INSTITUTO
UNIVERSITÁRIO
DE LISBOA

Decoding the numbers and language behind financial statement fraud

João de Brito Brás de Oliveira

Master's in Data Science

Supervisors:

Dr. Anabela Ribeiro Dias da Costa, Assistant Professor
ISCTE – Instituto Universitário de Lisboa

Dr. Diana Elisabeta Aldea Mendes, Associate Professor
ISCTE – Instituto Universitário de Lisboa

September, 2024

Department of Quantitative Methods for Management and Economics

Department of Information Science and Technology

Decoding the numbers and language behind financial statement fraud

João de Brito Brás de Oliveira

Master's in Data Science

Supervisors:
Dr. Anabela Ribeiro Dias da Costa, Assistant Professor
ISCTE – Instituto Universitário de Lisboa

Dr. Diana Elisabeta Aldea Mendes, Associate Professor
ISCTE – Instituto Universitário de Lisboa

September, 2024

Acknowledgements

Dado que nada se consegue sozinho tenho muita gente a quem agradecer a conclusão deste trabalho pelo apoio do qual beneficiei antes e ao longo do mesmo.

Desta forma, agradeço primeiramente, à Professora Anabela Ribeiro Dias da Costa e à Professora Diana Elisabeta Aldea Mendes por terem aceite orientar esta dissertação. Agradeço-lhes pelos ensinamentos ao longo do mestrado e da licenciatura, pela sua disponibilidade e ainda por me terem dado a oportunidade de seguir este tema por mais sinuoso que aparentasse ser ao início o caminho.

Logo em seguida, agradeço à Professora Maria do Rosário Veiga, que, apesar de não ter orientado esta dissertação, teve um impacto preponderante na mesma por ter estimulado em mim a curiosidade que a fez nascer.

Adicionalmente, gostaria de deixar um profundo agradecimento para todos aqueles que fazem do ISCTE o ISCTE e que mantêm a funcionar todas as suas instalações por me fazerem sentir tão bem-vindo.

Em seguida, queria também agradecer aos meus amigos Francesco, Luís e Ricardo pelos momentos de amizade e de partilha, por me ajudarem a crescer enquanto pessoa e ainda terem tido a paciência necessária para lidar com a minha ausência.

Não posso deixar de agradecer à minha família. Principalmente, à minha irmã Ana, à minha Tia Anabela, à minha avó Felicidade e aos meus avós Carlota, Renato e Roque.

O meu maior agradecimento vai para os meus pais que para além de me apoiarem ao longo de todas as etapas que me levaram aqui, sabendo que "*o sonho comanda a vida*", me ensinaram a sonhar e me deram todas as condições e ferramentas para fazê-lo.

Por último, gostaria de agradecer à minha namorada, a Madalena, por ter, mesmo de longe, acompanhado diariamente e de perto toda a extensão do meu mestrado e ter acreditado em mim nos momentos em que eu não dei esse benefício a mim mesmo. Sem ela não teria sido possível. Peço desculpa pelas dores de cabeça acrescidas que este período trouxe, mas que agora terminam.

Resumo

A fraude de reporte financeiro custa às empresas, a par da corrupção e da apropriação indevida de ativos, mais de 5 biliões de dólares americanos por ano. A deteção atempada desta infração desempenha um papel crucial nos danos sofridos. Por conseguinte, é essencial dispor de métodos automatizados capazes de identificar ocorrências com elevada probabilidade de fraude. Neste sentido, este estudo avaliou o potencial dos Modelos de Linguagem de Grande Escala (LLMs) como o BERT e o FinBERT, comparando o seu desempenho com modelos como a Regressão Logística e o XGBoost.

Para tal, analisou-se a secção "*Management's Discussion & Analysis*" de 1850 relatórios 10-K (1436 não fraudulentos e 414 fraudulentos), juntamente com rácios financeiros e variáveis contabilísticas de empresas, entre 1993 e 2014. Os modelos treinados utilizaram três tipos de variáveis: financeiras, textuais e uma combinação de ambas. A avaliação baseou-se em três métricas: AUC, NDCG@k e uma '*Captura'* baseada num valor limite, visto que, neste caso, as probabilidades de fraude podem ser mais informativas do que as classes preditas pelo modelo.

Os resultados sugerem que a última parte da secção MD&A capta informações mais relevantes do que a inicial. Além disso, a média das previsões dos modelos baseados na primeira e na última parte da secção aparenta não melhorar significativamente os resultados apesar de melhorar a captura. O FinBERT superou o BERT e obteve valores de AUC comparáveis aos modelos tradicionais que utilizam o *'text-embedding-3-large*' da *OpenAI*, obtendo também valores superiores de NDCG@k e de *'Captura'*.

Keywords: Fraud detection; Financial statements; SEC; Deep learning; Machine learning; LLM
JEL Classification: C63, M41.

Abstract

Financial statement fraud costs companies, in addition to corruption and asset misappropriation, over 5 trillion US dollars annually. The timely detection of this offense plays a crucial role in the damage suffered. Therefore, automated methods capable of identifying high-probability fraud occurrences are essential. Therefore, this study evaluates the potential of Large Language Models (LLMs) such as BERT and FinBERT by comparing their performance to that of well-established models like the Logistic Regression and the XGBoost.

To accomplished this, in our study, we went over the Management's Discussion & Analysis (MD&A) section of 1850 10-K reports (1436 non-fraud and 414 fraud), alongside financial ratios and raw accounting variables from companies which were known to have manipulated at least a single report in the past spanning from 1993 to 2014. Models were trained using three variable types: financial, text, and a combination of both. Evaluation was done using three metrics, AUC, NDCG@k and a threshold-based 'Capture', as to the specific problem, probabilities can be more informative than labels.

The results suggest that the last part of the MD&A section captures more relevant information than the beginning. Additionally, rank-averaging predictions from models based on the first and last parts of the section did not yield significant improvements despite the improved capture. FinBERT outperformed BERT and achieved AUC scores comparable to traditional models that leverage OpenAI's 'text-embedding-3-large' and surpass them in both NDCG@k and capture rates. Thus, FinBERT's domain-specific pretraining proved to be particularly advantageous in enhancing fraud detection performance.

Keywords: Fraud detection; Financial statements; SEC; Deep learning; Machine learning; LLM

JEL Classification: C63, M41.

Table of Contents

List of Figures

List of Tables

List of Abbreviations

AAER – Accounting and Auditing Enforcement Releases

AI – Artificial Intelligence

API – Application Programming Interface

AUC – Area Under the Curve

BERT – Bidirectional Encoder Representations from Transformers

CIK – Central Index Key

EDGAR – Electronic Data Gathering, Analysis, and Retrieval system

GVKEY – Global Company Key

IDS – Iscte Discovery Service

LLM – Large Language Model

LR – Logistic Regression

MD&A – Management's Discussion and Analysis

ML – Machine Learning

NaN – Not a Number

NDCG@k – Normalized Discounted Cumulative Gain at the position k

NLP – Natural Language Processing

ROC – Receiver Operating Characteristic

SEC – Securities and Exchange Commission

SOX – Sarbanes Oxley Act

SVM – Support Vector Machine

USA – United States of America

XGBoost – eXtreme Gradient Boosting

1. Introduction

1.1 Fraud overview

Fraud is a broad legal concept that permeates diverse domains, from Payments and Accounting to Insurance, Opinion, and Consumption. Among these, Accounting has garnered substantial research attention, ranking second only to Payments [1] and with good reason.

Accounting, reports, and financial statements have an essential role in the way the market works. They are vital instruments, conveying an organization's performance to stakeholders like investors, creditors, and regulators. These stakeholders rely on such information to make informed decisions. However, documents are susceptible to misrepresentation when mistakes are made or opportunities, rationalizations, and pressures to achieve favorable outcomes converge, resulting in the intentional dissemination of misleading information.

In the business landscape, corporate fraud deemed "*occupational fraud*" encompasses corruption, asset misappropriation, and financial statement fraud. These phenomena are estimated to cost companies around 5% of their annual revenues, amounting to over 5 trillion US dollars globally [2]. Nevertheless, the true potential losses are likely higher, as not all instances are detected or reported, and the full extent of the damage includes not just direct financial losses but also reputational harm and indirect costs, which are inherently difficult to quantify and often overlooked [2].

Financial statement fraud is, according to the Public Company Advisory Oversight Board (PCAOB) [3], a form of intentional misstatements or omissions of amounts or disclosures in financial statements designed to deceive financial statement users and possesses a branch of its own (Appendix A.1). The Association of Certified Fraud Examiners (ACFE) states that, although it corresponds to the least prevalent offense it is the one that produces the highest average damage ($766,000 per case) and the one where time before detection plays a more significant role [2]. Their findings also suggest that common perpetrators are often first offenders and frequently hold positions of power within organizations, benefiting from information asymmetry. As a result, external stakeholders often struggle to grasp the company's true financial situation until it is already too late, and the losses are practically irrecoverable.

This highlights the need for fraud detection tools that can identify patterns and expedite the detection process, reducing costs and improving efficiency by helping direct investigations [4].

1.2 Understanding Fraud Dynamics in the USA

In the United States of America (USA) the Securities and Exchange Commission (SEC) regulates securities markets and requires, to prevent misrepresentations and fraud, publicly traded companies to periodically disclose information about their business to be listed on major U.S. stock exchanges [43]. These compulsory companies' filings are then made available to the public through the Electronic Data Gathering, Analysis, and Retrieval system (EDGAR). One of them, Form 10-K, whose structure is shown on Figure 1.1, requires CFO and CEO certification, provides "*a detailed summary of a company's business, the risks it faces, and the operating and financial results for a fiscal year*" [44] and is the focus of this study.

	Item	Section Name
Part I	Item 1	Business
	Item 1A	Risk Factors
	Item 1B	Unresolved Staff Comments
	Item 2	Properties
	Item 3	Legal Proceedings
	Item 4	Mine Safety Disclosures
Part II	Item 5	Market
	Item 6	Consolidated Financial Data
	Item 7	Management's Discussion and Analysis
	Item 7A	Quantitative and Qualitative Disclosures about Market Risks
	Item 8	Financial Statements
	Item 9	Changes in and Disagreements With Accountants
	Item 9A	Controls and Procedures
	Item 9B	Other Information
Part III	Item 10	Directors, Executive Officers and Corporate Governance
	Item 11	Executive Compensation
	Item 12	Security Ownership of Certain Beneficial Owners
	Item 13	Certain Relationships and Related Transactions
	Item 14	Principal Accounting Fees and Services
Part IV	Item 15	Exhibits and Financial Statement Schedules Signatures

Figure 1.1: The structure of a 10-K report – Adapted from Loukas et al. [5]

These documents are later reviewed by the SEC to ensure compliance with regulations. When requirements are not met, investigations begin and if warranted, legal action is taken through a subset of SEC administrative proceedings and litigation releases that receive a designation of Accounting and Auditing Enforcement Release (AAER) which are made available on the regulator's website [6].

However, due to the scarcity and high cost of forensic accountants, investigations are often prolonged, resulting in a significant lag between the offenses and their public disclosure, lessening the agencies' potential impact.

1.3 Fraud Motivations

Uncovering the motivations behind fraudulent behavior has been seen to anticipate it and therefore, has been the goal of many studies and theories. These studies trace back to Edwin H. Sutherland´s concept of "*white-collar crime*," which he introduced in 1940 as a crime committed by a person of respectability and high social status in their occupation. Sutherland theorized that criminal behavior is learned through interactions with others where individuals acquire values, attitudes, techniques, and motives [7]. Donald Cressey [8, 9] by carrying out interviews with prisoners tried for fraud, expanded on this by identifying three motivational factors that contribute to fraud: opportunity, pressure, and rationalization, known as the "*Fraud Triangle*." This model has become a cornerstone for understanding and addressing fraud, influencing various accounting audit standards and becoming the baseline for new theories, which, over the years, continued to emerge, providing different or extending existing dimensions.

Over time, additional theories and models have been built on these foundations. Albrecht [10] developed the Fraud Scale, which quantified the impact of fraud and replaced "*rationalization*" with "*personal integrity*." Wolfe and Hermanson [11] further expanded the "*opportunity*" dimension to include "*capacity*" transforming the fraud triangle into a diamond. Kranacher [12] proposed the MICE model, which expanded the "pressure" dimension to include money, ideology, coercion, and ego.

1.4 Financial Statement Analysis for Fraud Detection

Financial statements are written records that convey the financial activities of a company. From them, there are three main ones: the balance sheet – which provides a snapshot of a company's assets, liabilities, and shareholders' equity at a specific time and date – and the income and cashflow statements – which provide an overview of revenues, expenses, net income, and earnings per share and outline where money is coming from, as well as how it is being spent, respectively, for a given period (annual or quarterly).

The goal of analyzing financial statements is to detect "*red flags*", facts that very seldom occur naturally. This remains a challenging task since "*... for every fraud risk indicator, there is a possible non-fraud explanation*." Still, according to Zack [13], Wells [14] and the Fraud Examiners Manual [15] three distinct approaches can be outlined to achieve it: vertical, horizontal and ratio based.

Vertical analysis corresponds to measuring a single account, or a group of accounts, as a percentage of some larger total, and it enables a comparison of companies of different sizes. Horizontal analysis/trend analysis compares data across multiple time periods, being helpful in detecting changes over time. Finally, ratio analysis uses liquidity, activity, leverage and profitability ratios.

A company can have an anomaly for legitimate reasons, but simultaneous anomalies increase the likelihood of fraud, although horizontal and vertical analysis are by their own limited, using multiple indicators raises the likelihood of being able to reduce false positives.

1.5 Machine learning & LLMs

Artificial Intelligence (AI) techniques are a great means for capturing relationships between variables and have experienced rapid development, presenting numerous opportunities to the accounting and finance industry [46].

