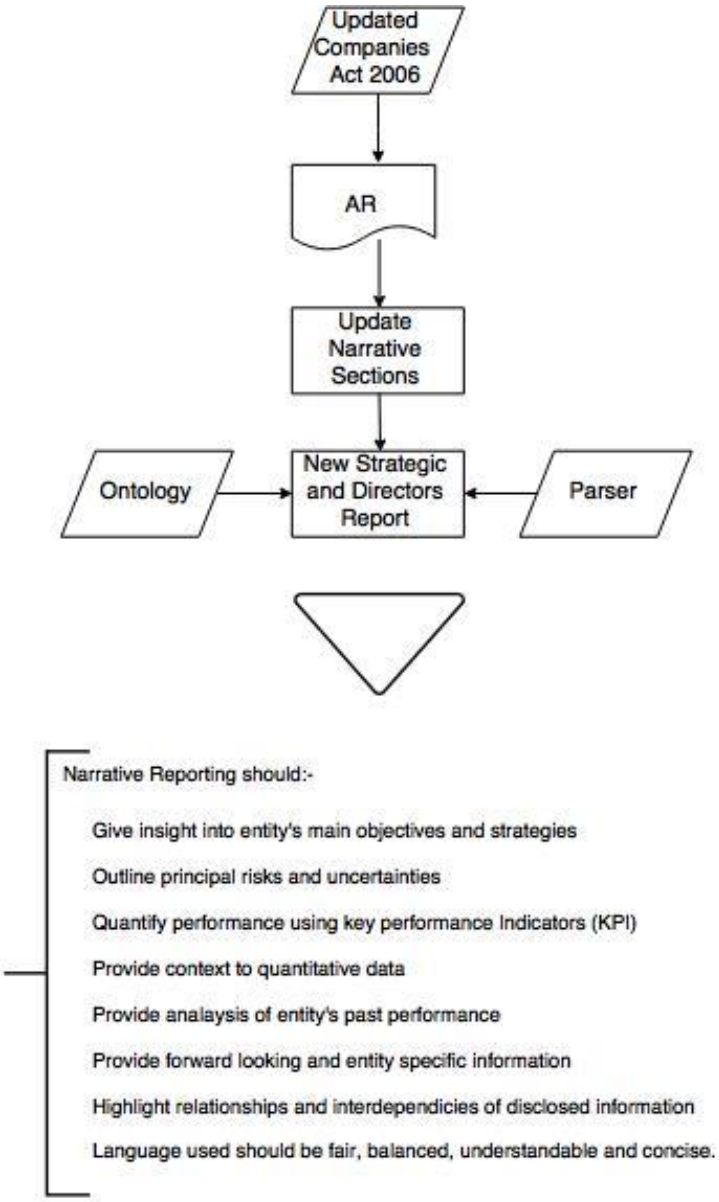


Figure 7.2 Further exploration to aid transparency and check quality of narrative.

any internal structure" [22]. However as Clark [22] adds : "*we would like a procedure which, given vectors for each word in a phrase or sentence, combines the vectors in some way to produce a single vector representing the meaning of the whole phrase or sentence. This would allow the meanings of whole phrases and sentences to be easily compared*". This would be done as previously for word vectors by using cosine measures between the sentence vectors. The order of words are disregarded in the traditional bag of words approach used to model documents in vector space. Figure

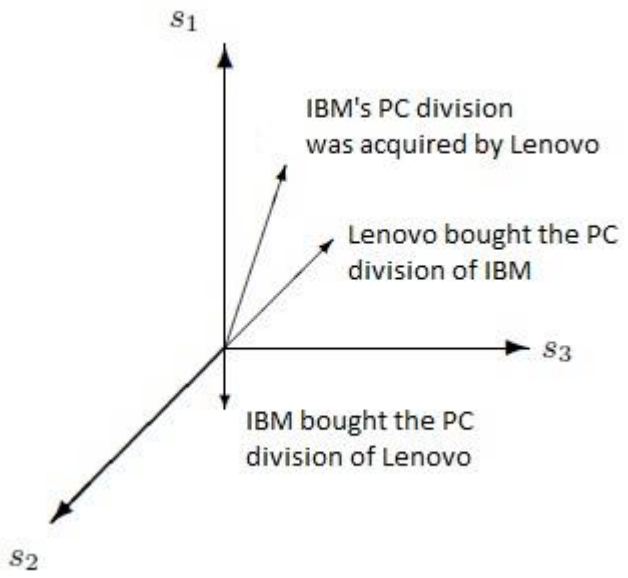


Figure 7.3 Capture of compositional nature of sentences.

7.3 shows that the order of words matters when it comes to sentence meaning [22]. An approach that captures the compositional meaning in a sentence which is then modelled in vector space is described by Clark [22]. This would mean that documents could be compared at a larger unit of analysis, with reduced ambiguity. Such a matrix of derived sentences with counts could then be passed to the classifers to aid in FSF detection. At a more extensive level such an approach would also help in determining the quality of the narrative as it would be easier to decipher what was said in the reports.

Another technique untried on the corpus are word embeddings, a successor to LSA and LDA. Their central construction is captured by the Firthian phrase: "*you shall know a word by the company it keeps*" [21]. Word embeddings are vectors that capture the semantic or contextual information of a word [381, 382]. Typically these embeddings are produced using tools like Word2vec. This is: "*a two-layer neural net that processes text. Its input is a text corpus and its output is a set of vectors: feature vectors for words in that corpus"* [383]. These word embeddings could then be used in a host of NLP tasks. For example the word embeddings could be produced for the keywords or the most frequent words identified in the corpus to further highlight the differences in their usage between fraud and non-fraud firms.

As can be viewed from the MDS and clustering results, separation of the two classes of documents fraud and non-fraud is possible. In some cases the k-means algorithm

produced well self-contained clusters. A techniques that uses clustering with supervised learning task could also aid in separating a fraud from a non-fraud firm. This combination of clustering with classification has been previously conducted with good results [344, 345].

A number of other classifier models could also have been tried to guage performance in this binary classification task. In particular Naive Bayes has been cited to perform well in a corpus of modest size and like LR is a high bias/low variance classifier and requires less data for training [313]. Artifical Neural Nets have also been tried in this data set and have been known to perform well where decision boundaries are non-linear [346]. It is a low bias/high variance classifier. They fulfil the goal of: "*endeavor to discover algorithms that can learn highly complex functions, with minimal need for prior knowledge*" [347]. Given the intractable and evolutionary nature of fraud. This aim would be particularly apt. However ANN are known to overfit and strategies must be deployed (eg tuning of parameters) to mitigate against this tendency.

7.2 Final Thoughts

Fraud and the deception it carries seems to be unstoppable. Headlines are blazoned on a regular basis on fraud cases detailing the extent of deception involved and the staggering sums swindled. Its prevalence is also astounding. Financial scams committed "*every 15 seconds*" [348]. Recent prominent FSF cases involved one of UK's biggest retailers Tesco with profits overstated by 326 million pounds [349] and Toshiba overstated it operating profits by a total $1.22 billion since 2008 [350]. Language is an under-utilized armoury that can aid in uncovering this misconduct. It is ubiquitous and its imprints found in the form of text is ingrained into our existence. Documents, reports, emails, blogs, tweets are abounding and growing. Eighty percent of all information in the world is in this unstructured form [23]. At a general level there is a need for language processing applications that can understand this flood of information to extract actionable knowledge. This thesis has documented some of the ways how this knowledge can be extracted for deception detection in financial documents. Surprisingly such work given the enormity of the problem is still small scale. Unaided deception is difficult to uncover, in financial reporting auditors detect relatively few significant frauds [351]. The thesis has demonstrated that aids can be

developed that can alert to potential anomalies. It has gathered ground truth evidence in the form of a corpus. It has investigated linguistic markers of deception and then using a variety of tools and techniques shown how such markers can be extracted from text. Lastly the application of state of the art machine learning algorithms using these markers show that narratives from fraud and non-fraud firms can be distilled. Given the scale of the problems, this is a strand of research worth pursuing.

BIBLIOGRAPHY

[1] J. Croft, "Fraud Costs the UK up to £193bn per year report says," in *Financial Times*, 25th May 2016 https://www.ft.com/content/fbb5c2e8-21ad-11e6-9d4d-c11776a5124d.

[2] Z. Rezaee and R. Riley, *Financial Statement Fraud: Prevention and Detection*, 2 ed.: Wiley, 2009.

[3] (2014, December 13) The Dozy Watchdogs. *The Economist*.

[4] J. Reiss, "Struggling Over the Soul of Economics: Objectivity Versus Expertise," in *Experts and Consensus in Social Science*, C. Martini and M. Boumans, Eds., ed Cham: Springer International Publishing, 2014, pp. 131-152.

[5] T. McEnery and A. Wilson, *Corpus Linguistics, An Introduction*: Edinburgh University Press, 2005.

[6] T. McEnery and A. Hardie, *Corpus linguistics: method, theory and practice*. Cambridge: Cambridge University Press, 2012.

[7] "The Future of Narrative Reporting, The Government Response," ed: Department of Business Innovation and Skills, March 2012.

[8] G. Ingersoll, T. Morton, and A. Farris, *Taming Text*. New York: Manning Publications Co, 2013.

[9] J. F. Sowa, "Why Has Artificial Intelligence Failed? And How Can it Succeed?," *Computación y Sistemas,* vol. 18, 2014.

[10] S. Pinker, *The Language Instinct*: New York, NY: Harper Perennial Modern Classics, 1994.

[11] E. Cambria and A. Hussain, *Sentic Computing: Techniques, Tools and Applications*. Dordrecht, Netherlands: Springer, 2012.

[12] KA Perera, K. Kuruppu, M. Gamage, S. Gunasekara, and G. Kodagoda, "A step towards a Natural Language Programming Tool (NLPT) " *NCTM - SLIIT,* June 2014.

[13] R. High, "The Era of Cognitive Systems: An Inside Look at IBM Watson and How it Works " Copyright IBM Corp. 2012. All rights reserved.2012.

[14] J. Bloomberg. (2014) Teaching Computers to Understand Meaning: From Banking to Healthcare. *Forbes*.

[15] "PARC Natural Language Processing," ed: Palo Alto Research Center, 2007.

[16] N. Chomsky, *Aspects of the Theory of Syntax*. Cambridge, Massachusetts: The MIT Press, 1965.

[17] D. Jurafsky and J. Martin, *Speech and Language Processing*. New Jersey: Pearson, Prentice Hall, 2014.

[18] D. B. Lenat, "CYC: A Large-Scale Investment in Knowledge Infrastructure," *Communications of the ACM,* vol. 38, November 1995.

[19] C. Locke, "Common Knowledge or Superior Ignorance?," *IEEE Expert,* 1990.

[20] W. Knight. (2016) An AI with 30 Years' Worth of Knowledge Finally Goes to Work. *MIT Technology Review*.

[21] J. R. Firth, in *Papers in Linguistics (1934-1951).* ed: Oxford: Oxford University Press, 1957.

[22] S. Clark, "Vector Space Models of Lexical Meaning," in *The Handbook of Contemporary Semantic Theory, 2nd Edition*, ed, 2015.

[23] O. Müller, I. Junglas, J. Brockeom, and S. Debortoli, "Utilizing big data analytics for information systems research: challenges, promises and guidelines," *European Journal of Information Systems,* vol. 25, pp. 289-302, 2016.

[24] J. Swales, "The concept of genre " in *The Discourse Studies Reader: Main currents in theory and analysis* ed, 1990, pp. 305-316.

[25] T. Rakow, "Risk, uncertainty and prophet: The psychological insights of Frank H. Knight," *Judgment and Decision Making,* vol. 5, pp. 458-466, 2010.

[26] P. Ormerod, "Ostrich Economics " 2009.

[27] M. Jensen and W. Meckling, "Theory of the Firm: Managerial Behavior, Agency Costs and Ownership Structure " *Journal of Financial Economics,* vol. 3, 1976.

[28] H. J. Baginski SP, Hillison WA, "Voluntary causal disclosures: tendencies and capital market reaction," *Rev Quant Account Finance* vol. 15, pp. 47-67, 2000.

[29] A. H. Adelberg, "Narrative disclosures contained in financial reports: means of communication or manipulation," *Account Bus Res* vol. 9, pp. 179-90, 1979.

[30] D. M. Merkl-Davies and N. Brennan, "Discretionary disclosure strategies in corporate narratives: incremental information or impression management," *Journal of Accounting Literature,* vol. 26, pp. 116–196, 2007.

[31] M. El-Haj, P. E. Rayson, S. Young, M. Walker, A. Moore, V. Athanasakou*, et al.*, "Learning tone and attribution for financial text mining.," in *Proceedings of LREC 2016, Tenth International Conference on Language Resources and Evaluation. ed. / Nicoletta Calzolari; Khalid Choukri; Thierry Declerck; Marko Grobelnik; Bente Maegaard; Joseph Mariani; Asuncion Moreno; Jan Odijk; Stelios Piperidis. European Language Resources Association (ELRA)*, 2016.
[32] A. Sen. (2010) The economist manifesto. *New Statesman*.
[33] A. Smith, *An Inquiry into the Nature and Causes of the Wealth of Nations* 1776.
[34] G. Zack, *Financial statement fraud: strategies for detection and investigation.* Hoboken, New Jersey: John Wiley & Sons, 2012.
[35] P. Ravisankar, V. Ravi, G. Raghava Rao, and I. Bose, "Detection of financial statement fraud and feature selection using data mining techniques," *Decision Support Systems,* vol. 50, pp. 491-500, 2011.
[36] J. West and M. Bhattacharya, "Some Experimental Issues in Financial Fraud Mining," *Procedia Computer Science,* vol. 80, pp. 1734-1744, // 2016.
[37] S. Goel, J. Gangolly, S. R. Faerman, and O. Uzuner, "Can Linguistic Predictors Detect Fraudulent Financial Filings?," *Journal of Emerging Technologies in Accounting,* vol. 7, pp. 25-46, 2010.
[38] E. Fitzpatrick and J. Bachenko, "Building a Data Collection for Deception Research," in *Proceedings of the EACL 2012 Workshop on Computational Approaches to Deception Detection,*, Avignon, France, 2012, pp. 31-38.
[39] J. Hirschberg and C. D. Manning, "Advances in natural language processing," *Science,* vol. 349, pp. 261-266, 2015.
[40] M. Zaki and W. Meira, *Data Mining and Analysis Fundamental Concepts and Algorithms*. New York: Cambridge University Press, 2014.
[41] A. O'Mara-Eves, J. Thomas, J. McNaught, M. Miwa, and S. Ananiadou, "Using text mining for study identification in systematic reviews: a systematic review of current approaches," *Systematic Reviews,* vol. 4, p. 5, 2015.
[42] R. Hasan, "Linguistic sign and the science of linguistics: the foundations of appliability," in *Developing Systemic Functional Linguistics*, Y. Fang and J. J. Webster, Eds., ed London: Equinox, 2014.
[43] M. Halliday, *An Introduction to Functional Grammar*, fourth ed. New York: Routledge, 2014.
[44] Mick O'Donnel, "Introduction to Systemic Functional Linguistics for Discourse Analysis. ," 2011.
[45] N. Chomsky, *Topics in the Theory of Generative Grammr*. The Hague: Mouton and Co N.V 1966.
[46] A. Carnie, *Syntax: A Generative Introduction (Introducing Linguistics)*: John Wiley and Sons, 2013.
[47] N. Chomsky, *Syntactic Structures*. Paris: Mouton, 1957.
[48] C. D. Manning and H. Schutze, *Foundations of Statistical Natural Language Processing*: MIT Press, 1999.
[49] C. D. Manning, P. Raghavan, and H. Schütze, *Introduction to information retrieval*: Cambridge University Press, 2012.
[50] D. S. McNamara, A. C. Graesser, P. M. McCarthy, and Z. Cai, *Automated Evaluation of Text and Discourse with Coh-Metrix*. New York: Cambridge University Press 2014.
[51] P. Baker and T. McEnery, *Corpora and Discourse: Integrating Discourse and Corpora.* London: Palgrave, Macmillan, 2015.
[52] V. Brezina, T. McEnery, and S. Wattam, "Collocations in context: A new perspective on collocation networks," *International Journal of Corpus Linguistics,* vol. 20, pp. 139-173, 2015.
[53] S. Jean, K. Cho, R. Memisevic, and Y. Bengio, "On using very large target vocabulary for neural machine translation," 2015.
[54] S. Young, M. Gasi, B. Thomson, and J. D. Williams, "POMDP-based Statistical Spoken Dialogue Systems: a Review," *PROC IEEE,* vol. 101, pp. 1160-1179, 2013.
[55] M. A. Russell, *Mining the Social Web, Data Mining Facebook, Twitter, LinkedIn, Google+, GitHub, and More*, 2nd Edition ed.: O'Reilly Media, 2013.
[56] M. Ott, C. Cardie, and J. T. Hancock, ""Estimating the prevalence of deception in online review communities."," in *21st International Conference on World Wide Web Conference,*, Lyon, France, 2012, pp. 201-210.
[57] N. Elhadad, L. Gravano, D. Hsu, S. Balter, V. Reddy, and H. Waechter, "Information extraction from social media for public health," presented at the KDD at Bloomberg Workshop, New York, 2014.