Machine Learning (ML) models offer computational power and flexibility to support analyses. A big field within ML, classification, is divided into binary and multiclass problems depending on whether only two or more classes exist for the target variable, respectively, and involves assigning an unknown observation to one of the predefined categories based on known data.

Large Language Models (LLMs) are AI algorithms based on the transformer architecture that have observed particularly accelerated activity and progress in recent times. These models, by leveraging Natural Language Processing (NLP) techniques and receiving pre-training through self-supervised learning on vast datasets, can outperform traditional algorithms in a wide range of tasks. Although their most significant application is text generation, they can be utilized in numerous downstream tasks via fine-tuning, such as sentiment analysis, summarization, and even some forms of reasoning [16].

1.6 Research Aim and Objectives

This dissertation aims to deepen the existing knowledge about mixed financial statement fraud detection models that employ both financial variables and text information and provide insights into the potential application of Large Language Models in identifying possible instances of fraud and misstatements. This is in the hope of improving resource allocation by helping better direct investigations, thereby helping to shield stakeholders who may make decisions from

misleading reports. Ultimately, this research aspires to help empower regulators with knowledge of automated tools that can help fight financial crime effectively.

The main objectives of this research are interlinked with the following research questions:

RQ1: "*How can different ways of dealing with Large Language Models max token input affect classification performance in financial statement fraud detection problems?*"

RQ2: "*How do Large Language Models fare against benchmark models*?"

RQ3: "*Can Large Language Model assist stakeholders in signalling textual indicators/"red-flags" within the Management´s Discussion and Analysis section?*"

1.7 Dissertation Organization

This dissertation is organized into five main chapters: Chapter 2, "Literature Review," surveys existing research to identify key contributions and knowledge gaps. Chapter 3, "Methodology," details the methods used for labelling 10-K reports as genuine or fraudulent, as well as data collection, selection, and preprocessing. Chapter 4, "Results and Discussion," presents and interprets the study's findings by addressing the research questions while also discussing limitations. Finally, Chapter 5, "Conclusion and Future Work," summarizes key insights, and suggests directions for future research. Then, follow "References," and "Appendix" with supplementary materials.

2. Literature Review

To ensure rigor, the review followed an approach adapted from Kitchenham and Charters [17]. As such, the process began by defining clear research objectives and formulating a strategy with targeted search queries, databases, and inclusion/exclusion criteria to ensure high-quality study selection. Studies were sourced from Scopus, Google Scholar, and the Iscte Discovery Service (IDS). Relevant data was extracted, and the quality of the studies was assessed. Finally, the data was synthesized to produce results aligned with the research objectives.

2.1 Research Questions

The following outlined questions guided the literature review research:

RQ1: "*What datasets and features have been most commonly used in this field*?" – To identify prevalent data sources and types of analyses used in the research.

RQ2: "*What financial reporting fraud detection models have been developed*?" – To explore benchmark models and methods, highlighting those with the best results.

RQ3: "*How did researchers/previous studies handle/address the class imbalance problem which is specifically common in fraud detection modeling attempts?*" – To identify effective strategies used to tackle common challenges in past studies.

2.2 Research Strategy

The research strategy focused on identifying academic articles on financial reporting fraud, particularly those utilizing automated tools like machine learning for detection and prevention, that followed certain criteria present on Table 2.1. Then, the search phase began by querying the Scopus database with the combinations of keywords such as "*fraud*", "*machine learning*", "*large language models*" and "*financial statement fraud*" using Boolean logic (AND, OR) to refine results and target relevant studies. Then, to ensure broader coverage, Google Scholar and IDS were also queried with similar terms (as can be seen from the queries utilized to each source on Table 2.2). Additionally, Connected Papers was also used to visualize relationships between papers, aiding in the discovery of highly relevant studies by displaying networks of important papers

Table 2.1: Inclusion criteria (IC) and exclusion criteria (EC) defined.

Decision	**Criteria**
Include	IC1: Full research articles published in academic journals. IC2: Research articles that use automated tools for the detection and prevention of financial statement fraud.
Exclude	EC1 Articles that do not focus on financial reporting fraud, but rather on the prevention of other financial frauds. EC2: Conceptual and review articles. EC3: Conference papers/lecture notes, etc. EC4: Studies that cannot be accessed free of charge. EC5: Studies that are not available in their entirety.

Table 2.2: Sources and queries used, and number of results found before and after applying the inclusion and exclusion criteria (Table 2.1).

Source	**Search query used (n = #total results/#total articles) (Queries)**
Scopus	TITLE-ABS-KEY (("fraud"AND"machine learning")) n = 3.706/682. TITLE-ABS-KEY (("financial statement fraud")) n = 350/75. TITLE-ABS-KEY ((“financial statement fraud"AND"machine learning")) n = 51/11. TITLE-ABS-KEY (("financial statement fraud" AND "deep learning")) n = 18/3. TITLE-ABS-KEY (("fraud" AND "large language models")) n = 0/0.
Google Scholar	“Accounting statement fraud” “deep learning” n = 3/0. “Financial statement fraud” “machine learning” n = 2.570/234.
IDS	TITLE-ABS-KEY ((“financial statement fraud"AND"machine learning")) n = 43/8. TITLE-ABS-KEY (("fraud" AND "large language models")) n = 3/0.

2.3 Literature Results & Discussion

This section summarizes the main conclusions obtained resulting from the research carried out, for each research question posed previously:

In what follows we summarize the findings related with the second research question – RQ1: “*What datasets and features have been most commonly used in this field*?” (Appendix A.2)

Financial statement fraud detection is a worldwide research topic in many countries. Nevertheless, past research has emphasized on the USA and China given that these are more “developed” markets where data can be obtained with more ease.

Historically, research in this field has approached financial statement fraud detection as a binary supervised classification problem. While slight variation has occurred on the types of features used, these have predominantly been financial [20, 21, 22, 23, 24, 25, 26, 27, 30, 31, 34, 35, 36, 37]. Still, distinct research avenues were explored other than only refining accrual

models, such as incorporating text analysis [18, 19, 24, 28, 29, 33] and other non-financial variables to enhance detection methods. Given there is no standardized dataset or evaluation framework for this problem, which makes it difficult to directly compare the results of different methods and input features across existing studies [57].

Concerning the sources of the data used, as can be observed from Appendix A.2, the target label has predominantly come from market regulatory bodies – SEC AAER [18, 19, 20, 22, 24, 25, 26, 28, 29, 32, 33, 35, 36, 37] in the US and the China Securities Regulatory Commission's (CSRC) [21] or the China Stock Market and Accounting Research (CSMAR) [34] in China. Meanwhile, the explanatory variables have typically been sourced the SEC's publicly accessible EDGAR system [18, 19, 24, 26, 29, 32, 33] and from proprietary databases, such as, specifically, *Compustat*, [18, 20, 22, 23, 24, 25, 27, 30, 35, 36, 37], and *BoardEx* [20].

In studies where financial features were used, they were taken from the financial statements previously stated in section 1.3 and have most appeared in the form of financial ratios, although some studies have experimented with raw accounting variables directly, achieving arguably greater results. Nevertheless, theoretically rooted studies have also been found to yield better results than simply relying on an extensive list of financial data items [22].

Based on the hypothesis that, as textual information is not subject to the same degree of regulation as its financial counterpart and could be conducive to the detection of fraud [32] leveraging linguistic and textual information became a significant focus. Stemming from the fact that the MD&A section of 10-K reports offers investors the possibility of reviewing the performance of the company as well as its future potential from the perspective of management, it has been heavily studied [18, 19, 24, 26, 29, 32, 33]. Additional credence to this was provided by the fact that the regulator themselves, the SEC, admitted to attempting to develop a program that could analyze the MD&A section to decipher the "word shell game", stating that companies try to "deflect attention from a core problem by talking a lot more about a benign" as well as "underreporting important risks" [40].

Furthermore, linguistic approaches focused on emotion and sentiment analysis through the applications of lexicons (e.g., The Linguistic Inquiry and Word Count (LIWC) (Pennebaker et al., [39]), the Multi Perspective Question Answering subjectivity lexicon (Wilson et al., [40]) or the 10-K report, domain-specific, Loughran and McDonald dictionary [42]) where words were pre-scored across certain categories (positivity, negativity, uncertainty, etc.).

Whereas textual approaches addressed classification or topic modelling problems and relied on extractions of informative word representations through several distinct techniques, like

Bag-of-Words (BOW), Term Frequency–Inverse Document Frequency (TF-IDF) or more recently, word embeddings. Major drawbacks for TF-IDF and BOW, however, are that these, opposite to word embeddings which capture the semantic relationships between words, do not capture much beyond word frequency or importance (rarity across a set of documents) in documents [24].

Non-financial studies focused on establishing relationships between the labels and the key roles and positions both within and external to an organization, including the auditors, board of directors, CEO, CFO, and other executives [20, 21].

Finally, some authors adopted mixed approaches by incrementally combining diverse data sources to assess whether performance improvements would follow a similar incremental pattern. For instance, Purda [29] found that their model ("*probability of truth*") and Dechow´s F-score [35] measures complemented one another, helping reduce the number of false positives. Then, Hakej [26] concluded that combining financial and linguistic features could improve the detection of financial statement fraud. Later, Craja [24] evaluated the combination of information from financial ratios and text from the MD&A section of 10-K reports and arrived at a similar conclusion. Schneider [20], using financial data combined with non-financial data from *BoardEx* also concluded that combining these variables was beneficial.

In what follows we summarize the findings related with the second research question - RQ2: "*What financial reporting fraud detection models have been developed*?" (Appendix A.3)

Fraud detection methodologies have undergone significant evolution over time. Foundational fraud detection efforts were heavily influenced by financial distress studies, as both domains share a focus on analyzing financial data to identify risks and anomalies. Altman's Z-score model [41] was crucial in shaping early fraud detection methods. Discriminant analysis with pair matching is used to combine traditional ratio analysis with advanced statistical techniques focusing on a company's financial data from a single year to predict bankruptcy, laying the groundwork for more advanced models designed to detect financial misstatements, earnings manipulation, and fraud.

Beneish's M-Score [37] expanded upon this foundation by incorporating horizontal analysis, analyzing data over two years to identify trends in financial misreporting. This temporal component introduced by the M-Score allowed for a more dynamic view of financial data, setting the stage for the application of more advanced statistical and machine learning models.

Traditional ML Methods

Logistic regression (LR) and decision trees were among the first machine learning algorithms to be widely adopted in fraud detection due to their robustness in handling binary classification problems. LR became a staple in fraud detection research for its simplicity, interpretability, and effectiveness in identifying fraudulent behavior in financial data. Cecchini [36] and Dechow [35] demonstrated the powerful utility of LR and Support Vector Machine (SVM) models in detecting fraud, establishing them as benchmarks for subsequent studies.

While these traditional ML models provided strong baselines, more advanced methods soon began to outperform them. Bao [22] introduced ensemble learning approaches that significantly improved upon the predictive power of LR and SVM, showcasing the potential for combining multiple models to enhance fraud detection accuracy.

Gradient-boosted algorithms, particularly XGBoost (an implementation of gradient-boosted regression trees), have emerged as a highly effective tool in fraud detection. Bertomeu [23] confirmed that tree-based models, such as XGBoost, offer a significant performance advantage over simpler algorithms. Despite being more computationally intensive, XGBoost's ability to capture complex patterns in the data makes it an excellent choice for detecting fraud in large financial datasets. Xu [21] further validated the superiority of tree-based models, with random forests (RF) and XGBoost outperforming other algorithms in terms of predictive accuracy.

Deep Learning and Large Language Models (LLMs)

Ravinsakar [34] pioneered the use of deep learning (DL) models in fraud detection. Although deep learning models have proven capable of capturing intricate patterns in complex datasets, their adoption has been slower due to higher computational costs and the need for large amounts of labeled data.

Later work by Craja [24] through a hierarchical attention network (HAN) embodied two different attention mechanisms at the word and sentence level, which allowed content to be differentiated in terms of its importance in the process of constructing the document representation. Nevertheless, XGBoost provided competitive performance.

A recent study by Sivasubramanian and Skillicorn [19] compared a range of deep learning models and large language models (LLMs) with traditional algorithms like LR and decision trees. The study concluded that while transformers and DL models show promise in capturing semantic and contextual information, traditional models, particularly XGBoost, still offer strong performance when considering the performance/computational efficiency trade-off.

As can be seen from Appendix A.3, which outlines the types of models used in previous studies, present in page 49, although LLMs have shown promise in other domains, to the best of our knowledge, only two studies have attempted to use LLMs, Sivasubramanian and Skillicorn [19] and Bhattacharya [18].

On the one hand, Sivasubramanian and Skillicorn [19] compared LLMs to the performance of the traditional ones and concluded that although transformers can achieve increased performance, this increase is marginal to that the XGBoost with embeddings and for that it does not compensate for the incurred additional complexity.

On the other hand, Bhattacharya [18], through a different approach inspired by Sun et al. (2019) on fine-tuning Bidirectional Encoder Representations from Transformers (BERT) achieved contrasting results. Bhattacharya positioned BERT as a strong alternative, outperforming Brown's [25] LDA and Bao's [22] RUSBoost[1] models by 15% and 12%, respectively. Bhattacharya argued that the initial tokens in MD&A reports typically contain introductory comments, while the final tokens often include the company's future vision and next steps—fundamentally different but equally important information. To capitalize on this, two BERT models were fine-tuned on the first and last 512 tokens of the MD&A section and then combined by averaging the ranks of their predictions.

As such, we can understand that although LLMs are generating increasingly genuine recent research efforts no current proper consensus exists on their potential application in financial statement fraud detection, highlighting a significant gap in the field. Similarly, it can be said that both LR and XGBoost remain highly relevant in modern fraud detection.

[1] *RusBOOST* – is an algorithm used to efficiently alleviate class imbalance problems by combining data sampling and boosting. It employs random undersampling (RUS), by randomly removing examples from the majority class [68].