[58] D. Ferrucci, A. Levas, S. Bagchi, D. Gondek, and E. T. Mueller, "Watson: Beyond Jeopardy!," *Artificial Intelligence,* vol. 199-200, pp. 93-105, 2013.
[59] E. Davis and G. Marcus, "Commonsense reasoning and commonsense knowledge in artificial intelligence," *Communications of the ACM,* vol. 58, pp. 92-103, 2015.
[60] K. E. Markon, "Ontology, Measurement, and Other Fundamental Problems of Scientific Inference," *Psychological Inquiry,* vol. 26, pp. 259-262, 2015/07/03 2015.
[61] L. Breiman, "Statistical Modeling: The Two Cultures (with comments and a rejoinder by the author)," pp. 199-231, 2001/08 2001.
[62] P. Norvig, "On Chomsky and the Two Cultures of Statistical Learning," 2012.
[63] A. Atkinson and J. Stiglitz, "Lectures on public economics," ed: Princeton University Press, 2015.
[64] A. Pepper and J. Gore, "Behavioral agency theory: new foundations for theorizing about executive compensation," *Journal of Management,* vol. 41, pp. 1045-1068, 2015.
[65] G. Akerlof, "The market for "Lemons": Quality, Incertainty and Market Mechanism," *The Quarterly Journal of Economics,* vol. 83, pp. 488-500.
[66] P. M. Healy and K. G. Palepu, "Information asymmetry, corporate disclosure, and the capital markets: A review of the empirical disclosure literature," *Journal of Accounting and Economics,* vol. 31, pp. 405-440, 2001.
[67] A. Beyer, D. A. Cohen, T. Z. Lys, and B. R. Walther, "The financial reporting environment: Review of the recent literature," *Journal of Accounting and Economics,* vol. 50, pp. 296-343, 2010.
[68] M. Clatworthy and M. J. Jones, "The effect of thematic structure on the variability of annual report readability," *Accounting, Auditing & Accountability Journal,* vol. 14, pp. 311-326, 2001.
[69] K. C. Yalcin, "Market Rationality: Efficient Market Hypothesis versus Market Anomalies," *2016,* vol. 3, p. 16, 2016-02-03 2010.
[70] D. M. Merkl-Davies and N. M. Brennan, "A conceptual framework of impression management: new insights from psychology, sociology and critical perspectives," *Accounting and Business Research,* vol. 41, pp. 415-437, 2011.
[71] R. Benabou and G. Laroque, "Using Privelaged Information to Manipulate Marktes: Insiders, Gurus, and Credibility," *The Quarterly Journal of Economics,* vol. 104, 1992.
[72] S. Brown. (2016) Why cognitive bias is the investor's biggest foe. *Finweek*.
[73] H. Simon, "Theories of Bounded Rationality," in *Decsion and Organisation*, ed: North Holland Publishing Company, 1972.
[74] G. Mallard, *Bounded Rationality and Behavioural Economics*. New York: Routledge, 2016.
[75] D. Kahneman, *Thinking fast and slow*: Penguim Group, 2011.
[76] J. Hand, "A test of the Extended Functional Fixation Hypothesis," *The Accounting Review,* vol. 65, pp. 740-763, 1991.
[77] C. M. Schrand and B. R. Walther, "Strategic benchmarks in earnings announcements: The selective disclosure of prior-period earnings components," *Accounting Review,* vol. 75, pp. 151-177, 2000.
[78] J. R. Frederickson and J. S. Miller, "The Effects of Pro Forma Earnings Disclosures on Analysts' and Nonprofessional Investors' Equity Valuation Judgments," *The Accounting Review,* vol. 79, pp. 667-686, 2004/07/01 2004.
[79] F. Li, "Annual report readability, current earnings, and earnings persistence," *Journal of Accounting and Economics,* vol. 45, pp. 221-247, 2008.
[80] R. Bloomfield, "The "Incomplete Revelation Hypothesis" and Financial Reporting," *Accounting Horizons,* vol. 16, p. 233, 2002.
[81] H. Einhorn and R. Hogarth, "Confidence in judgment: Persistence of the illusion of validity," *Psychological Review,* vol. 85, pp. 395-416, 1978.
[82] J. Baird and R. Zelin, "The effects of information ordering on investor perceptions: an experiment utilizing president's letters," *Journal of Financial and strategic Decisions,* vol. 13, pp. 71-81, 2000.
[83] C. Mark and J. Michael John, "The effect of thematic structure on the variability of annual report readability," *Accounting, Auditing & Accountability Journal,* vol. 14, pp. 311-326, 2001/08/01 2001.
[84] V. Beattie, B. McInnes, and S. Fearnley, "A methodology for analysing and evaluating narratives in annual reports: a comprehensive descriptive profile and metrics for disclosure quality attributes," *Accounting Forum,* vol. 28, pp. 205-236, 9// 2004.
[85] S. Minhas, S. Poria, A. Hussain, and K. Hussainey, "A review of artificial intelligence and biologically inspired computational approaches to solving issues in narrative financial

disclosure," presented at the Proceedings of the 6th international conference on Advances in Brain Inspired Cognitive Systems, Beijing, China, 2013.
[86] S. P. Kothari, X. Li, and J. E. Short, "The Effect of Disclosures by Management, Analysts, and Business Press on Cost of Capital, Return Volatility, and Analyst Forecasts: A Study Using Content Analysis," *The Accounting Review,* vol. 84, pp. 1639-1670, 2009.
[87] E. Demers and C. Vega, "Linguistic Tone in Earnings Announcements: News or Noise?," ed: Working Paper, INSEAD., 2011.
[88] R. Feldman, S. Govindaraj, J. Livnat, and B. Segal, "The Incremental Information Content of Tone Change in Management Discussion and Analysis," *http://ssrn.com/abstract=1126962.*, 2008.
[89] P. C. Tetlock, M. Saar-Tsechansky, and S. Macskassy, "More than Words: Quantifying Language to Measure Firms' Fundamentals," *The Journal of Finance,* vol. 63, pp. 1437-1467, 2008.
[90] F. Li, "The Information Content of Forward-Looking Statements in Corporate Filings—A Naïve Bayesian Machine Learning Approach," *Journal of Accounting Research,* vol. 48, pp. 1049-1102, 2010.
[91] S. V. Brown and J. W. Tucker, "Large-Sample Evidence on Firms' Year-over-Year MD&A Modifications," *Journal of Accounting Research,* vol. 49, pp. 309-346, 2011.
[92] D. Slattery, "The power of language in corporate financial reports," *Communication and Language at work,* vol. 1, pp. 55-63, 2014.
[93] J. West and M. Bhattacharya, "Intelligent financial fraud detection," *Comput. Secur.,* vol. 57, pp. 47-66, 2016.
[94] "Occupational Fraud & Abuse.," Association of Certified Fraud Examiners (ACFE)2012.
[95] S. Chen, "Detection of fraudulent financial statements using the hybrid data mining approach," *SpringerPlus,* vol. 5, 2016.
[96] E. W. T. Ngai, Y. Hu, Y. H. Wong, Y. Chen, and X. Sun, "The application of data mining techniques in financial fraud detection: A classification framework and an academic review of literature," *Decision Support Systems,* vol. 50, pp. 559-569, 2011.
[97] Z. Rezaee, "Causes, consequences, and deterence of financial statement fraud," *Critical Perspectives on Accounting,* vol. 16, pp. 277-298, 2005.
[98] M. Beasley, "The Empirical Analysis of the Relation Between th Board of Director Composition and Financial Statement Fraud," *The Accounting Review,* vol. 71, pp. 443-465, 1996.
[99] M. S. Beasley, J. V. Carcello, D. R. Hermanson, and T. Neal, "Fraudulent Financial Reporting 1998-2007," 2010.
[100] N. Mohamed and M. Handley-Schachelor, "Financial Statement Fraud Risk Mechanisms and Strategies: The Case Studies of Malaysian Commercial Companies," *Procedia - Social and Behavioral Sciences,* vol. 145, pp. 321-329, 2014.
[101] "Report to the Nations on Occupational Fraud and Abuse," Association of Certified Fraud Examiners2014.
[102] D. Roden, S. Cox, and K. Joung, "The Fraud Triangle as a Predictor of Corporate Fraud," *Academy of Accounting & Financial Studies Journal* vol. 20, pp. 80-92, 2016.
[103] S. L. Humpherys, K. C. Moffitt, M. B. Burns, J. K. Burgoon, and W. F. Felix, "Identification of fraudulent financial statements using linguistic credibility analysis," *Decision Support Systems,* vol. 50, pp. 585-594, 2011.
[104] C. Albrecht, D. Holland, R. Malagueño, S. Dolan, and S. Tzafrir, "The Role of Power in Financial Statement Fraud Schemes," *Journal of Business Ethics,* vol. 131, pp. 803-813, 2015.
[105] D. R. Cressey, "The Differential Association Theory and Compulsive Crimes," *The Journal of Criminal Law, Criminology, and Police Science,* vol. 45, pp. 29-40, 1954.
[106] S. Y. Huang, C.-C. Lin, A.-A. Chiu, and D. C. Yen, "Fraud detection using fraud triangle risk factors," *Information Systems Frontiers,* pp. 1-14, 2016.
[107] A. Schuchter and M. Levi, "The Fraud Triangle revisited," *Security Journal,* vol. 29, pp. 107-121, 2016.
[108] "Deterring and Detecting Financial Reporting Fraud A Platform for Action," Center for Audit Quality2010.
[109] O. B. Ani, "Fraudulent Financial Reporting: The Nigerian Experience," in *The Clute Institute International Academic Conference*, San Antonio, 2014.
[110] M. E. Lokanan, "Challenges to the fraud triangle: Questions on its usefulness," *Accounting Forum,* vol. 39, pp. 201-224, 9// 2015.