In what follows we summarize the findings related with the second research question - RQ3: "*How did researchers/previous studies handle/address the class imbalance problem which is specifically common in fraud detection modeling attempts*?" (Appendix A.4)

Fraud studies always had to deal with the imbalance of the target variable as it would directly impact on the way that ML models would work, possibly resulting in a worse performance than could otherwise be achieved. Moreover, the choice of scoring methods and evaluation metrics is crucial in model construction, as they play a key role in addressing and mitigating these imbalances.

Appendix A.4, which outlines the balancing and evaluation approaches used in previous studies, present in page 50, reveals the use of diverse strategies. Some studies did not specify any balancing technique [18, 20, 25, 29, 30, 33, 34, 35, 36, 37], while matched-pair sampling was a popular method among those that did [26, 28, 31, 32]. This approach pairs cases and controls with similar characteristics but different outcomes (fraud vs. non-fraud). Random undersampling, often via RUSBoost [22, 23, 24], was another common technique, though it risks losing valuable information. Algorithm-level adjustments, like *class_weight* in DL and *scale_pos_weight* in ML models [19], were also used to enhance balance.

As for evaluation metrics, these have aimed at minimizing Type I and Type II errors, which in fraud detection mean falsely alleging fraud and fraud going undetected. Although accuracy was frequently reported [19, 20, 24, 25, 26, 27, 28, 31, 32, 34, 36], it can be misleading in imbalanced datasets. Metrics like F1-Score, Precision, and Recall offered better insight into the model's performance on the minority class. Area Under the Curve (AUC) was widely used for its ability to evaluate models without assuming equal error costs, which is essential in fraud detection.

More complex metrics, such as Normalized Discounted Cumulative Gain at the position k[2] (NDCG@k) [18, 21, 22, 25], were also employed to treat fraud detection as a ranking task, prioritizing the most suspicious cases given limited investigative resources.

[2] NDCG@k – Normalized Discounted Cumulative Gain at position k is a ranking quality metric which can take values from 0 to 1, where 1 indicates a match with the ideal order, and lower values represent a lower quality of ranking. It attempts to compare the economic significance of the predictions by comparing the number of fraudulent firms that could be captured by investigating the same number of firms [59].

3. Methodology

The project conducted followed the Cross Industry Standard Process for Data Mining (CRISP-DM) methodology, a standard in data science. Therefore, this chapter comprises four sections named after its four first steps: The first section (3.1) exhibits business understanding. The second section (3.2) corresponds to the data understanding. Then section 3.3 focuses on data preparation. Finally, section 3.4 is concerned with the evaluation metrics, the ML models used for the supervised binary classification problem at-hand as well as their fine-tuning. The Evaluation phase will be thoroughly analyzed in the next chapter, "Results and Discussion".

3.1 Business Understanding

As seen before, financial statement fraud detection is vital in finance and governance. This project aims to identify the best machine learning model for detecting it, analyze each model's strengths and weaknesses, and provide insights to improve detection strategies.

To ensure project success, models will be benchmarked against others - known to have had previous good performances - and results must be interpretable to provide insights to non-technical stakeholders. Success criteria should account for the difference between Type I (falsely alleging fraud) and Type II (missing fraud) error costs, as well as for the unequal proportion of fraudulent versus non-fraudulent statements, which pose challenges for classification models.

3.2 Data Understanding

3.2.1) Data Collection

3.2.1.1) Data sources

The data used for this research was obtained from distinct sources:

(1) Target Variable: Taking values of 0 (for no fraud instances) or 1(for fraud instances), was attained from a dual-source approach:

Firstly, through the application of several Google Dorks search queries (*filetype:pdf site:sec.gov "ACCOUNTING AND AUDITING ENFORCEMENT" "Material Weaknesses" "10-K*"; filetype:pdf site:sec.gov "*ACCOUNTING AND AUDITING ENFORCEMENT" "Material Weaknesses" "10-K" -LLP -CPA*; *filetype:pdf site:sec.gov "ACCOUNTING AND AUDITING ENFORCEMENT" "Material Weaknesses" "10-K" "fraud" -LLP -CPA; filetype:pdf site:sec.gov "ACCOUNTING AND AUDITING ENFORCEMENT" "fraud" "intentional*

misstatements" "10-K" -LLP -CPA) a collection of Accounting and Auditing Enforcements (AAER) issued by the U.S. Securities and Exchange Commission (SEC) which are publicly available on its website [47] was found and selected. However, as these were deemed insufficient to proceed with the analysis the dataset from Dechow, Ge, Larson and Sloan (2011)/USCMarshall [48] available for acquisition through the designated platform was also used.

(2) Textual Data: was obtained via EDGAR-CORPUS, an open-source *Hugging Face* (HF) dataset collected by Loukas et al. [5] which contains the text of the annual 10-K reports of over 38.000 public companies from the period between 1993 and 2020 from SEC EDGAR filings. Access to this dataset can be obtained via HF [49].

(3) Financial Variables: were retrieved from previous literature, specifically from Bao [22] "*data_FraudDetection_JAR2020.csv*" file present on the author´s GitHub [50].

The selected sources were specifically chosen for their degree of trust and high volume of information. As previously seen, SEC's AAER´s have been used as source for the target variable on a lot of previous literature, being regarded as a provider of robust evidence of manipulation, as the likelihood of incorrectly alleged frauds is extremely low, although there have certainly been missed cases of fraud. Edgar-Corpus is one of the most extensive text corpuses available and Bao [22] research was peer-reviewed and published in the *Journal of Accounting Research*, one of the most notorious journals in the areas, giving credence to the dataset which he derived from financial statement data, originally taken from *Compustat*.

3.2.1.2) Data Quality/Consistency Issues

Financial

Attempts made at creating our own data set were, however, unsuccessful as obtaining reliable current and historical financial data for a wide range of companies was not possible. Although at first appearing a difficult but accomplishable feat as there are on the market a couple of Application Programming Interfaces (API)´s (*Alpha Vantage* [51], *Tradefeeds* [52] and *Financial Modeling Prep* [53]) that could, in theory, achieve that data. After subscribing to said API´s, on paid plans, the historical data provided was lackluster as it did not match the advertised period, and there were consistency/quality issues.

API´s usually provide the company's financial statements in a quarter or annual fashion and on an "as reported" or API-structured basis. On the one hand, data retrieved on an "as reported" basis is difficult to combine for many companies and reporting years as very striking differences in formatting are noticeable. On the other hand, data retrieved on an API -structured basis has

data that is incorrect. This was observed after putting in place general accounting conditions (observable in Table 3.1) and then looking up the reports filed with the SEC on EDGAR´s website from which the API´s were supposedly extracting the data.

Table 3.1: General conditions used to check data consistency and Apple Inc´s example.

Condition name/Formula	
Assets difference	Total Assets – Current Assets – Non Current Assets
Liabilities difference	Total Liabilities – Current Liabilities – Non Current Liabilities
Golden rule	Total Assets – Total Equity – Total Liabilities
Net income difference	NetIncome_x – NetIncome_y

Example: Apple Inc.´s 10-K Consolidated Balance Sheets over the years

Fiscal Year	Assests difference	Liabilities difference	Golden rule	Net income difference
2015	-1.340000e+08	0	0	0
2009	-1.640000e+09	0	0	0
2008	1.283000e+09	0	0	0
2005	-3.500000e+07	0	0	0
2003	0	0	0	1000000
1997	0	0	0	-281000000
1994	7.0600000e+05	0	0	0
1993	5.700000e+04	0	0	0

CONSOLIDATED BALANCE SHEETS
(In millions, except number of shares which are reflected in thousands and par value)

ASSETS:	September 26, 2015	September 27, 2014
Current assets:		
Cash and cash equivalents	$ 21,120	$ 13,844
Short-term marketable securities	20,481	11,233
Accounts receivable, less allowances of $82 and $86, respectively	16,849	17,460
Inventories	2,349	2,111
Deferred tax assets	5,546	4,318
Vendor non-trade receivables	13,494	9,759
Other current assets	9,539	9,806
Total current assets	89,378	68,531
Long-term marketable securities	164,065	130,162
Property, plant and equipment, net	22,471	20,624
Goodwill	5,116	4,616
Acquired intangible assets, net	3,893	4,142
Other assets	5,556	3,764
Total assets	$ 290,479	$ 231,839

Figure 3.1: Consolidated Balance Sheets off Apple Inc.´s 2015 10-K Form.

Table 3.2: API retrieved data for Apple Inc.´s 2015 10-K Form, in millions.

Calendar Year	Total Current Assets	Total Non-current Assets	Total Assets
2015	89.378	200.967	290.479

Figure 3.1 and Table 3.2 attempt to show the differences between the supposed source data – Apple Inc.´s 2015 Form 10-K Consolidated Balance Sheets – and the data retrieved from the API. As can be seen from comparing the two it is possible to understand that both the values

for Total Current Assets (89.378) and Total Assets (290.479) are correct, as they are the same as the ones on the Figure, however, Total Non-current Assets isn´t, showing a difference equal to 134 million (290.479-89.378=201.101, 201.101-200.967=134). Which is in accordance with Table 3.1 which for fiscal year 2015 reports an "*Assets difference*" equal to that value. It was also found that this difference happened in different fashions across different accounts over the years. In Apple Inc´s case there seemed to only have happened issues with the net income and assets, however, distinct instances took place for different companies.

These issues gave rise to a lack of trust in the API´s results and made it necessary to use Bao [22] dataset – if financial information was to be incorporated in hopes of enriching the analysis.

Textual

EdgarCorpus, the text source for our study, was found to also have some problems regarding its data, specifically for section 7 which was used. In some cases, it showed rows as NaN for 10-K forms where it was stated that the text had been "*incorporated by reference*", which could be deemed as misleading or inaccurate. This, however, was not seen as a big deal as in such occurrences, the text is not present on the form itself and no "*error*" had taken place.

In other cases, although the original forms available at the SEC website seem to contain the MD&A section it appeared as NaN on the dataset which was clearly not correct. These two types of circumstances amounted to a total of 153 instances from 1993 to 2014.

Although not all the remaining samples were confirmed to be correct, a random sample of a tenth of the reports was compared to their original counterpart to check for inconsistencies. From the checked sample no material inconsistencies were found, and therefore, *EdgarCorpus* was used as the text data source.

3.2.1.3) Excluded variables

Although some variables could have been interesting to study as they have been found significant in prior research, they had to be excluded from consideration because they are not publicly available, namely:

- Exploring the organizations ownership structure (CEO, CFO, board of directors and other executives' committees under the board of directors) as well as its corporate governance.
- Exploring data related to the relations with auditors.
- Extending the financial and textual data to include all data available for the initially identified firms.

3.3 Data Preparation

3.3.1) Data Integration Process

The data collection started off with the selection of misstatement firm-years. A total of 1.214 firm-years were gathered from the Dechow, Ge, Larson and Sloan [48] dataset, specifically from the annual tab as only companies that had misstatements occur on annual reporting periods were of interest. A Microsoft Excel file with the list of companies that were selected was created along with their Central Index Key (CIK), their ticker symbol, their Global Company Key (GVKEY), and the year/s of the reports that each of them had misstated.

Then, the Edgar-Corpus dataset was fully downloaded and crossed with the previous Excel through the '*CIK*' column. As such, the data within Edgar-Corpus was filtered so that it contained only values of CIK which were present on the Excel so that, our data consisted of all the reports from companies which were known to have manipulated at least a single report in the past.

Then, a column named '*Fraud*' was created so that rows corresponding to years present in the Excel were labeled as 1 (Fraud has occurred) and the remaining as 0 (Fraud has not occurred). After this step, after an outlier analysis present within Appendix B, correspondent to step 4 of Table 3.3, missing values and lower bound outliers (identified using Tukey´s fences – Q1-1.5*IQR, as we were not only concerned with looking for extreme outliers) were dropped. This action resulted in an approximate 28% decrease in observations.

After this, using the '*GVKEY*' column, the target and textual information was crossed with the financial variables from Bao [22]. As a result, approximately 42% of instances were lost. After this merge, operation rows which had missing values were also discarded from the dataset.

As such the final dataset consisted of a total of 1.850 rows where each represented a unique filing (company and year combination). From the total rows, 1.436 and 414 were non-fraud and fraud filings, respectively. Table 3.3 immediately below provides a summary of the steps taken specifying the data sources and the absolute and relative change in the number of firm-years resulting from each decision.

Lastly, a random 65/15/20 train, validation and test stratified sample split was performed to first deal with the data imbalance by keeping a constant fraud-to-nonfraud ratio, maintaining a predominance of the positive class in detriment of negative one in a ratio of approximately 3.5:1 in all three sets (a train set with 933 non-fraud instances and 269 fraud instances whereas the validation and test sets remained with 216-62 and 287-83 fraud/non-fraud samples, respectively).

Table 3.3: Summary table of data integration process followed in this study.

Step	Process Followed	Data Source(s)	# firm-years
1	Identify 1.214 firm-year observations subject to SEC enforcement actions for alleged financial misreporting and violation of GAAP - annual basis	AAER & USC Marshall dataset/Dechow (2011)	1.214
2	Collect the textual content from 10-K report annual reports via HF	EDGARCORPUS	220.377
3	Cross the enforcement actions information with the textual data from EDGARCORPUS based on the 'CIK' columns	AAER & USC Marshall dataset Dechow (2011) & EDGARCORPUS	3.971
4	Drop all rows with missing values within the MD&A column	AAER & USC Marshall dataset Dechow (2011) & EDGARCORPUS	3.818
5	Drop all lower outliers found using Tukey's fences (1.5 IQR) on logarithm of the number of characters within 'section_7'	AAER & USC Marshall dataset Dechow (2011) & EDGARCORPUS	3.280
6	Cross the SEC enforcement actions and EDGARCORPUS data with Bao´s (2020) dataset using the 'gvkey´ column	AAER & USC Marshall dataset Dechow (2011) & EDGARCORPUS & Bao (2020)	1.905
7	Drop rows with missing values on the newly added financial data columns	AAER & USC Marshall dataset Dechow (2011) & EDGARCORPUS & Bao (2020)	1.850
8	Create train (0.65), validation (0.15) and test (0.2) using train_test_split	AAER & USC Marshall dataset Dechow (2011) & EDGARCORPUS & Bao (2020)	1.202 278 370

3.3.2) Data Cleaning and Transformation

3.3.2.1) Data Cleaning

As neither missing nor duplicate values existed after the data integration process described earlier, the only data cleaning step that took place was applied to '*section_7*' so that a fully processed and token-ready version was available for later transformations.

To achieve this, the text was first converted to lowercase for consistency. Secondly, portions of the text which are based on newline characters (\r, \n). were also removed to separate irrelevant headers. Then patterns found using regular expressions from previously performed exploratory analysis, like HTML tags and non-word characters were also removed. Finally, characters are filtered so that only alphanumeric characters and spaces are retained, removing any remaining special characters or punctuation.as well as unnecessary leading or trailing spaces.

Although it is commonplace within NLP to remove stopwords and numbers it has also been argued to be detrimental [19]. In our study we tested both approaches.

3.3.2.2) Data Transformation

As the financial data was taken from Bao [22] it did not need any further transformations other than normalization which was achieved via *StandardScaler*[3]. By contrast, to apply models to the unstructured text data its transformation into a structured representation was necessary. So, text was first converted to lower case and then tokenization was applied to break the text into tokens, necessary to further converting the text into numerical vectors. Embedding words as numeric vectors was achieved via *Open AI´*s '*text-embedding-3-large*'[4]. This model takes a maximum of 8.191 tokens as input and possesses a size of 3.072 dimensions. This was our embedding of choice as it corresponds to one of the newest and most performant embedding models that are currently available, and it was the one that had achieved the best results within the Massive Text Embedding Benchmark (64.6%) [62].

Another reason to choose this embedding model is the fact that it had been trained with a technique (Matryoshka Representation Learning) that allows for it to be shortened to a size of 256 (by removing numbers from the end of the sequence) while maintaining a sound level of performance [63].

[3] *StandardScaler* – This estimator standardizes features by removing the mean and scaling to unit variance [69].

[4] *text-embedding-3-large* – OpenAI's text embeddings measure the relatedness of text strings and are commonly used for a wide range of NLP tasks, including Classification [63].

LLMs used the BERTForSequenceClassification model and as such data had to be structured into the desired input sequences of up to 512 tokens. For this, several preprocessing steps were required to prepare the data. First, the models' native tokenizers broke down raw text into tokens, which were then encoded into numeric IDs. Meanwhile, to manage varying input lengths, padding and truncation were done to ensure that all sequences were uniformly sized for batch processing. Alongside this, an attention mask was generated to differentiate between actual content and padding. Finally, all previous processed inputs, are converted into tensors.

3.3.3) Feature Selection

Even though prior research typically saw fit to use popular input variables as predictors, this projects strategy saw it as beneficial to initially include all available information rather than risk losing potentially crucial input. Therefore, the initial financial feature set corresponded to Bao [22] 42 explanatory variables feature set. These comprised 28 raw accounting variables (Appendix A.5) and 14 financial ratios (Appendix A.6) which originated from the widely regarded and established models of Dechow [35], Beneish [37] and Altman [41].

The hypothesis was that, while Dechow's model variables may detect accrual manipulation, Beneish's model could catch revenue and expense manipulation, and Altman's model might highlight structural financial weaknesses that motivate fraud, cross-verifying potential red flags and reducing the chances of false positives.

Nonetheless, to create simpler, less redundant and more interpretable models the starting feature set was reduced. To accomplish this, first, constant, quasi-constant (very little variability) and duplicate features were addressed and as no constant nor duplicate features were present, only one, quasi-constant, feature was discarded ('*dch_inv*'). Then, Pearson´s correlation, a filter-based method, was employed to discard features that were highly correlated (using a threshold of 0.8) to avoid multicollinearity issues. This resulted in the exclusion of 16 raw accounting variables ('*rect*', '*ceq*', '*che*', '*lct*', '*ppegt*', '*sale*', '*txt*', '*xint*', '*invt*', '*ivst*', '*cogs*', '*ni*', '*lt*', '*csho*', '*at*', '*dp*'). Finally, model specific embedded based methods were also used. By them, features were chosen based on their impact on the output, so that variables with a contribution lower than the mean were dropped. When doing this, different features were deemed as not relevant for the model and models were trained on the original and on the reduced datasets where it was possible to observe that results were either better or marginally worse.

3.3.4) Final Dataset Exploration

To ensure validity and add trust to our dataset quality despite its potential known flaws, the conduction of the following experiments and analyses was adamant.

Target Variable Analysis:

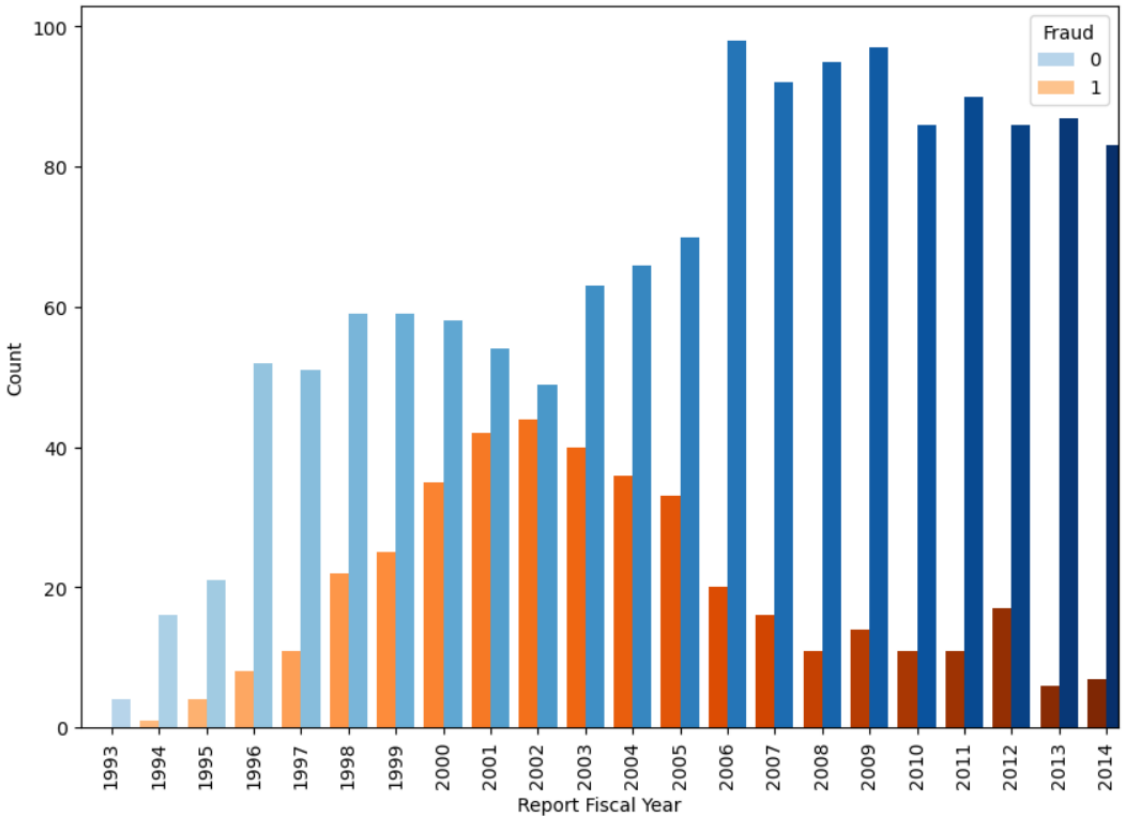

Figure 3.2: Distribution of fraudulent reports and non-fraudulent reports per fiscal year.

Figure 3.2 shows the distribution of fraud (1) and non-fraud reports (0) across the fiscal years and suggests a declining trend of fraud instances after 2003. This phenomenon seems to be in accordance with the legislative effort of the Sarbanes-Oxley Act (SOX) following Enron´s 2000 and another high-profile company bankruptcies [55] where the SEC engaged in a more aggressive stance.

Textual Features Analysis

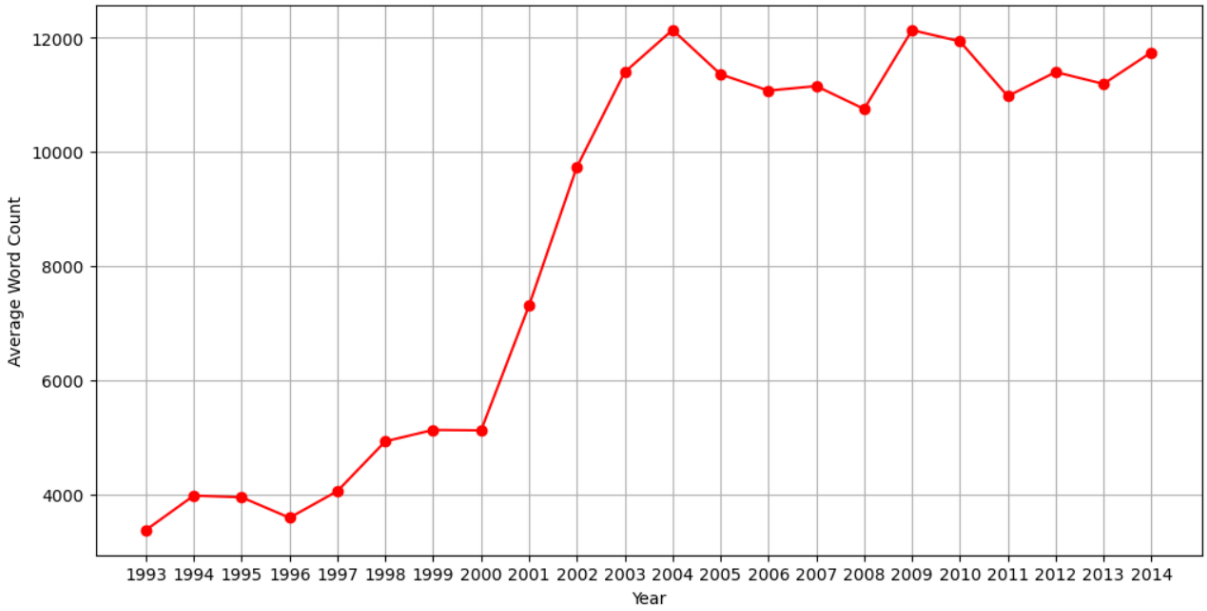

Figure 3.3: Average MD&A section word count, per year.

Figure 3.3 illustrates the average word count over the years, from 1997 to 2014. Initially, from 1997 to 2000, the average word count appeared to remain relatively stable, fluctuating between approximately 5.000 and 6.000 words. However, starting in 2001, there is a notable increase, reaching a peak by 2004 with the average word count roughly doubling over this period. Since then, although some fluctuations were recorded, the values stabilize around 11.000 words. This finding is in accordance with Lin [56], which found that the SOX act had an impact on the size of the MD&A section length but nonetheless made no material changes where text content and language style are concerned.

Bhattacharya [18] found that the average MD&A section of that spanning 1994 and 2013 contained 8.619 words. Sivasubramanian and Skillicorn ´s [19] average length of MD&As corresponded to 5.000 words, however his dataset spanned 1991 to 2006 which resulted in the direct exclusion of years with longer texts. Our average number of words is equal to approximately 10.228, which reflects the difference between the years and number of observations present within the different data samples and seems to be in accordance with prior studies. This fact will be especially important when dealing with the maximum tokens input limits.

Overall, a total 13.154 unique words were found and from these 10.353 were also present in the Loughran and McDonald dictionary [42] while the remaining 2.801 which mostly included firm specific names, locations and numbers.

Figure 3.4: Word clouds for the non-fraud (blue) and fraud (orange) classes.

The word clouds within Figure 3.4 depict the most frequently occurring words in the MD&A section of 10-K reports common within the negative class (blue) or the positive class (orange).

Overall, both word clouds share significant overlap in certain key financial terms, with "*million*" "*increase*" "*company*", "*fiscal*" and "*result*" standing out the distinction may lie in the surrounding context.

Financial Features Analysis

From the financial features statistics present in Table 3.4 immediately below; it is important to note the following:

Most financial ratios (9 out of 14 – '*dch_wc*', '*ch_rsst*', '*dch_rec*', '*dch_inv*', '*ch_cs*', '*issue*', '*EBIT*', '*bm*', '*dpi*') exhibit higher means for fraud firm years, indicating that these firms often report more significant changes when compared to non-fraud ones. However, certain variables appear to show almost no significant difference between fraud and non-fraud firms as is the case of '*soft_assets*' and '*ch_roa*', suggesting that these metrics may be less useful for distinguishing between the two groups. '*dch_inv*' appears to be especially bad at this, since its standard deviation is equal to 0.002, the lowest of all variables.

Table 3.4: Summary descriptive statistics table of financial ratio data.