[111] A. Gepp and K. Kumar, "Predicting Financial Distress: A Comparison of Survival Analysis and Decision Tree Techniques," *Procedia Computer Science,* vol. 54, pp. 396-404, 2015/01/01 2015.
[112] J. Dorminey, A. S. Fleming, M.-J. Kranacher, and R. A. R. Jr, "The Evolution of Fraud theory," *Issues in Accounting Education,* vol. 27, pp. 555-579, 2012.
[113] P. M. Dechow, W. Ge, C. R. Larson, and R. G. Sloan, "Predicting Material Accounting Misstatements*," *Contemporary Accounting Research,* vol. 28, pp. 17-82, 2011.
[114] L. Peng, "The Importance of Fraud Detection Techniques From the Enron Case and the T.J. Maxx Data Breach," Masters, James Madison University, 2013.
[115] P. Meyer, *Liespotting: Proven Techniques to Detect Deception* St Martin's Press, 2010.
[116] (2016) Ponzis to punters. *The Economist*.
[117] R. Shiller, *Irrational Exuberance*. New Jersey: Princeton University Press, 2005.
[118] J. Bachenko, E. Fitzpatrick, and M. Schonwetter, "Verification and implementation of language-based deception indicators in civil and criminal narratives," presented at the Proceedings of the 22nd International Conference on Computational Linguistics - Volume 1, Manchester, United Kingdom, 2008.
[119] C. M. Fuller, D. P. Biros, J. Burgoon, and J. Nunamaker, "An Examination and Validation of Linguistic Constructs for Studying High-Stakes Deception," *Group Decision and Negotiation,* vol. 22, pp. 117-134, 2012.
[120] P. M. McCarthy, N. D. Duran, and L. M. Booker, "The Devil Is in the Details: New Directions in Deception Analysis," in *Twenty-Fifth International Florida Artificial Intelligence Research Society Conference*, Florida, 2012.
[121] V. L. Rubin and N. Conroy, *Discerning truth from deception: Human judgments and automation efforts*, 2012.
[122] K. Moffitt, W. Felix, and J. K. Burgoon, "Using Lexical Bundles to Discriminate between Fraudulent and Non-fraudulent Financial Reports," presented at the SIG-ASYS Pre-ICIS 2010 workshop, 2010.
[123] L. Zhou, J. Burgoon, J. Nunamker, and D. Twitchell, "Automating Linguistics-Based Cues for Detecting Deception in Text-based Asynchronous Computer Mediated Communication," *Group Decision and Negotiation,* vol. 13, pp. 81-106, 2004.
[124] J. Burgoon, W. J. Mayew, J. S. Giboney, A. C. Elkins, K. Moffitt, B. Dorn*, et al.*, "Which Spoken Language Markers Identify Deception in High-Stakes Settings? Evidence From Earnings Conference Calls," *Journal of Language and Social Psychology,* vol. 35, pp. 123-157, 2015.
[125] D. B. Buller and J. K. Burgoon, "Interpersonal Deception Theory," *Communication Theory,* vol. 6, pp. 203-242, 1996.
[126] T. Loughran and B. McDonald, "When Is a Liability Not a Liability? Textual Analysis, Dictionaries, and 10-Ks," *The Journal of Finance,* vol. 66, pp. 35-65, 2011.
[127] M. L. Newman, J. W. Pennebaker, D. S. Berry, and J. M. Richards, "Lying Words: Predicting Deception From Linguistic Styles," *Personality and Social Psychology Bulletin* vol. 29, pp. 665-675, 2003.
[128] M. B. Burns and K. C. Moffitt, "Automated deception detection of 911 call transcripts," *Security Informatics,* vol. 3, p. 8, 2014.
[129] B. G. Amado, R. Arce, F. Fariña, and M. Vilariño, "Criteria-Based Content Analysis (CBCA) reality criteria in adults: A meta-analytic review," *International Journal of Clinical and Health Psychology,* vol. 16, pp. 201-210, 5// 2016.
[130] V. Oberlader, C. Naefgen, J. Koppehele-Gossel, L. Quinten, R. Banse, and A. Schmidt, "Validity of content-based techniques to distinguish true and fabricated statements: A meta-analysis.," *Law Human Behaviour,* vol. 40, pp. 440-57, 2016.
[131] A. Vrij, S. Mann, K. Susanne, and R. P. Fisher, "Cues to Deception and Ability to Detect Lies as a Function of Police Interview Styles," *Law and Human Behavior,* vol. 31, pp. 499-518, 2007.
[132] J. T. Hancock, L. E. Curry, S. Goorha, and M. Woodworth, "On Lying and Being Lied To: A Linguistic Analysis of Deception in Computer-Mediated Communication," *Discourse Processes,* vol. 45, pp. 1-23, 2007.
[133] V. Hauch, I. Blandon-Gitlin, J. Masip, and S. L. Sporer, "Are Computers Effective Lie Detectors? A Meta-Analysis of Linguistic Cues to Deception," *Personality and Social Psychology Review,* vol. 19, pp. 307-342, 2014.
[134] A. Mehrabian, "Attitudes inferred from non-immediacy of verbal communications," *Journal of Verbal Learning and Verbal Behavior,* vol. 6, pp. 294-295, 1967/04/01 1967.
[135] p. ekman, "Emotional and conversational nonverbal signals," 2004.

[136] J. K. Courtis, "Corporate report obfuscation: artefact or phenomenon?," *The British Accounting Review,* vol. 36, pp. 291-312, 9// 2004.
[137] R. J. Bloomfield, "The “Incomplete Revelation Hypothesis” and Financial Reporting," *Accounting Horizons,* vol. 16, pp. 233-243, 2002.
[138] B. A. Rutherford, "Obfuscation, Textual Complexity and the Role of Regulated Narrative Accounting Disclosure in Corporate Governance," *Journal of Management and Governance,* vol. 7, pp. 187-210, 2003.
[139] K. Moffitt and M. Burns, "What Does That Mean? Investigating Obfuscation and Readability Cues as Indicators of Deception in Fraudulent Financial Reports," in *AMCIS 2009* 2009.
[140] E. Baraibar-Diez, M. D. Odriozola, and J. L. Fernández Sánchez, "A Survey of Transparency: an Intrinsic Aspect of Business Strategy," *Business Strategy and the Environment,* pp. n/a-n/a, 2016.
[141] N. D. Duran, C. Hall, P. M. McCarthy, and D. S. McNamara, "The linguistic correlates of conversational deception: Comparing natural language processing technologies," *Applied Psycholinguistics,* vol. 31, pp. 439-462, 2010.
[142] A. Bailin and A. Grafstein, *Readability: Text and Context*: Palgrave Macmillan, 2016.
[143] T. Loughran and B. McDonald, "Measuring Readability in Financial Disclosures," *The Journal of Finance,* vol. 69, pp. 1643-1671, 2014.
[144] D. S. Carstens and G. Vandlandingham, "Through a Glass, Darkly: The Promise and Reality of State Transparency Websites " *Online Journal of Applied Knowledge Management,* vol. 3, 2015.
[145] N. Fligstein and A. F. Roehrkassea, "The Causes of Fraud in the Financial Crisis of 2007 to 2009 Evidence from the Mortgage-Backed Securities " *American Sociological Review,* pp. 1-27, 2016.
[146] M. Halliday and R. Hasan, *Cohesion in English*: Routeledge, 1976.
[147] D. S. McNamara, E. Kintsch, N. B. Songer, and W. Kintsch, "Are Good Texts Always Better? Interactions of Text Coherence, Background Knowledge, and Levels of Understanding in Learning from Text," *Cognition and Instruction,* vol. 14, pp. 1-43, 1996.
[148] T. Landauer, D. McNamara, S. Dennis, and W. Kintsch, *Handbook of latent semantic analysis*. New Jersey: Mahwah, 2007.
[149] K. Cain and H. M. Nash, "The influence of connectives on young readers' processing and comprehension of text," *Journal of Educational Psychology,* vol. 103, pp. 429-441, 2011.
[150] A. C. Graesser and D. S. McNamara, "Computational Analyses of Multilevel Discourse Comprehension," *Topics in Cognitive Science,* vol. 3, pp. 371-398, 2011.
[151] M. Davoudi and H. R. H. Moghadam, "Critical Review of the Models of Reading Comprehension with a Focus on Situation Models," *International Journal of Linguistics,* vol. 7, p. 172, 2015.
[152] M. Coltheart, "The MRC psycholinguistic database," *The Quarterly Journal of Experimental Psychology Section A,* vol. 33, pp. 497-505, 2007.
[153] E. Charniak, "A maximum-entropy-inspired parser," presented at the Proceedings of the 1st North American chapter of the Association for Computational Linguistics conference, Seattle, Washington, 2000.
[154] M. P. Marcus, M. A. Marcinkiewicz, and B. Santorini, "Building a large annotated corpus of English: the penn treebank," *Comput. Linguist.,* vol. 19, pp. 313-330, 1993.
[155] T. A. V. Dijik and W. Kintsch, "Strategies of Discourse Comprehension," New York1983.
[156] M. Templin, "Certain language skills in children," University of Minnesota Press, Minneapolis:1957.
[157] L. M. Van Swol and M. T. Braun, "Communicating Deception: Differences in Language Use, Justifications, and Questions for Lies, Omissions, and Truths," *Group Decision and Negotiation,* vol. 23, pp. 1343-1367, 2013.
[158] P. Kalbfleisch, "Deceit, distrust and the social milieu: Application of deception research in a troubled world," *Journal of Applied Communication Research,* vol. 20, pp. 308-334, 1992.
[159] T. Fornaciari and M. Poesio, "On the use of homogenous sets of subjects in deceptive language analysis," presented at the Proceedings of the Workshop on Computational Approaches to Deception Detection, Avignon, France, 2012.
[160] D. F. Larcker and A. A. Zakolyukina, "Detecting Deceptive Discussions in Conference Calls," *Journal of Accounting Research,* vol. 50, pp. 495-540, 2012.
[161] M. Cecchini, H. Aytug, G. J. Koehler, and P. Pathak, "Making words work: Using financial text as a predictor of financial events," *Decision Support Systems,* vol. 50, pp. 164-175, 2010.
[162] Y.-J. Chen, "On Fraud Detection Method for Narrative Annual Reports," in *Proceedings of The Fourth International Conference on Informatics & Applications*, Japan, 2015.