Variable	Non-Fraud Mean	Fraud Mean	Mean Difference	Non-Fraud Std Dev	Fraud Std Dev	Std Dev Difference
dch_wc	0.007	0.014	0.008	0.112	0.105	-0.008
ch_rsst	0.023	0.072	0.048	0.237	0.254	0.016
dch_rec	0.012	0.032	0.020	0.074	0.078	0.004
dch_inv	0.007	0.016	0.009	0.053	0.056	0.002
soft_assets	0.622	0.625	0.003	0.216	0.222	0.006
ch_cs	0.162	0.304	0.142	0.862	0.575	-0.287
ch_cm	0.069	-0.055	-0.124	1.746	1.978	0.233
ch_roa	0.001	0.001	-0.001	0.210	0.185	-0.025

issue	0.944	0.973	0.029	0.229	0.161	-0.068
bm	0.478	0.545	0.067	1.006	0.542	-0.464
dpi	1.008	1.066	0.058	0.300	0.399	0.099
reoa	-0.813	-0.341	0.472	5.231	2.005	-3.226
EBIT	0.002	0.008	0.005	0.377	0.327	-0.050
ch_fcf	-0.009	-0.056	-0.047	0.263	0.328	0.065

Then from the raw accounting variables statistics present in Table 3.5 immediately below, it is important to note the following:

Across almost all variables, there is a notable pattern where the non-fraud class has positive mean values, while the fraud class appears negative. Simultaneously non-fraud reports also exhibit higher variability across many financial indicators, possibly indicating that non-fraud cases exhibit stronger financial performance and experience more dynamic financial changes.

Table 3.5: Summary descriptive statistics table of raw accounting variables data.

Variable	**Non-Fraud Mean**	**Fraud Mean**	**Mean Difference**	**Non-Fraud Std Dev**	**Fraud Std Dev**	**Std Dev Difference**
act	0.037	-0.129	0.166	1.062	0.734	0.328
ap	0.027	-0.095	0.123	1.052	0.787	0.265
at	0.036	-0.126	0.162	1.041	0.832	0.209
ceq	0.029	-0.102	0.131	1.098	0.523	0.575
che	0.032	-0.110	0.141	1.102	0.493	0.609
cogs	0.034	-0.118	0.152	1.056	0.766	0.290
csho	0.039	-0.137	0.176	1.106	0.449	0.657
dlc	0.012	-0.042	0.054	1.028	0.897	0.131
dltis	0.018	-0.061	0.079	1.025	0.907	0.119
dltt	0.040	-0.140	0.181	1.046	0.810	0.236
dp	0.038	-0.130	0.168	1.014	0.940	0.074
ib	0.035	-0.122	0.157	1.118	0.341	0.777
invt	0.048	-0.165	0.213	1.065	0.707	0.358
ivao	0.011	-0.037	0.048	1.117	0.376	0.741
ivst	0.020	-0.069	0.089	1.128	0.228	0.900
lct	0.033	-0.114	0.147	1.039	0.844	0.195
lt	0.039	-0.134	0.172	1.007	0.964	0.043
ni	0.036	-0.124	0.159	1.116	0.369	0.747
ppegt	0.027	-0.094	0.121	1.001	0.992	0.009
pstk	-0.011	0.039	-0.050	0.751	1.586	-0.835
re	0.007	-0.024	0.031	1.112	0.429	0.683
rect	0.028	-0.096	0.124	1.059	0.755	0.304
sale	0.044	-0.153	0.197	1.068	0.699	0.368
sstk	0.008	-0.028	0.036	1.095	0.556	0.540

txp	0.025	-0.086	0.110	1.052	0.790	0.262
txt	0.033	-0.114	0.147	1.116	0.365	0.751
xint	0.035	-0.123	0.158	1.036	0.857	0.179
prcc_f	-0.028	0.098	-0.126	1.034	0.869	0.165

3.4) Modelling

Given this study's goals different combinations of models were trained over three dimensions: types of variables, algorithm and data balancing technique.

Regarding the types of variables, financial, textual and financial combined with textual variables were used. The four different trained algorithms are specified in the following section and these three different data balancing techniques were used: no balancing, undersampling – via RandomUnderSampler – and oversampling – via Synthetic Minority Oversampling Technique. As for LLMs only text could be used, no financial models were built and the financial plus text models were rank averaged with financial classifiers, yielding a total of 26 models. The rank average approach consisted of calculating the mean probabilities for each model, assuming the weights of each model were the same. Additionally, attempting to improve the results obtained by the latter approach, a weighted rank average of the selected models' prediction values was computed as follows:

$$Rank\ Avg_{model} = w * rank(Rank\ Avg_{text}) + (1 - w) * rank(LR/XGBoost_{fin}) \quad (1)$$

Where:

- $Rank\ Avg_{text}$ corresponds to the model (utilising only textual variables) obtained via rank averaging of the first and last 512 tokens of the MD&A section.
- w corresponds to the weight given to the $Rank\ Avg_{text}$ model and is within [0.1, 0.9].
- $LR/XGBoost_{fin}$ corresponds to either the LR or XGBoost model (utilising only financial variables).

3.4.1) Machine Learning Classification Models' Selection

Baseline models

To establish a comparative analysis, financial and textual benchmarks had to be established with LR being selected as the baseline model. For the financial benchmark, no balancing was undertaken while for the textual the ['Negative', 'Positive', 'Uncertainty', 'Litigious', 'Constraining', 'Complexity', 'Weak Modal' and 'Strong Modal'] categories from the Loughran and McDonald [42] lexicon-based approach were used as features. Additionally, based on prior known performance, the following models were chosen:

Logistic Regression

LR is a widely used statistical method for binary classification. LR is a classification model that employs the sigmoid function as a cost function to return a probability value that can be mapped to discrete classes [70]. In the context of this study, it serves as a choice for initial exploration and analysis for its simplicity, interpretability, and efficiency.

eXtreme Gradient Boosting

XGBoost constructs a powerful predictive model through an iterative process that focuses on minimising errors. This iterative refinement, driven by gradient descent optimisation (the ability of the model to find the answer with the least error quickly), enables XGBoost to continually enhance its predictive accuracy by strategically updating the parameters of the decision trees.

XGBoost works by combining multiple decision trees. Each tree is built sequentially, with each new tree correcting the mistakes made by the previous ones. This approach allows the model to learn from its errors, gradually improving its accuracy with each iteration.

The model uses a technique called gradient descent optimization, which helps it quickly find the best parameters to minimize errors. In simple terms, this means that XGBoost continually refines its predictions by adjusting its decision trees based on the mistakes it makes. By doing so, it effectively enhances its ability to make accurate predictions, making it a popular choice for many predictive modeling tasks.

Likewise, XGBoost was chosen for being considered as a staple model within the realm of financial fraud detection for its past performances.

BERT & FinBERT

Concerning LLMs, although GPT models like GPT-4o would have been interesting to study, these are proprietary and as such require payment. In contrast, BERT [64, 65] and FinBERT [60, 66] are widely used and recognized for NLP tasks while being freely available via the HF Transformers library. This made us opt for the second group of models.

BERT (‘*google-bert/bert-base-uncased*’) and FinBERT (‘*yiyanghkust/finbert-pretrain*’) are both pre-trained models which can handle a maximum of 512 tokens, which can lead to losses of crucial information, affecting their performance. Their major difference lies on the fact that the first was trained on *BookCorpus*, a dataset consisting of 11.038 unpublished books and English Wikipedia (excluding lists, tables and headers) while the second is domain-specific and was trained on 10-K and 10-Q corporate reports, earnings call transcripts and analyst reports.

This difference would suggest that the FinBERT model could have a greater understanding of financial terminology, jargon, and context which was the motivation to choose this model. Additionally, other than Sivasubramanian ([19]), FinBERT has not been used in prior research in this context.

3.4.2) Evaluation Metrics

The first metric chosen (which was used a scorer) was Receiver Operator Characteristic (ROC) AUC as it, by combining the true positive rate and the false positive rate, could provide the probability that a randomly selected fraud sample would be ranked higher than a randomly selected non-fraud sample. This allows it to measure the model's capacity to correctly predict the "most important" positive class [18, 20, 22, 23, 24, 25, 26, 29, 30, 34].

Then, for the problem of fraud, as probabilities can be more informative than labels, NDCG@k, and a threshold-based "*Capture*" were also chosen. These metrics goal was to evaluate the model's ability to prioritize cases that warrant investigation and thereby reduce the cost of examining numerous predicted fraud cases, comparing the economic significance of the models [18, 21, 22, 25]

NDCG@k provides insight into the structure of top k observations that have the highest probability of being fraudulent by measuring how well the model sorts observations by their predicted score. It compares the actual ranking with the best possible ranking possible, ranging from 0 to 1. It is calculated by dividing the Discounted Cumulative Gain at position K (DCG@K) by the Ideal DCG.

$$DCG@K = \sum_{i=1}^{K} \frac{rel_i}{\log_2(i+1)} \tag{2}$$

$$NDCG@K = \frac{DCG@K}{IDCG@K} \tag{3}$$

Where:

- K corresponds to the user-defined number of observations that will be used as the cut-off point where to look for relevant items.
- rel_i corresponds to the relevance score of the item at position i.

The threshold-based "*Capture*" metric was inspired by Bhattacharya [18] and changed a bit to corresponds to a measure of how many actual frauds, in absolute terms, are being captured in the top k observations with the highest probabilities while also exceeding the threshold (having been correctly classified by the model). The use of the threshold is a significant

difference between Bhattacharya [18] since their approach by not using a threshold may be overestimating the model's performance by counting potentially low-probability predictions as "*captured*" positives.

To set a realistic estimation, a value of k equal to 10% of each set´s total observations, was deemed adequate. Although previously literature had used a k value of 1% of the total observations, given the size of our dataset, using the same value was not correct.

3.4.3) Fine-tuning/Hyperparameter search

For fine-tuning traditional models *RandomSearchCV*() was used while LLMs were handled using *Optuna*(). *RandomSearchCV*() was used with a cross-validation parameter of 5 and a number of iterations equal to 60, totaling 300 fits. This number of iterations was chosen since probabilistic explanations suggest that there is a 95% probability that only 60 iterations are needed to obtain an answer in the top performing 5% of all possible solutions [58]:

$$1 - q^n \geq p \rightarrow n \geq \frac{\log(1-p)}{\log(q)} \quad (4)$$

When q is equal to 0.95, which yields n≥59, which elucidates our choice of 60 iterations.

Optuna(), on the other hand was used with a number of 5 trials with the objective of maximizing the ROC-AUC score. For computational reasons, we were unable to apply the hyperparameter tuning to its full potential since augmenting the search parameters would considerably increase processing time, even when using powerful Virtual Machines. As such, parameters other than those found by tuning had to selected based on prior adequacy in literature. For LLM´s in specific, the number of training epochs (3) and the batch size (8) were chosen based on [19] since attempting to search within the [1, 2, 4, 6, 8] range for both hyperparameters was found to not viable after some experiments.

Table 3.6 provides the grid of hyperparameters for each of the models utilized. As BERT and FinBERT share the same hyperparameters as the same grid was used for both.

Table 3.6: Models and respective hyperparameter tuning grids.

Model	**Tuning Grid Parameters tested**
Logistic Regression	'penalty': ['l1', 'l2', 'elasticnet', 'none'] 'C': [0.001, 0.01, 0.1, 1, 10] 'solver': ['lbfgs', 'liblinear', 'saga'] 'max_iter': [100, 200, 500, 1000] 'tol': [1e-4, 1e-3, 1e-2] 'fit_intercept': [True, False] 'l1_ratio': [0.1, 0.5, 0.9]
XGBoost	'learning_rate': stats.uniform(0.0001, 0.2),

	'max_depth': stats.randint(3, 15), 'subsample': stats.uniform(0.5, 1.0), 'colsample_bytree': stats.uniform(0.4, 1.0), 'n_estimators': stats.randint(100, 500), 'min_child_weight': stats.randint(1, 10), 'gamma': stats.uniform(0, 0.5), 'reg_alpha': stats.uniform(0, 1.0), 'reg_lambda': stats.uniform(0, 1.0)
BERT+ FinBERT	'learning_rate' – (1e-6, 5e-5), 'weight_decay' – (0.0, 0.3)

For XGBoost in particular, besides *RandomSearchCV(), Optuna*() was also employed for fine-tuning the model, although unsuccessfully. This was done by means of a procedure to systematically refine the model. The procedure started by calculating the *'scale_pos_weight'* parameter as the ratio of negative to positive samples in the training labels. Then, the *'max_depth'* and *'min_child_weight'* parameters were tuned using cross-validation to identify the best parameters, which were then updated. The same optimization reasoning was also applied to other parameters like *'gamma'*, *'subsample'* and *'colsample_bytree'*. Then, the model was retrained with the optimized parameters incorporating 50 early stopping to prevent overfitting. Nevertheless, this approach did not improve the results of the model, which after more than 60 attempts and employing many different combinations of hyperparameters would require further inspection in future studies.

4. Results & Discussion

The present results are divided in three sections: the analysis of the tested models (section 4.1), the answers to the proposed research questions (section 4.2) and the limitations associated with this study (section 4.3). To assess the discriminative power of the models, the total and number of observations that were classified "*easily*" (when the actual class was 0 and the probability of being class 1 was 0.3 or lower or the actual class was 1 and the probability of being class 1 exceeded 0.7).

4.1) Model Analysis

Logistic Regression – present within Appendix D, pages 50-51.

LR baseline models yielded relatively poor performance. The financial baseline achieved a train AUC of 0.66 and a test AUC of 0.61 (see Figure D.1). Meanwhile, its performance on the other two metrics was similar, with a test set NDCG@k of 0.36, and a capture metric reflecting an inability to detect fraud (with a total 0 captures across validation and test sets). This was expected, given the LR inherent calibration and the difference in proportion of samples between the sets, leaving class 1 with an expected probability value of only 0.225 (83/370). No class 1 instances were "*easy*" while 92% of class 0 were.

Then, when balancing techniques were used results improved, achieving modest performances, although the AUC and NDCG@k were eerily like those of the baseline. Undersampling was able to capture 13 fraud instances on the test set, and oversampling captured 12, indicating their superiority to capture a higher number of relevant items, as can be seen from Figure D.1. Adittionally, undersampling was able to classify more instances "*easily*", with 12 class 1 and 33 class 0 instances to oversampling´s 5 and 17 instances over a larger set. In comparison, Bao [22] using 14 financial ratios and 28 raw financial data items achieved averaged values of 0.702 and 0.71 AUC and 0.023 and 0.011 values of NDCG@k when evaluating out-of-sample performance for the test period of 2003-2014.

Similarly, the text baseline also provided very weak results, with a train AUC of 0.53 and test AUC of 0.51, failing miserably at detecting fraud, achieving 2 captures on the test set.

Then, when using text data, (as can be seen from Figure D.2) LR fared significantly better, although it left a lot to be desired in terms of generalization and could classify fewer instances "*with ease*". The no balancing approach provided a solid test AUC of 0.73 and test NDCG@k of 0.61. On the other hand, oversampling, captured a record 18 fraud cases.

When using both financial and text data (see Figure D.3), performance was like that of the solely text-based approaches, although the capture metric rose. The best result was achieved with no balancing, yielding a test AUC of 0.73 and 21 captures on the test set.

Finally, it is possible to understand that without balancing, the LR models struggled significantly, capturing no instances for class 1 and excelling only at detecting the majority class (265 corresponding to 92.33%) for Fin, Text and Fin+Text approaches.

XGBoost – present within Appendix D, pages 51-52.

As can be seen from Figures D.4, D.5 and D.6 XGBoost suffers from severe overfitting and generalization issues as the train metrics are always the highest possible (AUC = 1, NDCG@k = 1 and threshold-based "*Capture*" = 120) which makes the results obtained by the model to not be trustworthy.

BERT & FinBERT – present within Appendix D, pages 53-55.

Regarding the BERT models with no balancing, the First 512 model resulted in a training AUC value of 0.76, indicating a good performance in distinguishing classes. However, the AUC values for the validation and test sets are 10% and 13% lower than the training set, which suggests inefficiency in generalizing from training to unseen data (see Figure D.7). This drop in the AUC values of the validation and test sets is consistent for the three models.

The Last 512 model attained a slightly better performance, reflected by a higher training AUC value of 0.82, as well as higher validation and test AUC values of 0.70 and 0.71, respectively. This improvement is also evident in the NDCG@k, which increased for all sets, especially for the training set.

The combination of both the First and Last 512 models reveals an increase in the capture metrics for all sets, while the AUC and NDCG@k values do not significantly vary from the Last 512 model.

When rank-averaging the BERT model with the LR, the model correctly classified 52 out of 83 class 1 instances (see Figure D.15). Although the model has a high false positive rate, in this case, it is not so severe as failing to classify class 1.

Regarding the FinBERT models, the three models demonstrate high training AUC values of 0.81, 0.85 and 0.81 for the First 512, Last 512 and RankAveraged, respectively. Nonetheless, for all three models, there is a decrease in the AUC values of the validation and test sets ranging from 9% to 13%, indicating their generalization inability.

The Last 512 model is the model with the highest AUC and NDCG@k values, while the capture values are lower than the ones of the RankAveraged model.

Finally, as expected, the FinBERT model combined with the LR led to an improvement in the performance, when comparing to the rank averaged BERT model (see Figure D.16). The false positive rate decreased accompanied by an increase in the true positive rate.

4.2) Addressing the Research Questions

In what follows we summarize the findings related with the first research question – RQ1: "*How can different ways of dealing with Large Language Models max token input affect classification performance in financial statement fraud detection problems*?"

Regarding LLM performance, to answer the research question, the token limit was handled in different ways and the effects of stopwords and number removal were tested.

As for the token limit, the last 512 tokens of the MD&A section seemed to have provided a better overall performance than the first 512. This can suggest that the last part of the text may capture more relevant information for fraud detection than the beginning of the text. Additionally, while rank-averaging the first with the last extent of 512 tokens may enhance certain aspects, as captures, as was claimed by Bhattacharya [18] it does not necessarily translate to better generalization.

Then, it was tested whether the removal of stopwords and numbers would impact performance for both no balancing and undersampling (Figure D.7 to Figure D.14) and no significant performance impact was observed, although FinBERT models saw slight improvements in NDCG and capture rates, suggesting that financial texts may contain numerical information and stopwords that contribute to the overall context.

In what follows we summarize the findings related with the second research question – RQ2: "*How do Large Language Models fare against benchmark models*?"

FinBERT models achieved AUC scores comparable to the best traditional models and surpassed these models in both NDCG and capture rates, suggesting superior ranking ability, which is crucial in applications where the position of correctly classified instances matters.

This may suggest that FinBERT, as was initially expected, is better suited for these types of tasks given the text financial corpus on which it was pre-trained. In fact, FinBERT obtained two models in the top performing 5, whose metrics are presented within Table 4.1.

Table 4.1: Top 5 performing models with their train, validation and test evaluation metrics.

Methods	Train AUC	Val AUC	Test AUC	Train NDC	Val NDC	Test NDC	Train Cap	Val Cap	Test Cap
Baseline (F)	0.66	0.64	0.61	0.41	0.50	0.36	2	0	0
LR (T, nb)	0.85	0.73	0.73	0.79	0.44	0.61	65	7	11
FinBERT (f+l) (nb)	0.81	0.70	0.71	0.72	0.35	0.44	81	10	16
LR (F+T, nb)	0.86	0.75	0.73	0.82	0.44	0.76	84	12	21
FinBERT+LR (F+T)	0.82	0.74	0.74	0.71	0.41	0.62	79	14	21
LR (F+T, us)	0.80	0.72	0.69	0.90	0.50	0.36	47	14	15

It is also important to mention regarding the RankAverage approach that although an apparent improvement in the capture metric was verified for both BERT and FinBERT RankAveraged models, no class 0 instances were "*easily*" classified, suggesting that the rank average approach is not the most trustworthy. Furthermore, regarding the balancing techniques compared despite undersampling having resulted in less true positive instances, a higher percentage was "*easily*" classified by the BERT and FinBERT models.

Finally, an important comparison to establish between traditional models and LLMs is their execution times. As LLMs are more complex, as previously mentioned by Sivasubramanian [19], their explainability diminishes and they suffer from long execution times when compared to their traditional model's counterpart.

In our project, when using limited tuning capabilities, LLMs took approximately twenty minutes to run, since they also incorporate preprocessing. This meant that the rank average models took close to fifty minutes to execute. Although traditional models also involve a preprocessing phase with long computational times (~20 minutes for retrieving the embeddings from *OpenAI* API), they have very fast training times. As such, the trade-off between performance and the availability of computational resources needs to also be considered carefully when deciding the models to use.

In what follows we summarize the findings related with the first research question – RQ3: "*Can Large Language Model assist stakeholders in signalling textual indicators/"red-flags" within the Management´s Discussion and Analysis section*?"

As language is dynamic, signalling specific red-flag words or sentences that could be related to fraud within a specific text is not a straightforward task as would be expected. On top of this, LLMs are on the black-box model spectrum, and, as such, understanding their predictions is harder than with traditional models. Nonetheless, one interpretability tool called

LIME (Local Interpretable Model-Agnostic Explanations) was used to attempt to uncover the reasoning behind the best performing model.

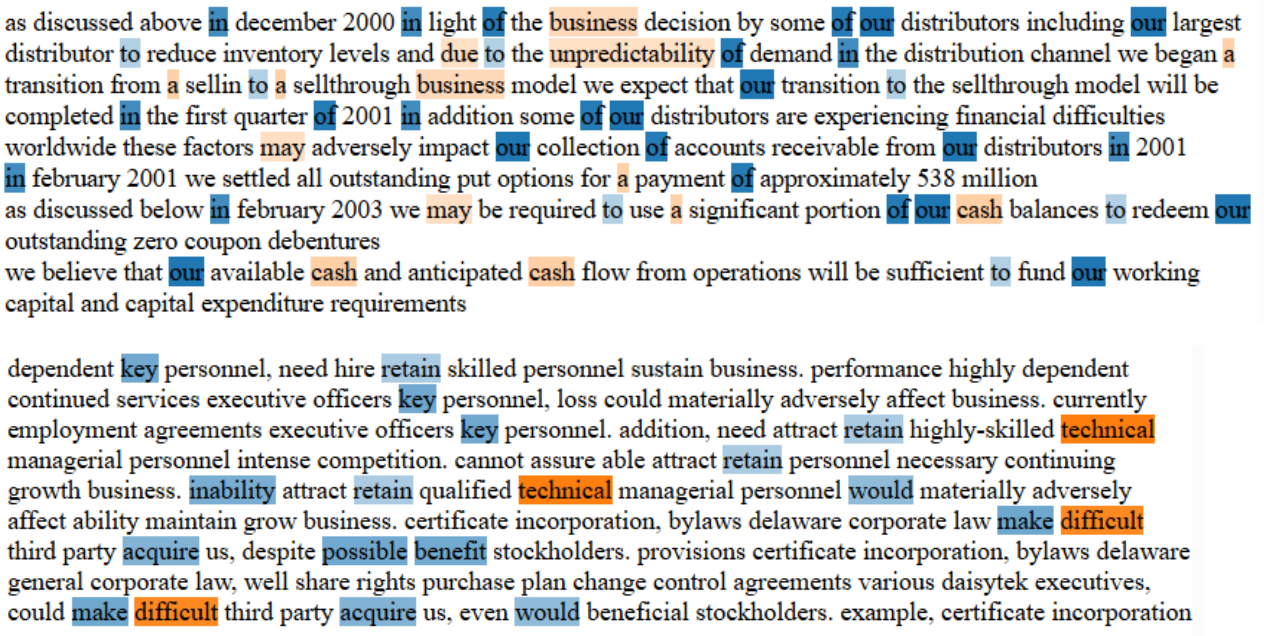

Figure 4.1: Two instances (for datasets with and without numbers and stopwords) with important words for the prediction process highlighted using LIME (Local Interpretable Model-Agnostic Explanations). Blue words are relative to non-fraud whereas orange ones are relative to fraud.

As can be seen from Figure 4.1, the model deemed as relevant towards the models' predictions a blend of suspected significant and insignificant words. In fact, a lot of stopwords such as "*our*" "*of*" "*in*" and "*to*" were highlighted which at first glance seems to not make sense. On the one hand, this may indicate that although commonly used, these words are important in terms of sentence structure and the overall meaning of the text.

On the other hand, as can be seen from the second image where stopwords and numbers were removed the model focuses on words that are more common to the understanding of business ("*key*", "*technical*"). Still, the results from the models appear to suggest that contrary to popular belief/common sense removing the stopwords does not bring added value to the predictions as demonstrated by the performance decay of the models that occurred with the removal of stopwords and numbers, mentioned in section 4. 2.

4.3) Limitations

Potential inherent target variable bias of neglecting the lag between fraud reports – According to Soltes [61], although the AAER database fares the best (64% of value-relevant events are omitted), the SEC tends to lag the initial public revelation of the misconduct in an average of 1,017 days after the initial public revelation. As such, control samples based on firms or firm-years that do not appear AAER databases may include firms and firm-years that did, in fact, have restatements, lawsuits, or SEC enforcement activity. Zavitsanos [57] also reinforces that fraud reports often go undetected for many years after their submittance, and when they get noticed, they usually affect a series of reports by the company.

However, this limitation is shared by all studies relying on the AAER´s to identify fraud.

Financial sector companies within the sample & Serial fraud problem – According to Sivasubramanian and Skillicorn [19] the language employed by companies from the financial sector appears to vary considerably from that of firms with distinct business natures. In our study companies were not selected based on their Standard Industrial Classification Code. Additionally, our sample comprises companies with multiple fraudulent reports that spanned multiple consecutive reporting periods, creating a situation of so-called "serial fraud" [22] These two occurrences could potentially have hindered the model's performance.

Data availability & Computational resources – Our study only utilized data from the reporting years of 1997 to 2014 and is therefore conditioned by the reality of said period. Although the period of sample can help reduce the potential effects of the first limitation, with a generous 6-year confidence interval, data to until 2018 could have been utilized. Additionally, as stated within section 3.2.1.3, our study did not include non-financial variables that could have potentially been useful to improve the predictions. As such, refinements could still be made to the models developed in this study.

Also, while both a local and a cloud environment were used for this project, computational resource constraints were still relevant since the hyperparameter search could not be accomplished to its fullest extent, possibly limiting and resulting in overfitting.

5. Conclusion & Future Work

Technology-enabled auditing methods are valuable tools, playing a complementary role to that of human expertise [67]. Together, fight against financial statement fraud can be enhanced.

Based on the obtained results, it can be concluded that LLMs, particularly FinBERT, fare well against benchmark and traditional models in financial text classification tasks. These models were able to not only match traditional models in terms of AUC but also exceed them in ranking performance and capture rates. This makes LLMs, especially domain-specific ones, valuable tools in scenarios where correctly ranking the most relevant instances are crucial.

However, their benefits come with increased computational costs and the potential for overfitting if not properly managed. All things considered, LR remains competitive, offering advantages in simplicity and speed.

Other findings related to the input token positioning suggest that the models using the last 512 tokens generally performed better than those using the first 512 tokens. This seems to indicate that important information may be located towards the end of the documents.

Overfitting was a significant concern across all models since this could translate to reduced performance in real-world applications. XGBoost was the model that suffered the most from this, achieving perfect training AUCs and perhaps requiring an even wider search grid to achieve convergence.

To address this issue, future work could potentially take two very distinct research avenues. One possibility is to attempt to enhance the model by applying distinct methods, either by compiling a larger corpus of financial texts, (which could help the models capture nuanced domain-specific language and terminology) with Edgar-Corpus, the text data source used in this study for instance, presents a valuable data source for this purpose. Or, alternatively, experiments could focus on investigating whether cost-sensitive learning yield promising results, since misclassification costs can vary significantly across different contexts.

Another possibility, would be to attempt to employ different types of variables and focus on correcting the possible data issues already identified on the limitations

By pursuing these avenues, we can work towards mitigating the risks of overfitting while improving model robustness and applicability in the financial domain.