[163] W. Dong, S. Liao, B. Fang, X. Cheng, Z. Chen, and W. Fan, "The Detection of fraudulent financial startements: An Integrated Language Model," in *PACIS 2014 Proceedings*, 2014.
[164] W. Dong, S. Liao, and L. Liang, "Financial Statement Fraud Detection using Text Mining: A Systemic Functional Linguistics Theory Perspective," in *PACIS 2016 Proceedings*, 2016.
[165] F. H. Glancy and S. B. Yadav, "A computational model for financial reporting fraud detection," *Decision Support Systems,* vol. 50, pp. 595-601, 2011.
[166] C.-H. Lee, E. Lusk, and M. Halperin, "Content Analysis for Detection of Reporting Irregularities: Evidence from Restatements during the SOX-Era," *Journal of Forensic and Investigative Accounting,* vol. 6, 2014.
[167] C.-C. Lee, N. T. Churyk, and D. Clinton, "Validating Early Fraud Prediction using narrative disclosures," *Journal of Forensic and Investigative Accounting,* vol. 5, 2013.
[168] L. Purda and D. Skillicorn, "Accounting Variables, Deception, and a Bag of Words: Assessing the Tools of Fraud Detection," *Contemporary Accounting Research,* vol. 32, pp. 1193-1223, 2015.
[169] C. S. Throckmorton, W. J. Mayew, M. Venkatachalam, and L. M. Collins, "Financial fraud detection using vocal, linguistic and financial cues," *Decision Support Systems,* vol. 74, pp. 78-87, 2015.
[170] M. Zuckerman, B. M. DePaulo, and R. Rosenthal, "Verbal and Nonverbal Communication of Deception1," in *Advances in Experimental Social Psychology*. vol. Volume 14, B. Leonard, Ed., ed: Academic Press, 1981, pp. 1-59.
[171] S. Adams and J. Jarvis, "Indicators of veracity and deception: an analysis of written statements made to police," *International Journal of speech language and the law,* vol. 13, 2006.
[172] B. Wang and X. Wang, "Deceptive Financial Reporting Detection: A Hierarchical Clustering Approach Based on Linguistic Features," *Procedia Engineering,* vol. 29, pp. 3392-3396, 2012.
[173] W. Zhou and G. Kapoor, "Detecting evolutionary financial statement fraud," *Decision Support Systems,* vol. 50, pp. 570-575, 2011.
[174] R. Kanapickienė and Ž. Grundienė, "The Model of Fraud Detection in Financial Statements by Means of Financial Ratios," *Procedia - Social and Behavioral Sciences,* vol. 213, pp. 321-327, 2015.
[175] M. E. Alden, D. M. Bryan, B. J. Lessley, and A. Tripathy, "Detection of Financial Statement Fraud Using Evolutionary Algorithms," *Journal of Emerging Technologies in Accounting,* vol. 9, pp. 71-94, 2012.
[176] N. S. Gill and R. Gupta, "Analysis of Data Mining Techniques for Detection of Financial Statement Fraud," *The IUP Journal of Systems Management,* vol. X, pp. 7-15, 2012.
[177] Y. J. Kim, B. Baik, and S. Cho, "Detecting financial misstatements with fraud intention using multi-class cost-sensitive learning," *Expert Systems with Applications,* vol. 62, pp. 32-43, 2016.
[178] R. Li, "Detection of Financial Reporting Fraud Based on Clustering Algorithm of Automatic Gained Parameter K Value," *International Journal of Database Theory and Application,* vol. 8, pp. 157-168, 2015.
[179] P.-F. Pai, M.-F. Hsu, and M.-C. Wang, "A support vector machine-based model for detecting top management fraud," *Knowledge-Based Systems,* vol. 24, pp. 314-321, 2011.
[180] J. Perols, "Financial Statement Fraud Detection: An Analysis of Statistical and Machine Learning Algorithms," *AUDITING: A Journal of Practice & Theory,* vol. 30, pp. 19-50, 2011.
[181] X.-P. Song, Z.-H. Hu, J.-G. Du, and Z.-H. Sheng, "Application of Machine Learning Methods to Risk Assessment of Financial Statement Fraud: Evidence from China," *Journal of Forecasting,* vol. 33, pp. 611-626, 2014.
[182] Tarjo and N. Herawati, "Application of Beneish M-Score Models and Data Mining to Detect Financial Fraud," *Procedia - Social and Behavioral Sciences,* vol. 211, pp. 924-930, 2015.
[183] D. G. Whiting, J. V. Hansen, J. B. McDonald, C. Albrecht, and W. S. Albrecht, "Machine Learning Methods for Detecting Patterns of Management Fraud," *Computational Intelligence,* vol. 28, pp. 505-527, 2012.
[184] M El-Haj, P. Rayson, S. Young, and M. Walker, "Detecting document structure in a very large corpus of UK financial reports.," in *Proceedings of the ninth international conference on language resources and evaluation*, 2014.
[185] S. Young, "The drivers, consequences and policy implications of non-GAAP earnings reporting. ," *Accounting and Business Research,* vol. 40, pp. 444-465, 2014.
[186] S. Abraham and P. J. Shrives, "Improving the relevance of risk factor disclosure in corporate annual reports," *The British Accounting Review,* vol. 46, pp. 91-107, 3// 2014.
[187] S. L. Summers and J. T. Sweeney, "Fraudulently Misstated Financial Statements and Insider Trading: An Empirical Analysis," *The Accounting Review,* vol. 73, pp. 131-146, 1998.

[188] P. Rayson, "Computational Tools and Methods for Corpus Compilation and Analysis," in *Cambridge Handbook of English Corpus Linguistics*, ed, 2015.
[189] P. Rayson, "From key words to key semantic domains," *International Journal of Corpus Linguistics,* vol. 13, pp. 519-549, 2008.
[190] B. C. Camiciottoli, *Rhetoric in financial discourse*. The Netherlands: Rodopi, 2013.
[191] S. T. Piantadosi, "Zipf's word frequency law in natural language: A critical review and future directions," *Psychonomic Bulletin & Review,* vol. 21, pp. 1112-1130, 2014.
[192] *Corpus Linguistics An International Handbook* vol. 2. Berlin: Walter de Gruyter GmbH and Co, 2009.
[193] G. Zipf, *Human Behavior and the Principle of Least Effort*. New York: Addison-Wesley, 1949.
[194] A. Baron, P. Rayson, and D. Archer, "Word frequency and key word statistics in historical corpus linguistics," *International Journal of English Studies,* vol. 20, pp. 41-67.
[195] L. Anthony. (2014). *AntConc (Version 3.4.3). Available from http://www.laurenceanthony.net/*.
[196] A. Kilgarriff, "Language is never, ever, ever, random," in *Corpus Linguistics and Linguistic Theory* vol. 1, ed, 2005, p. 263.
[197] J. Lijffijt, T. Nevalainen, T. Säily, P. Papapetrou, K. Puolamäki, and H. Mannila, "Significance testing of word frequencies in corpora," *Digital Scholarship in the Humanities,* 2014.
[198] P. Rayson and R. Garside, "Comparing corpora using frequency profiling " in *Proceedings of the workshop on Comparing Corpora*, Hong Kong, 2000.
[199] B. Rutherford, "Genre Analysis of Corporate Annual Report Narratives: A Corpus Linguistics Based Approach," *Journal of Business Communication,* vol. 42, pp. 349-378, 2005.
[200] R. Larson and B. Farber, *Elementary Statistics: Picturing the World*. England: Pearson, 2015.
[201] W. Anderson and J. Corbett, *Exploring English With Online Corpora: An Introduction*. China: Palgrave Macmillan, 2009.
[202] P. Rayson, "Corpus Analysis of Key Words," in *The Encyclopedia of Applied Linguistics*, ed: Blackwell Publishing Ltd, 2012.
[203] M. Bondi and M. Scott, *Keyness in Texts*: John Benjamins Publishing Company, 2010.
[204] P. Rayson, D. Berridge, and B. Francis, "Extending the Cochran rule for the comparison of word frequencies between corpora," in *Proceedings of the 7th International Conference on Statistical analysis of textual data* Belgium, 2004, pp. 926-936.
[205] T. Loughran and B. McDonald, "The Use of Word Lists in Textual Analysis," *Forthcoming in the Journal of Behavioral Finance,* 2015.
[206] I. Pollach, "Taming Textual Data: The Contribution of Corpus Linguistics to Computer-Aided Text Analysis," *Organizational Research Methods,* vol. 15, pp. 263-287, 2011.
[207] S. Gries, "Dispersions and adjusted frequencies in corpora," *International Journal of Corpus Linguistics,* vol. 13, 2008.
[208] J. Sinclair, *Corpus, Concordance, Collocation*. Oxford: Oxford University Press, 1991.
[209] J. R. Firth, *Papers in Linguistics 1934–1951*. London: Oxford University Press, 1957.
[210] N. Schmitt, "Formulaic Language and Collocation," 2012.
[211] S. T. Gries and N. C. Ellis, "Statistical Measures for Usage-Based Linguistics," *Language Learning,* vol. 65, pp. 228-255, 2015.
[212] D. Lindemann and I. S. Vicente, "Building Corpus-based Frequency Lemma Lists," *Procedia - Social and Behavioral Sciences,* vol. 198, pp. 266-277, 2015/07/24 2015.
[213] M. Keikha, A. Khonsari, and F. Oroumchian, "Rich document representation and classification: An analysis," *Knowledge-Based Systems,* vol. 22, pp. 67-71, 2009.
[214] T. Strzalkowski, "Document Representation in Natural Language Text Retrieval," in *Proceedings of the Human Language Technology (HLT) Conference*, 1994, pp. 364-369.
[215] P. Flach, *Machine Learning: The Art and Science of Algorithms that Make Sense of Data*: Cambridge University Press, 2012.
[216] S. Jayanthi and S. Prema, "MEASURING THE PERFORMANCE OF SIMILARITY PROPAGATION IN AN SEMANTIC SEARCH ENGINE," *ICTACT JOURNAL ON SOFT COMPUTING,* vol. 4, 2013.
[217] G. Giannakopoulos, P. Mavridi, G. Paliouras, G. Papadakis, and K. Tserpes, "Representation models for text classification: a comparative analysis over three web document types," presented at the Proceedings of the 2nd International Conference on Web Intelligence, Mining and Semantics, Craiova, Romania, 2012.
[218] P. Turney and P. Pantel, "From Frequency to Meaning: Vector Space Models of Semantics," *Journal of Artificial Intelligence Research,* vol. 37, pp. 141 -188, 2010.
[219] Q. Le and T. Mikolov, "Distributed Representations of Sentences and Documents," in *Proceedings of the 31 st International Conference on Machine Learning*, Beijing, 2014.