References

[1] Bockel-Rickermann, C., Verdonck, T., & Verbeke, W. (2023). Fraud analytics: A decade of research. *Expert Systems With Applications*, *232*, 120605. https://doi.org/10.1016/j.eswa.2023.120605

[2] ACFE, 2024 Report to the Nation on Occupational Fraud and Abuse. Available at: www.acfe.com (Accessed in June 2024)

[3] Public Company Accounting Oversight Board. (2023). AS 2401: Consideration of fraud in a financial statement audit. Available at: https://pcaobus.org/oversight/standards/auditing-standards/details/AS2401 (Accessed in June 2024)

[4] Hogan, C. E., Rezaee, Z., Riley, R. A., & Velury, U. K. (2008). Financial Statement Fraud: Insights from the Academic Literature. *Auditing a Journal of Practice & Theory*, *27*(2), 231–252. https://doi.org/10.2308/aud.2008.27.2.231

[5] Loukas, L., Fergadiotis, M., Androutsopoulos, I., & Malakasiotis, P. (2021). EDGAR-CORPUS: Billions of Tokens Make The World Go Round. *Arxiv*. https://doi.org/10.18653/v1/2021.econlp-1.2

[6] U.S. Securities and Exchange Commission. (n.d.). Fraud-related enforcement actions. Available at: https://www.sec.gov/divisions/enforce/friactions (Accessed in July 2024)

[7] Sutherland, E. H. (1940). White-Collar criminality. *American Sociological Review*, *5*(1), 1-12. https://doi.org/10.2307/2083937

[8] Cressey, D. R. (1950). The criminal violation of financial trust. *American Sociological Review*, *15*(6), 738. https://doi.org/10.2307/2086606

[9] Jacobs, J. A., & Cressey, D. R. (1954). Other People's Money. A Study in the Social Psychology of Embezzlement. The Journal of Criminal Law Criminology and Police Science, 45(4), 464. https://doi.org/10.2307/1140029

[10] Albrecht, W. S., Howe, K. R., & Romney, M. B. (1984). *Deterring Fraud: The Internal Auditor's perspective*. http://ci.nii.ac.jp/ncid/BA24265998

[11] Wolfe, D. T., & Hermanson, D. (2004). The fraud diamond: Considering the four elements of fraud. The CPA Journal, December, 1–5.

[12] Kranacher, M., & Riley, R. (2011). *Forensic Accounting and Fraud Examination*. John Wiley & Sons.

[13] Zack, G. M. (2012). *Financial Statement Fraud*. https://doi.org/10.1002/9781118527436

[14] Wells, J. T. (2014). *Principles of Fraud Examination*. John Wiley & Sons. ISBN: 978-1-118-80326-4.

[15] Association of Certified Fraud Examiners. (2014). Fraud Examiners Manual - International Edition. Austin, TX: Association of Certified Fraud Examiners.

[16] Hadi, M. U., Tashi, Q. A., Qureshi, R., Shah, A., Muneer, A., Irfan, M., Zafar, A., Shaikh, M. B., Akhtar, N., Wu, J., & Mirjalili, S. (2023). A Survey on Large Language Models: Applications, Challenges, Limitations, and Practical Usage. *Research Gate*. https://doi.org/10.36227/techrxiv.23589741.v1

[17] Kitchenham, B. and Charters, S. (2007) Guidelines for Performing Systematic Literature Reviews in Software Engineering, Technical Report EBSE 2007-001, Keele University and Durham University Joint Report. Available at: https://legacyfileshare.elsevier.com/promis_misc/525444systematicreviewsguide.pdf (Accessed in July 2024)

[18] Bhattacharya, I., & Mickovic, A. (2024). Accounting fraud detection using contextual language learning. *International Journal of Accounting Information Systems*, *53*, 100682. https://doi.org/10.1016/j.accinf.2024.100682

[19] Sivasubramanian, S. V., & Skillicorn, D. (2024). Predicting fraud in MD&A sections using deep learning. *Journal of Business Analytics*, *7*(3), 197–206. https://doi.org/10.1080/2573234x.2024.2342773
[20] Schneider, M., & Brühl, R. (2023). Disentangling the black box around CEO and financial information-based accounting fraud detection: machine learning-based evidence from publicly listed U.S. firms. *Journal of Business Economics*, *93*(9), 1591–1628. https://doi.org/10.1007/s11573-023-01136-w
[21] Xu, X., Xiong, F., & An, Z. (2022). Using machine learning to predict Corporate fraud: Evidence based on the GONE Framework. *Journal of Business Ethics*, *186*(1), 137–158. https://doi.org/10.1007/s10551-022-05120-2
[22] Bao, Y., Ke, B., Li, B., Yu, Y. J., & Zhang, J. (2019). Detecting accounting fraud in publicly traded U.S. firms using a machine learning approach. *Journal of Accounting Research*, *58*(1), 199–235. https://doi.org/10.1111/1475-679x.12292
[23] Bertomeu, J., Cheynel, E., Floyd, E., & Pan, W. (2020). Using machine learning to detect misstatements. *Review of Accounting Studies*, *26*(2), 468–519. https://doi.org/10.1007/s11142-020-09563-8
[24] Craja, P., Kim, A., & Lessmann, S. (2020). Deep learning for detecting financial statement fraud. *Decision Support Systems*, *139*, 113421. https://doi.org/10.1016/j.dss.2020.113421
[25] Brown, N. C., Crowley, R. M., & Elliott, W. B. (2019). What Are You Saying? Using topic to Detect Financial Misreporting. *Journal of Accounting Research*, *58*(1), 237–291. https://doi.org/10.1111/1475-679x.12294
[26] Hajek, P., & Henriques, R. (2017). Mining corporate annual reports for intelligent detection of financial statement fraud – A comparative study of machine learning methods. *Knowledge-Based Systems*, *128*, 139–152. https://doi.org/10.1016/j.knosys.2017.05.001
[27] Kim, Y. J., Baik, B., & Cho, S. (2016). Detecting financial misstatements with fraud intention using multi-class cost-sensitive learning. *Expert Systems With Applications*, *62*, 32–43. https://doi.org/10.1016/j.eswa.2016.06.016
[28] Goel, S., & Uzuner, O. (2016). Do sentiments matter in fraud detection? Estimating semantic orientation of annual reports. *Intelligent Systems in Accounting Finance & Management*, *23*(3), 215–239. https://doi.org/10.1002/isaf.1392
[29] Purda, L., & Skillicorn, D. (2014). Accounting variables, deception, and a bag of words: Assessing the tools of fraud detection. *Contemporary Accounting Research*, *32*(3), 1193–1223. https://doi.org/10.1111/1911-3846.12089
[30] Throckmorton, C. S., Mayew, W. J., Venkatachalam, M., & Collins, L. M. (2015). Financial fraud detection using vocal, linguistic and financial cues. *Decision Support Systems*, *74*, 78–87. https://doi.org/10.1016/j.dss.2015.04.006
[31] Lin, C., Chiu, A., Huang, S. Y., & Yen, D. C. (2015). Detecting the financial statement fraud: The analysis of the differences between data mining techniques and experts' judgments. *Knowledge-Based Systems*, *89*, 459–470. https://doi.org/10.1016/j.knosys.2015.08.011
[32] Humpherys, S. L., Moffitt, K. C., Burns, M. B., Burgoon, J. K., & Felix, W. F. (2010). Identification of fraudulent financial statements using linguistic credibility analysis. *Decision Support Systems*, *50*(3), 585–594. https://doi.org/10.1016/j.dss.2010.08.009
[33] Glancy, F. H., & Yadav, S. B. (2010). A computational model for financial reporting fraud detection. *Decision Support Systems*, *50*(3), 595–601. https://doi.org/10.1016/j.dss.2010.08.010

[34] Ravisankar, P., Ravi, V., Rao, G. R., & Bose, I. (2010). Detection of financial statement fraud and feature selection using data mining techniques. *Decision Support Systems*, *50*(2), 491–500. https://doi.org/10.1016/j.dss.2010.11.006

[35] Dechow, P. M., Ge, W., Larson, C. R., & Sloan, R. G. (2011). Predicting material accounting misstatements*. *Contemporary Accounting Research*, *28*(1), 17–82. https://doi.org/10.1111/j.1911-3846.2010.01041.x

[36] Cecchini, M., Aytug, H., Koehler, G. J., & Pathak, P. (2010). Detecting management fraud in public companies. *Management Science*, *56*(7), 1146–1160. https://doi.org/10.1287/mnsc.1100.1174

[37] Beneish, M. D. (1999). The detection of earnings manipulation. *Financial Analysts Journal*, *55*(5), 24–36. https://doi.org/10.2469/faj.v55.n5.2296

[38] Wilson, T., Wiebe, J., & Hoffmann, P. (2005). Recognizing contextual polarity in phrase-level sentiment analysis. In HLT'05 Proceedings of the Conference on Human Language Technology and Empirical Methods in Natural Language Processing. Association for Computational Linguistics: Stroudsburg, PA; 347–354. https://doi.org/10.3115/1220575.1220619

[39] Newman, M. L., Pennebaker, J. W., Berry, D. S., & Richards, J. M. (2003). Lying Words: Predicting Deception from Linguistic Styles. Personality and Social Psychology Bulletin, 29(5), 665–675. https://doi.org/10.1177/0146167203029005010

[40] Eaglesham, J. 2013. Accounting fraud targeted: With crisis-related enforcement ebbing, SEC is turning back to Main Street. Wall Street Journal, May 27 Retrieved from https://www.wsj.com/articles/SB10001424127887324125504578509241215284044

[41] Altman, E. I. (1968). Financial ratios, discriminant analysis and the prediction of corporate bankruptcy. The Journal of Finance, 23(4), 589-609. https://doi.org/10.1111/j.1540-6261.1968.tb00843.x

[42] Loughran, T., and B. McDonald. 2011. When is a liability not a liability? The Journal of Finance 66 (1): 35–65. https://doi.org/10.1111/j.1540-6261.2010.01625.x

[43] Stanford Law School. Finding federal legislative history. Available at: https://guides.law.stanford.edu/c.php?g=646860&p=4534400 (Accessed in March 2024)

[44] U.S. Securities and Exchange Commission. How to read a 10-K. Available at: https://www.investor.gov/introduction-investing/getting-started/researching-investments/how-read-10-k (Accessed in March 2024)

[45] Zenzerović, R., & Šajrih, J. (2023). Financial statements fraud identifiers. Economic Research-Ekonomska Istraživanja, 36(3). https://doi.org/10.1080/1331677x.2023.2218916.

[46] Yi, Z., Cao, X., Chen, Z., & Li, S. (2023). Artificial Intelligence in Accounting and Finance: Challenges and Opportunities. IEEE Access, 11, 129100–129123. https://doi.org/10.1109/access.2023.3333389

[47] U.S. Securities and Exchange Commission. Accounting and Auditing Enforcement. Available at: https://www.sec.gov/enforcement-litigation/accounting-auditing-enforcement-releases

[48] University of Southern California. AAER dataset. Available at: https://sites.google.com/usc.edu/aaerdataset/home?authuser=0 (Accessed in May 2024)

[49] Hugging Face. EDGAR corpus.. Available at: https://huggingface.co/datasets/eloukas/edgar-corpus (Accessed in March 2024)

[50] GitHub. JarFraud/FraudDetection: Accounting Fraud Detection Using Machine Learning. GitHub. Available at: https://github.com/JarFraud/FraudDetection (Accessed in April 2024)

[51] Alpha Vantage. Available at: https://www.alphavantage.co/ (Accessed in January 2024)
[52] Tradefeeds. Available at: https://tradefeeds.com/ (Accessed in January 2024)
[53] Financial Modeling Prep. Available at: https://site.financialmodelingprep.com/
[54] Cornell Law School,.Legal Information Institute. 17 CFR § 229.10 - (Item 10) General Available at: https://www.law.cornell.edu/cfr/text/17/229.10. (Accessed in April 2024)
[55] Lane, R., & O'Connell, B. T. (2009). The changing face of regulators' investigations into financial statement fraud. Accounting Research Journal, 22(2), 118–143. https://doi.org/10.1108/10309610910987484.
[56] Lin, C. C., Chiu, A. A., Huang, S. Y., & Yen, D. C. (2015b). Detecting the financial statement fraud: The analysis of the differences between data mining techniques and experts' judgments. Knowledge-Based Systems, 89, 459–470. https://doi.org/10.1016/j.knosys.2015.08.011.
[57] Zavitsanos, E., Mavroeidis, D., Bougiatiotis, K., Spyropoulou, E., Loukas, L., & Paliouras, G. (2021). Financial misstatement detection. https://doi.org/10.1145/3490354.3494453.
[58] Cross Validated, The "Amazing hidden power" of random search? (2022). Available at: https://stats.stackexchange.com/questions/561164/the-amazing-hidden-power-of-random-search (Accessed in June 2024)
[59] EvidentlyAI, Normalized Discounted Cumulative Gain (NDCG) explained. Available at: https://www.evidentlyai.com/ranking-metrics/ndcg-metric (Accessed in March 2024)
[60] Yang, Y., Uy, M. C. S., & Huang, A. (2020). *FinBERT: A Pretrained Language Model for Financial Communications*. arXiv (Cornell University). https://doi.org/10.48550/arxiv.2006.08097
[61] Soltes, E., & Stanford, M. B. (2017). Proxies and Databases in Financial Misconduct Research. The Accounting Review, 92(6), 129-163. Georgetown McDonough School of Business Research Paper No. 2811778.
[62] OpenAI, New embedding models and API updates (2024). Available at: https://openai.com/index/new-embedding-models-and-api-updates/ (Accessed in January 2024)
[63] OpenAI Platform, Embeddings. Available at: https://platform.openai.com/docs/guides/embeddings (Accessed in January 2024)
[64] Devlin, J., Chang, M., Lee, K., & Toutanova, K. (2018, October 11). *BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding*. arXiv.org. https://arxiv.org/abs/1810.04805.
[65] Hugging Face, google-bert/bert-base-uncased. Available at: https://huggingface.co/google-bert/bert-base-uncased (Accessed in July 2024)
[66] Hugging Face, yiyanghkust/finbert-pretrain. Available at: https://huggingface.co/yiyanghkust/finbert-pretrain (Accessed in July 2024)
[67] Abidoye, A., Awolowo, I. F., & Chan, D. (2023). Bridging the Gap: Integrating Forensic Accounting Skillsets for Enhanced Audit Quality in the Post-Pandemic Era. Journal of Forensic Accounting Profession, 3(2), 63–81. https://doi.org/10.2478/jfap-2023-0010.
[68] Seiffert, C., Khoshgoftaar, T. M., Van Hulse, J., & Napolitano, A. (2008). RUSBoost: Improving classification performance when training data is skewed. Proceedings - International Conference on Pattern Recognition/Proceedings/International Conference on Pattern Recognition. https://doi.org/10.1109/icpr.2008.4761297
[69] Scikit-learn, StandardScaler. Available at: https://scikit-learn.org/stable/modules/generated/sklearn.preprocessing.StandardScaler.html (Accessed on June 2024)June 2024)
[70] Bishop, C. M. (2007). Pattern recognition and machine learning. Journal of Electronic Imaging, 16(4), 049901. https://doi.org/10.1117/1.2819119