[220] S. Bird, E. Klein, and E. Loper, *Natural Language Processing with Python* USA: O Reilly Media Inc, 2009.
[221] Y. Zhang, Y. Zhang, J. Xu, C. Xing, and H. Chen, "Sentiment Analysis on Chinese Health Forums: A Preliminary Study of Different Language Models," in *Smart Health: International Conference, ICSH 2015, Phoenix, AZ, USA, November 17-18, 2015. Revised Selected Papers*, X. Zheng, D. D. Zeng, H. Chen, and S. J. Leischow, Eds., ed Cham: Springer International Publishing, 2016, pp. 68-81.
[222] C.-M. Tan, Y.-F. Wang, and C.-D. Lee, "The use of bigrams to enhance text categorization," *Inf. Process. Manage.,* vol. 38, pp. 529-546, 2002.
[223] S. Wang and C. D. Manning, "Baselines and bigrams: Simple, good sentiment and topic classification," in *Proceedings of the International Conference on Machine Learning*, 2012.
[224] R. Bekkerman and J. Allan, "Using bigrams in text categorization," Center of Intelligent Information Retrieval, Amherst, Technical Report IR-4082004.
[225] M. H. Sawyer, "A REVIEW OF RESEARCH IN REVISING INSTRUCTIONALTEXT," *Journal of Literacy Research,* vol. XXIII, pp. 307-333, 1991.
[226] K. M. Sheehan, I. Kostin, Y. Futagi, and M. Flor, "Generating Automated Text Complexity Classifications That Are Aligned With Targeted Text Complexity Standards ", New Jersey2010.
[227] G. C. Biddle, G. Hilary, and R. S. Verdi, "How does financial reporting quality relate to investment efficiency?," *Journal of Accounting and Economics,* vol. 48, pp. 112-131, 2009.
[228] A. Lawrence, "Individual investors and financial disclosure," *Journal of Accounting and Economics,* vol. 56, pp. 130-147, 7// 2013.
[229] B. P. Miller, "The Effects of Reporting Complexity on Small and Large Investor Trading," *The Accounting Review,* vol. 85, pp. 2107-2143, 2010.
[230] K. Lo, F. Ramos, and R. Rogo, "Earnings management and annual report readability," *Journal of Accounting and Economics,* vol. 63, pp. 1-25, 2// 2017.
[231] J. Pennebaker, R. Boyd, K. Jordan, and K. Blackburn, "The development and psychometric properties of LIWC2015," University of Texas, Austin2015.
[232] Y. R. Tausczik and J. W. Pennebaker, "The Psychological Meaning of Words: LIWC and Computerized Text Analysis Methods," *Journal of Language and Social Psychology,* vol. 29, pp. 24-54, 2009.
[233] P. Rayson, "Matrix: A Statistical Method and Software Tool for Linguistic Analysis through Corpus Comparison," PhD, Lancaster University, 2003.
[234] R. Jordan, *Academic Writing Course*. London: Longman, 1999.
[235] S. Bonsall, Z. Bozanic, and P. Fischer. (2013). *The Informativeness of Disclosure Tone, Available at SSRN: https://ssrn.com/abstract=1598364*.
[236] Z. Bozanic, D. Roulstone, and A. V. Buskirk, "Management Earnings Forecasts and Forward-looking Statements.," The Ohio State University2012.
[237] R. Nyman, D. Gregory, S. Kapadia, P. Ormerod, P. Tuckett, and R. Smith, "News and narratives in financial systems: exploiting big data for systemic risk assessment.," 2014.
[238] A. Bodnaruk, T. Loughran, and B. McDonald, "Using 10-K Text to Gauge Financial Constraints," *Journal of Financial and Quantitative Analysis,* vol. 50, pp. 623-646, 2015/008/001 2015.
[239] B. Matthies and A. Coners, "Computer-Aided Text Analysis of Corporate Disclosures - Demonstration and Evaluation of Two Approaches," *The International Journal of Digital Accounting Research,* vol. 15, pp. 69-98, 2015.
[240] N. Brown, R. Crowley, and W. Elliott, "What are you saying? Using topic to detect financial misreporting. Working paper.," 2015.
[241] C. Lewis, "Keynote address," presented at the The 26th XBRL International Conference, 2013.
[242] D. M. Blei, A. Y. Ng, and M. I. Jordan, "Latent dirichlet allocation," *J. Mach. Learn. Res.,* vol. 3, pp. 993-1022, 2003.
[243] D. M. Blei, "Probabilistic topic models," *Communications of the ACM,* vol. 55, p. 77, 2012.
[244] A. K. McCallum. (2002). *MALLET: A Machine Learning for Language Toolkit http://mallet.cs.umass.edu.*
[245] C. Fellbaum, "WordNet and wordnets " in *Encyclopedia of Language and Linguistic*, ed Oxford: Elsevier, 2005, pp. 665-670.
[246] G. A. Miller, "WordNet: A Lexical Database for English," *COMMUNICATIONS OF THE ACM,* vol. 38, 1995.
[247] P. Domingos, "A few useful things to know about machine learning," *Commun. ACM,* vol. 55, pp. 78-87, 2012.
[248] I. Feinerer, "An introduction to text mining in R," in *R News*, ed, 2008, pp. 19-22.

[249] C. Elkan, "Magical thinking in data mining: lessons from CoIL challenge 2000," presented at the Proceedings of the seventh ACM SIGKDD international conference on Knowledge discovery and data mining, San Francisco, California, 2001.
[250] C. Phua, V. Lee, K. Smith, and R. Gayler, "A comprehensive survey of Data Mining-based Fraud Detection Research," *Artificial Intelligence Review,* // 2005.
[251] S.-Y. Huang, R.-H. Tsaih, and F. Yu, "Topological pattern discovery and feature extraction for fraudulent financial reporting," *Expert Systems with Applications,* vol. 41, pp. 4360-4372, 2014.
[252] J. Tang, S. Alelyani, and H. Liu, "Feature Selection for Classification: A Review," in *Data Classification*, ed: Chapman and Hall/CRC, 2014, pp. 37-64.
[253] K. Fukunaga, "Introduction to Statistical Pattern Recognition," Academic Press Professional, San Diego1990.
[254] R. Balakrishnan, X. Y. Qiu, and P. Srinivasan, "On the predictive ability of narrative disclosures in annual reports," *European Journal of Operational Research,* vol. 202, pp. 789-801, 5/1/ 2010.
[255] I. K. Fodor, "A survey of dimension reduction techniques," 2002.
[256] L. Parsons, E. Haque, and H. Liu, "Subspace Clustering for High Dimensional Data: A Review," *SIGKDD Explorations,* vol. 6, pp. 90-105, 2004.
[257] A. L. Blum and P. Langley, "Selection of relevant features and examples in machine learning," *Artificial Intelligence,* vol. 97, pp. 245-271, 1997/12/01 1997.
[258] Y. Yang and J. Pederson, "Feature selection in statistical learning of text categorization," in *Machine Learning Proceedings of the Fourteenth International Conference*, 1997, pp. 412-420.
[259] I. Guyon and A. Elisseeff, "An Introduction to Variable and Feature Selection," *Journal of Machine Learning Research,* vol. 3, pp. 1157-1182, 2003.
[260] G. Chandrashekar and F. Sahin, "A survey on feature selection methods," *Computers & Electrical Engineering,* vol. 40, pp. 16-28, 1// 2014.
[261] T. A. Abdallah and B. d. L. Iglesia, "Survey on Feature Selection," *arXiv preprint arXiv:1510.02892,* 2015.
[262] S. Deerwester, S. Dumais, G. Furnas, T. Landauer, and R. Harshman, "Indexing by latent semantic analysis," *Journal of the Association for Information Science and Technology,* vol. 41, pp. 391-407, 1990.
[263] T. Landauer, P. Foltz, and D. Laham, "Introduction to Latent Semantic Analysis," *Discourse Processes,* vol. 25, pp. 259-284, 1998.
[264] H. Ekbia, *Artifical Dreams: The Quest for Non-Biological Intelligence*. USA: Cambridge University Press, 2008.
[265] M.-W. Wang, J.-Y. Nie, and X.-Q. Zeng, "A latent semantic classification model," presented at the Proceedings of the 14th ACM international conference on Information and knowledge management, Bremen, Germany, 2005.
[266] M. Shafiei, S. Wnag, R. Zhang, M. Evangelos, B. Tang, J. Tougas*, et al.*, "Document Representation and Dimension Reduction for Text Clustering," in *Workshopr for Text Data Mining*, Turkey, 2007.
[267] D. Hand, H. Mannila, and P. Smyth, *Principles of Data Mining*: The MIT Press, 2001.
[268] I. T. Jolliffe, *Principle Component Analysis*. England, 1986.
[269] L. I. Smith, "A tutorial on Principal Components Analysis," ed, 2002.
[270] M. B. Kursa and W. R. Rudnicki, "Feature selection with the boruta package," *Journal of Statistical Software,* vol. 36, pp. 1-13, 2010.
[271] A. Engelhardt. (2010). *Feature selection: Using the caret package* https://www.r-bloggers.com/feature-selection-using-the-caret-package/.
[272] M. Breazu, D. Morariu, and R. Cretulescu, "Feature Selection in Document Classification," presented at the The fourth International Conference in Romania of Information Science and Information Literacy, 2013.
[273] A. G. K. Janecek, W. N. Gansterer, M. A. Demel, and G. F. Ecker, "On the Relationship Between Feature Selection and Classification Accuracy," in *JMLR: Workshop and Conference Proceedings*, 2008, pp. 90-105.
[274] S. Borgatti. (1997). *Multidimensional scaling. Retrieved from* http://www.analytictech.com/borgatti/mds.htm.
[275] N. Nilsson, "Introduction to Machine Learning," http://robotics.stanford.edu/people/nilsson/mlbook.html, 2015
[276] G. James, D. Witten, T. Hastie, and R. Tibshirani, *An Introduction to Statistical Learning with Applications in R*. New York: Spinger, 2013.
[277] Y. Abu-Mostafa, *Learning From Data*, 2012.
[278] A. Ng, *Machine Learning Yearning*, 2016.

[279] M. Dixon, D. Klabjan, and J. H. Bang, "Classification-based Financial Markets Prediction using Deep Neural Networks," 2016.
[280] C. Guestrin and E. Fox, "Machine Learning: Classification, https://www.coursera.org/learn/ml-classification," ed, 2016.
[281] S. B. Kotsiantis, "Supervised Machine Learning: A Review of Classification Techniques " *Informatica,* vol. 31, pp. 249-268, 2007.
[282] W. Liu, "Large-Scale Machine Learning for Classification and Search," 2012.
[283] H. Daume, "A Course in Machine Learning http://ciml.info/," 2016.
[284] H. Valpola, "Supervised vs. unsupervised learning http://users.ics.aalto.fi/harri/thesis/valpola_thesis/node34.html," ed, 2000.
[285] M. Ahmed, A. N. Mahmood, and M. R. Islam, "A survey of anomaly detection techniques in financial domain," *Future Generation Computer Systems,* vol. 55, pp. 278-288, 2016.
[286] A. Ng and M. Deisenroth, "Machine learning for a london housing price prediction mobile application," 2015.
[287] D. Sejdinovic, "Adapted from Statistical Data Mining and Machine Learning http://www.stats.ox.ac.uk/~sejdinov/teaching/sdmml15/materials/HT15_lecture12-nup.pdf," ed.
[288] M. Punniyamoorthy and P. Sridevi, "Identification of a standard AI based technique for credit risk analysis," *Benchmarking: An International Journal,* vol. 23, pp. 1381-1390, 2016.
[289] S. Janpuangtong and D. A. Shell, "Leveraging ontologies to improve model generalization automatically with online data sources," presented at the Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence, Austin, Texas, 2015.
[290] M. Martinez-Canales. (2014). *Ask a Data Scientist: The Bias vs. Variance Tradeoff http://insidebigdata.com/2014/10/22/ask-data-scientist-bias-vs-variance-tradeoff/*.
[291] R. Bellman, *Dynamic programming*. New Jersey: Princeton University Press, 1957.
[292] P. G. LeFloch and J.-M. Mercier, "An algorithm (CoDeFi) for overcoming the curse of dimensionality in mathematical finance," 2016.
[293] V. Spruyt. (2014). *The Curse of Dimensionality in classification http://www.visiondummy.com/2014/04/curse-dimensionality-affect-classification/*.
[294] D. Conway and J. M. White, *Machine Learning for email*: O Reilly Media Inc, 2012.
[295] C. Bishop, *Pattern Recognition and Machine Learning*. Singapore: Springer, 2006.
[296] M. Kuhn, "A Short Introduction to the caret Package," 2015.
[297] M. Kuhn, "Building Predictive Models in R Using the caret Package," *Journal of Statistical Software,* vol. 28, 2008.
[298] L. Breiman, J. Friedman, R. Olshen, and C. Stone, "Classification and Regression Trees," 1984.
[299] M. Keikha, N. Razavian, F. Oroumchian, and S. Hassan, "Document representation and quality of text: an analysis survey of text mining: clustering, classification, and retrieval.," *2nd ed. Spinger,* 2008.
[300] M. Kuhn and K. Johnson, *Applied Predictive Modeling*, 2013.
[301] T.-T. Wong, "Performance evaluation of classification algorithms by k-fold and leave-one-out cross validation," *Pattern Recogn.,* vol. 48, pp. 2839-2846, 2015.
[302] H. He and E. A. Garcia, "Learning from Imbalanced Data," *IEEE Transactions on Knowledge and Data Engineering,* vol. 21, 2009.
[303] I. Brown and C. Mues, "An experimental comparison of classification algorithms for imbalanced credit scoring data sets," *Expert Systems with Applications,* vol. 39, pp. 3446-3453, 2/15/ 2012.
[304] L. Obermann and S. Waack, "Demonstrating non-inferiority of easy interpretable methods for insolvency prediction," *Expert Syst. Appl.,* vol. 42, pp. 9117-9128, 2015.
[305] F. J. Valverde-Albacete and C. Peláez-Moreno, "100% Classification Accuracy Considered Harmful: The Normalized Information Transfer Factor Explains the Accuracy Paradox," *PLoS ONE,* vol. 9, p. e84217, 2014.
[306] J. Brownlee. (2014). *Classification Accuracy is Not Enough: More Performance Measures You Can Use http://machinelearningmastery.com/classification-accuracy-is-not-enough-more-performance-measures-you-can-use/*.
[307] P. Danenas and G. Garsva, "Credit risk evaluation modeling using evolutionary linear SVM classifiers and sliding window approach," *Procedia Computer Science,* vol. 9, pp. 1324-1333, 2012/01/01 2012.
[308] J. Sanz, A. Fernández, H. Bustince, and F. Herrera, "A genetic tuning to improve the performance of Fuzzy Rule-Based Classification Systems with Interval-Valued Fuzzy Sets: Degree of ignorance and lateral position," *International Journal of Approximate Reasoning,* vol. 52, pp. 751-766, 2011/09/01 2011.

[309] R. F. Lima and A. C. M. Pereira, "A fraud detection model based on feature selection and undersampling applied to Web payment systems," *IEEE-WIC-ACM International Conference on Web Intelligence and Intelligent Agent Technology,* 2015.
[310] S. Jarvis, Y. Bestgen, and S. Pepper, "Maximizing Classification Accuracy in Native Language Identification," in *Proceedings of the Eighth Workshop on Innovative Use of NLP for Building Educational Applications*, Atlanta, 2013, pp. 111-118.
[311] S. Fitzgerald, G. Mathews, C. Morris, and O. Zhulyn, "Using NLP techniques for file fragment classification," *Digital Investigation,* vol. 9, pp. S44-S49, 2012.
[312] J. Verostek, "Improving Model Performance http://www.johnverostek.com/wp-content/uploads/2014/06/Chapter-11.pdf," ed, 2014.
[313] E. Chen. (2011). *Choosing a Machine Learning Classifier http://blog.echen.me/2011/04/27/choosing-a-machine-learning-classifier/*.
[314] P. Sarlin, *Mapping Financial Stability*: Springer, 2014.
[315] H. Park, "An introduction to logistic regression: from basic concepts to interpretation with particular attention to nursing domain," *Journal of Korean Academic Nursing,* vol. 43, pp. 154-164, 2013.
[316] S. Sperandei, "Understanding logistic regression analysis," *Biochemia Medica,* vol. 24, pp. 12-18, 2014 2014.
[317] P. Allison, *Logistic Regression Using SAS Theory and Application*, 2012.
[318] Z. Zou, H. Peng, and L. Luo, "The Application of Random Forest in Finance," *Applied Mechanics and Materials,* vol. 740, pp. 947-951, 2015.
[319] C. Liu, Y. Chan, S. H. A. Kazmi, and H. Fu, "Financial Fraud Detection Model: Based on Random Forest " *International Journal of Economics and Finance,* vol. 7, 2015.
[320] D. Benyamin. (2012). *A Gentle Introduction to Random Forests, Ensembles, and Performance Metrics in a Commercial System https://citizennet.com/blog/2012/11/10/random-forests-ensembles-and-performance-metrics/*.
[321] E. Scornet, G. Biau, and J.-P. Vert, "Consistency of Random Forest," *The Annals of Statistics,* vol. 43, pp. 1716-1741, 2015.
[322] L. Breiman, "Random forests," *Machine Learning,* vol. 45, pp. 5-32, 2001.
[323] M. Fern, #225, ndez-Delgado, E. Cernadas, Sen, #233*, et al.*, "Do we need hundreds of classifiers to solve real world classification problems?," *J. Mach. Learn. Res.,* vol. 15, pp. 3133-3181, 2014.
[324] G. Louppe, "Understanding Random Forests From Theory to Practice," 2014.
[325] E. Kirkos, C. Spathis, and Y. Manolopoulos, "Data Mining techniques for the detection of fraudulent financial statements," *Expert Systems with Applications,* vol. 32, pp. 995-1003, 2007.
[326] I. Bose and J. Wang, "Data mining for detection of financial statement fraud in Chinese Companies.," in *Paper presented at the International Conference on Electronic Commerce, Administration, Society and Education*, Hong Kong, 2007.
[327] J. Han, M. Kamber, and J. Pei, *Data Mining Concepts and Techniques*: Morgan Kaufman, 2011.
[328] C. Yeh. (2015). *Support Vector Machines for classification http://efavdb.com/svm-classification/*.
[329] A. Natekin and A. Knoll, "Gradient boosting machines, a tutorial,," *Frontiers in neurorobotics,* vol. 7, 2013.
[330] M. Kumar and S. Upadhyay, "Predicting usefulness of online reviews using stochastic gradient boosting and randomized trees," in *Eleventh Australasian Data Mining Conference (AusDM13)*, Canberra, Australia, 2013, pp. 65-72.
[331] E. A. Freeman, G. G. Moisen, J. W. Coulston, and B. T. Wilson, "Random forests and stochastic gradient boosting for predicting tree canopy cover: comparing tuning processes and model performance," *Canadian Journal of Forest Research,* vol. 46, pp. 323-339, 2016/03/01 2015.
[332] G. Ridgeway, "Generalized Boosted Models: A guide to the gbm package. http://cran.r-project.org/web/packages/gbm/vignettes/gbm.pdf," 2007.
[333] O. Sutton, "Introduction to k Nearest Neighbour Classification and Condensed Nearest Neighbour Data Reduction," 2012.
[334] S. Jiang, G. Pang, M. Wu, and L. Kuang, "An improved K-nearest-neighbor algorithm for text categorization," *Expert Syst. Appl.,* vol. 39, pp. 1503-1509, 2012.
[335] J. Swathy and S. Surya, "Review on k -Nearest Neighbor Classification over Semantically Secure Encrypted Relational Data," *International Jpurnal of Computer Applications,* vol. 133, 2016.
[336] N. Bhatia and Vandana, "Survey of Nearest Neighbor Techniques," *International Journal of Computer Science and Information Security,,* vol. 8, 2010.

[337] A. Sabau, "Survey of Clustering based Financial Fraud Detection Research," 2012.
[338] H. Issa and M. Vasarhelyi, "Application of Anomaly Detection Techniques to Identify Fraudulent Refunds (August 16, 2011). Available at SSRN: https://ssrn.com/abstract=1910468 " 2011.
[339] S. Thiprungsri and M. Vasarhelyi, "Cluster Analysis for Anomaly Detection in Accounting Data: An Audit Approach," *The International Journal of Digital Accounting Research,* vol. 11, pp. 69-84, 2011.
[340] Q. Deng and G. Mei, "Combining self-organizing map and K-means clustering for detecting fraudulent financial statements," in *Granular Computing, 2009, GRC '09. IEEE International Conference on*, 2009, pp. 126-131.
[341] M. Albashrawi, "Detecting Financial Fraud Using Data Mining Techniques: A Decade Review from 2004 to 2015," *Journal of Data Scienc,* vol. 14, pp. 553-570, 2016.
[342] Deloitte, "Deloitte Annual report insights 2015 The reporting landscape," 2015.
[343] T. Copnell, "Guidance on the Strategic Report," *UK Audit Committee Institute*
[344] N. O. F. Elssied, O. Ibrahim, and A. H. Osman, "Enhancement of spam detection mechanism based on hybrid $$\varvec{k}$$ k -mean clustering and support vector machine," *Soft Computing,* vol. 19, pp. 3237-3248, 2015.
[345] P. Devi and R. K. Ranjan, "Classification with K-means Clustering and Decision Tree," *International Journal of Engineering Sciences and Research Technology,* pp. 775-779, 2014.
[346] E. M. Carneiro, L. A. V. Dias, A. M. d. Cunha, and L. F. S. Mialaret, "Cluster Analysis and Artificial Neural Networks: A Case Study in Credit Card Fraud Detection," in *Information Technology - New Generations (ITNG), 2015 12th International Conference on*, 2015, pp. 122-126.
[347] Y. Bengio and Y. LeCun, "Scaling Learning Algorithms towards AI," 2007.
[348] V. Shaw, "Financial scam takes place 'every 15 seconds' in UK," in *The Independent*, ed, 2016.
[349] H. Jones and J. Davey, "UK fraud agency charges three in Tesco accounting probe," in *Reuters*, ed, 2016.
[350] G. Smith. (2015) Toshiba just lost its CEO to a huge accounting scandal *Fortune*.
[351] A. Dyck, A. Morse, and L. Zingales, "How Pervasive is Corporate Fraud," 2013.
[352] F. Li, R. Lundholm and M. Minnis, “A Measure of Competition Based on 10-K Filings”, Chicago Booth Research Paper No. 11-30, 2012. Available at SSRN: htttps://ssrn.com/abstract=1908338 or http://dx.doi.org/10.2139/ssrn.1908338.
[353] V. Nagar, D. Nanda, and P. Wysocki, “Discretionary Disclosure and Stock-based Incentives”, Journal of Accounting and Economics 34: pp 283-309, 2003.
[354] S. Grossman and O.Hart, "Takeover Bids, The Free Rider Problem and the Theory of the Corporation", The Bell Journal of economics, Vol.11, No.1. pp 42-64, 1980.
[355] B.Bushee, I. Gow and D. Taylor, "Linguistic Complexity in Firm Disclosures: Obfuscation or Information?" Working Paper, January 2014.
[356] F. Li, “The Information Content of Forward-Looking Statements in Corporate Filings—A Naïve Bayesian Machine Learning Approach”, Journal of Accounting Research Vol. 48 No. 5, 2010.
[357] E. Athanasakou and K. Hussainey, “The perceived credibility of forward looking performance disclosures”, Accounting and Business Research, Vol 44, No 3, pp. 227-259, 2014.
[358] V. Muslu, S. Radhakrishnan, K. Subramanyam, and D. Lim, “Firm’s information environment and forward-looking disclosures in the MD&A” Working Paper, University of Texas at Dallas (2013).
[359] L. Field, M. Lowry and S. Shu, “Does Disclosure Deter or Trigger Litigation?”, Journal of Accounting & Economics, Available at SSRN: https://ssrn.com/abstract=604, 2003.
[360] K. Nelson and A. Pritchard, "Carrot or Stick? The Shift from Voluntary to Mandatory Disclosure of Risk Factors", Law & Economics Working Papers. Paper 104, 2014.
[361] J. Zimmerman, “Myth: External Financial Reporting Quality Has a First-Order Effect on Firm Value”, Accounting Horizons: December 2013, Vol. 27, No. 4, pp. 887-894, 2013.
[362] L. Holder and J. Cohen, “The Association between Disclosure, Distress, and Failure”, Journal of Business Ethics, vol. 75, No. 3, pp. 301-314, 2007.
[363] A. Agustini, "The Effect of Firm Size and Rate of Inflation on Cost of Capital: The Role of IFRS Adoption in the World", vol 219, pp 47-54, 2016.
[364] C.Koch, “The Relationship between Firm Level Corporate Governance and the Performance of Financial Analysts”, International Journal of Business and Social Science Vol. 6, No. 2; 2015.
[365] W. Shan, "Implications of Bias and Sentiment in the Financial Market", 2016 (PhD).
[366] A. Lindqvist, “What Drives Risk Disclosure Quality?- The Impact of the Financial Crisis”, 2016.

[367] S. Chen, Y. James Goo and Z. Shen, “A Hybrid Approach of Stepwise Regression, Logistic Regression, Support Vector Machine, and Decision Tree for Forecasting Fraudulent Financial Statements”, The Scientific World Journal, 2014.
[368] S. Abraham and P. Cox, “Analysing the determinants of narrative risk information in UK FTSE 100 annual reports”, British Accounting Review, vol. 39, pp. 227-248, 2007.
[369] H. Elzahar and K. Hussainey, "Determinants of narrative risk disclosures in UK interim reports", The Journal of Risk Finance, Vol. 13, pp.133 – 147, 2012.
[370] R. Lehavy, L. Feng and K. Merkley, “The Effect of Annual Report Readability on Analyst Following and the Properties of Their Earnings Forecasts”, The Accounting Review, Vol. 86, pp. 1087-1115, 2011.
[371] P. Linsley and P. Shrives, “Risk reporting: A study of risk disclosures in the annual reports of UK companies”, The British Accounting Review, vol 38, pp 387-404, 2006.
[372] M.Smith and R. Taffler,"The chairman’s statement - A content analysis of discretionary narrative disclosures", Accounting, Auditing & Accountability Journal, Vol. 13 pp.624-647, 2000.
[373] W. Beaver, M. McNichols and Z. Wang, “The Information Content of Earnings Announcements: New Insights on Intertemporal and Cross-Sectional Behavior”, Stanford University Graduate School of Business Research Paper No. 16-40; Available at SSRN: https://ssrn.com/abstract=2814387, 2016.
[374] H. You and X. Zhang, “Investor Under-reaction to Earnings Announcement and 10-K Report”, Barclays Global Investors, University of California, Berkeley, 2007.
[375] A. Kilgarriff et al. “The Sketch Engine: ten years on”, Lexicography (2014): 1–30, http://www.sketchengine.co.uk.
[376] A. Hardie, “Log Ratio – an informal introduction”, http://cass.lancs.ac.uk/?p=1133, 2014
[377] F. Wickelmaier, “An Introduction to MDS”, https://pdfs.semanticscholar.org/, 2003
[378] T.F. Cox, “Multidimensional Scaling”, Chapman and Hall, 2001.
[379] T. DAuria, “How to Build a Text Mining, Machine Learning Document Classification System in R”, https://www.youtube.com/watch?v=j1V2McKbkLo.
[380] M. Bernico, “Text Analytics - Latent Semantic Analysis”, https://www.youtube.com/watch?v=BJ0MnawUpaU&t=673s
[381] T Mikolov, I Sutskever, K Chen, GS Corrado, J Dean, “Distributed representations of words and phrases and their compositionality”, Advances in neural information processing systems, 3111-3119.
[382] A. Kumar, https://www.quora.com/What-is-the-definition-of-word-embedding-word-representation.
[383] https://deeplearning4j.org/word2vec.