Appendix A – Introduction, Literature Review & Methods

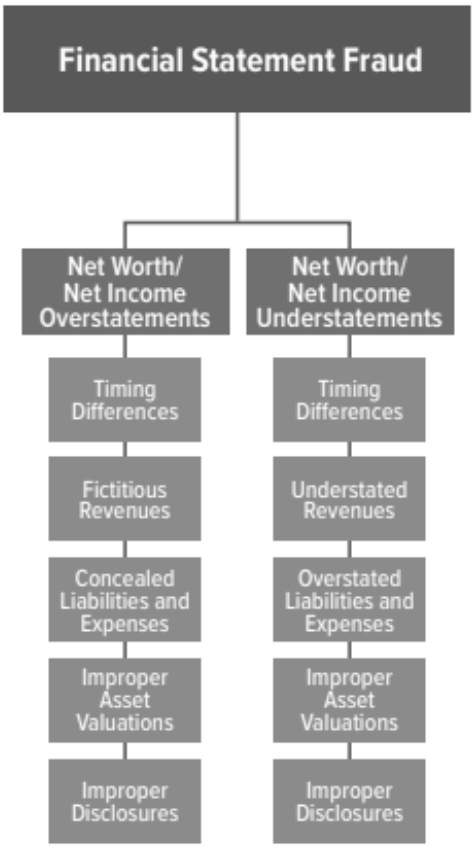

Figure A.1: Financial statement fraud tree – Adapted from ACFE [2]

Table A.2: Summary table of the characteristics of the data used in prior research, as well as its respective sources.

Ref.	Period	Sample (F/nF)	Feature type	Variables	Label Source	Source
[18]	1994-2013	289/30876	TXT	BERT embeddings	AAERs	EDGAR 10-K filings, Compustat
[19]	1991-2006	1130/4550	TXT	FastText word embedding: 300-dimensional, CBOW pre-trained word vectors for English with a window size of 5	AAERs	EDGAR 10-K filings
[20]	2000-2018	198/30178	FIN/nFIN	28 raw accounting variables and CEO characteristics (non-financial)	AAERs	Compustat and BoardEx
[21]	2009-2018	4440/31482	FIN/nFIN	45 variables: Fin: financial ratios and raw accounting variables; nFin: Printed news, online news, social media posts' counts	China Stock Market and Accounting Research (CSMAR)	CSMAR, Chinese Research Data Services (CNRDS), and social media data from CNRDS
[22]	1991-2014	1171/206026	FIN	28 raw accounting variables	AAERs	Compustat
[23]	2001-2014	3599/54354	FIN/nFIN	102 variables including financial ratios, audit variables, credit rating variables and corporate governance variables	Audit Analytics Non-Reliance Restatement database	Public records, Compustat, CRSP, and Audit Analytics
[24]	1993-2019	208/7549	FIN+TXT	Fin: 47 financial ratios; Text: tfidf, word2vec (300 dims), GPT-2 embeddings; Ling: Loughran and McDonald (2011)	AAERs	Compustat, EDGAR 10-K filings
[25]	1994 -2012	511/5427	FIN+TXT +LING	Fin: raw accounting variables and ratios; Text: LDA for topic modelling; Ling: Lexicon based - Loughran and McDonald	AAERs	Compustat, CRSP

[26]	2005-2015	311/311	FIN+LING	Fin: financial ratios; Ling: Lexicon approach using Loughran and McDonald (2011)	AAERs	EDGAR 10-K filings
[27]	1992-2005	788/2156	FIN/nFIN	49 variables: Fin: financial ratios and raw accounting variables; nFin: CEO Salary, change in operating lease activity, etc.	From Hennes et al. (2008)	Compustat and other non specified financial databases
[28]	1994-2012	180/180	LING+TXT	Ling: Lexicon-based - Loughran and McDonald (2011) dictionary, the MPQA subjectivity lexicon (Wilson et al., 2005) and the LIWC (Pennebaker et al., 2007)	AAERs	Lexis-Nexis, Compustat, The Wall Street Journal, The New York Times, The Financial Times
[29]	1994-2006	1127/3768	TXT	BOW	AAERs	EDGAR 10-K and 10-Q filings
[30]	2007	41/1531	FIN+LING	Fin: raw accounting variables; Ling: Lexicon based using Linguistic Inquiry and Word Count (LIWC) and non-verbal vocal cues	Audit Analytics restatements database	Compustat, University of Chicago's Center for Research in Security Prices, ThomsonReuters
[31]	1998-2010	129/447	FIN/nFIN	32 features: Fin: financial ratios; nFin: Number of CEO switches in the past three years, etc.	Taiwan Securities and Futures Bureau, Securities and Futures Investors Protection Center	Taiwan Economic Journal (TEJ) database and the Financial Supervisory Commission of The Executive Yuan
[32]	1995-2004	101/101	LING	Lexicon based - 24 linguistic cues from text	AAERs	EDGAR 10-K filings
[33]	2006-2008	11/20	TXT	BOW	AAERs	EDGAR 10-K filings

[34]	Not specified	101/101	FIN	35 features: financial ratios and raw accounting variables	Various Chinese stock exchanges	Various Chinese stock exchanges
[35]	1982-2005	676/2190	FIN/nFIN	Fin: raw accounting variables and ratios; nFin: abnormal change in employees, etc.	AAERs	Compustat
[36]	1991-2003	205/6427	FIN	Raw accounting variables and financial ratios	AAERs	Compustat
[37]	1982-1992	74/2332	FIN	8 financial ratios	AAERs	Compustat

Table A.3: Summary table of the types of models used in prior research.

Ref.	ML Models	DL Models	LLM´s	Best Classifier
[18]	RUSBoost, LDA, Ensemble	None	BERT-Base model (uncased)	BERT (82.6)
[19]	SVM, XGBoost	ANN, RNN, LSTM, BiLSTM, GRU, CNN, TCN, HAN, Transformer	BERT, FinBERT, *GPT-3, BART	Transformer (77, 91)
[20]	LR, SVM, RF, XGBoost	NN	None	RF (92.6)
[21]	RF, GBDT, RUSBoost, LR, SVM	ANN	None	RF (71.9)
[22]	RUSBoost	None	None	RUSBoost (72.5)
[23]	GBRT, RF, RUSBoost, LR	None	None	RF (77.5)
[24]	LR, RF, SVM, XGBoost	ANN, HAN	**GPT-2	HAN (92.6)
[25]	LDA, F-score model, Textual style models	None	None	Topic, F-score and Style model (75.2)
[26]	LR, NB, BBN, SVM, DT, Ensemble classifiers: Bagging, RF	MLP, VP	None	BBN (90.3)
[27]	Multinomial LR, SVM, Bayesian network	None	None	LR (88.4)
[28]	SVM	None	None	SVM (81.8)
[29]	SVM, DTs	None	None	SVM (89.0)
[30]	GLRT, Logistic Regression, Naïve Bayes, KNN	None	None	GLRT (81.0)
[31]	LR, CART	ANN	None	ANN (92.8)
[32]	LR, C4.5 DT, LWL, SVM, NB	None	None	C4.5 & NB (67.3)
[33]	Expectation Maximization Clustering, Hierarchical Clustering	None	None	Hierarchical Clustering (83.9)
[34]	MLFF, SVM, GP, GMDH, LR	PNN	None	PNN (98.1)
[35]	LR	None	None	LR (63.7)
[36]	SVM	None	None	SVM (82.0)
[37]	Probit regression	None	None	Probit Regression (89.5)

Table A.4: Summary table of the balancing techniques and evaluation metrics used in prior literature.

Ref.	Balancing	Evaluation metrics
[18]	Not specified	AUC, NDCG@k
[19]	Algorithm level techniques/model parameters: weight function (for DL), and scale_pos_weight (for ML)	F1-score, Accuracy
[20]	Not specified	AUC, Sensitivity, Specificity, Accuracy
[21]	Class weights via a loss function	NDCG@k, Precision@k, Recall@k
[22]	Random undersampling (RUSBoost)	AUC and NDCG@k
[23]	Random undersampling (RUSBoost)	TPR, FPR, Precision, Recall, ROC AUC, Fβ scores
[24]	Undersampling: Fraud-to-non-fraud ratio of 1:4	Accuracy, Sensitivity, Specificity, F1-score, F2-score, AUC
[25]	Not specified	AUC, Accuracy, NDCG@k
[26]	Matched-pair sampling approach	Accuracy, TPR, TNR, F-measure, AUC, MC (combination of FPR and FNR)
[27]	Cost-sensitive learning using MetaCost	Accuracy
[28]	Matched-pair sampling approach	Accuracy, Precision, Recall, F-score
[29]	Not specified	AUC
[30]	Not specified	TPR, FPR, AUC
[31]	Matched-pair sampling approach	Accuracy, Type II error
[32]	Matched-pair sampling approach	Accuracy, Precision, Recall, F-measure, RMSE
[33]	Not specified	Sign test for statistical significance
[34]	Not specified	Accuracy, Sensitivity, Specificity, AUC
[35]	Not specified	Type I and Type II error rates, marginal analysis, sensitivity analysis
[36]	Not balanced	Accuracy, Type I error (false positives), and Type II error (false negatives)
[37]	Not specified	Type I and Type II error rates, Wilcoxon Z, median χ^2 tests

Table A.5: Financial explanatory variables used in this study– raw accounting variables, including their feature name and description.

Feature	Description
act	Total Current Assets
ap	Account Payable
at	Total Assets
ceq	Total Common/Ordinary Equity

che	Cash and Short-Term Investments
cogs	Cost of Goods Sold
csho	Common Shares Outstanding
dlc	Total Debt in Current Liabilities
dltis	Long-Term Debt Issuance
dltt	Total Long-Term Debt.
dp	Depreciation and Amortization
ib	Income Before Extraordinary Items
invt	Total Inventory.
ivao	Investment and Advances
ivst	Total Short-Term Investments
lct	Total Current Liabilities
lt	Total Liabilities
ni	Net Income (Loss)
ppegt	Total Property, Plant and Equipment
pstk	Total Preferred/Preference Stock (Capital)
re	Retained Earnings
rect	Total Receivables
sale	Sales/Turnover (Net)
sstk	Sale of Common and Preferred Stock
txp	Income Taxes Payable
txt	Total Income Taxes
xint	Total Interest and Related Expense
prcc_f	Price Close, Annual, Fiscal

Table A.6: Financial explanatory variables used in this study– financial ratios, including their feature name, description and formula.

Feature	**Description**	**Formula**
dch_wc	WC accruals [35]	$\frac{(\Delta act - \Delta che) - (\Delta lct - \Delta dlc - \Delta txp)}{Average\ at}$
		$\frac{(\Delta WC + \Delta NCO + \Delta FIN)}{(Average\ at)}$
ch_rsst	RSST accruals [35]	$WC = (act - che) - (lct - dlc)$
		$NCO = (at - act - ivao) - (lt - lct - dltt)$
		$FIN = (ivst + Longterm\ Investments) - (dltt + dlc + pstk)$
dch_rec	Change in receivables [35]	$\frac{\Delta \text{rect}}{Average\ at}$
dch_inv	Change in inventory [35]	$\frac{\Delta \text{invt}}{Average\ at}$
soft_assset	% Soft assets [35]	$\frac{\text{at} - \text{ppegt} - \text{che}}{at}$
dpi	Depreciation index [37]	$DEPI = \frac{dp_{t-1}/(dp_{t-1} + ppegt_{t-1})}{dp_t/(dp_t + ppegt_t)}$

ch_cs	Change in cash sales [35]	$sale - \Delta\text{rect}$
ch_cm	Change in cash margin [35]	$1 - \frac{cogs - \Delta invt + \Delta ap}{sale - \Delta rect}$
ch_roa	Change in return on assets [35]	$\frac{ni_t}{Average\ at_t} - \frac{ni_{t-1}}{Average\ at_{t-1}}$
ch_fcf	Change in free cash flows [35]	$\frac{\Delta[re - ch\_rsst]}{Average\ at}$
reoa	Retained earnings over total assets [41]	$B = \frac{re}{at}$
ch_EBIT	Earnings before interest and taxes over total assets [41]	$C = \frac{EBIT}{at}$
issue	Actual issuance [35]	An indicator variable coded 1 if the firm issued securities during year t, 0 otherwise
bm	Book-to-market [35]	$\frac{\text{ceq}}{Market\ Value}$

Appendix B – Exploratory Data Analysis

This appendix section comprises visualizations created to investigate the datapoints collected and help make decisions regarding changes to the dataset.

Outlier Analysis

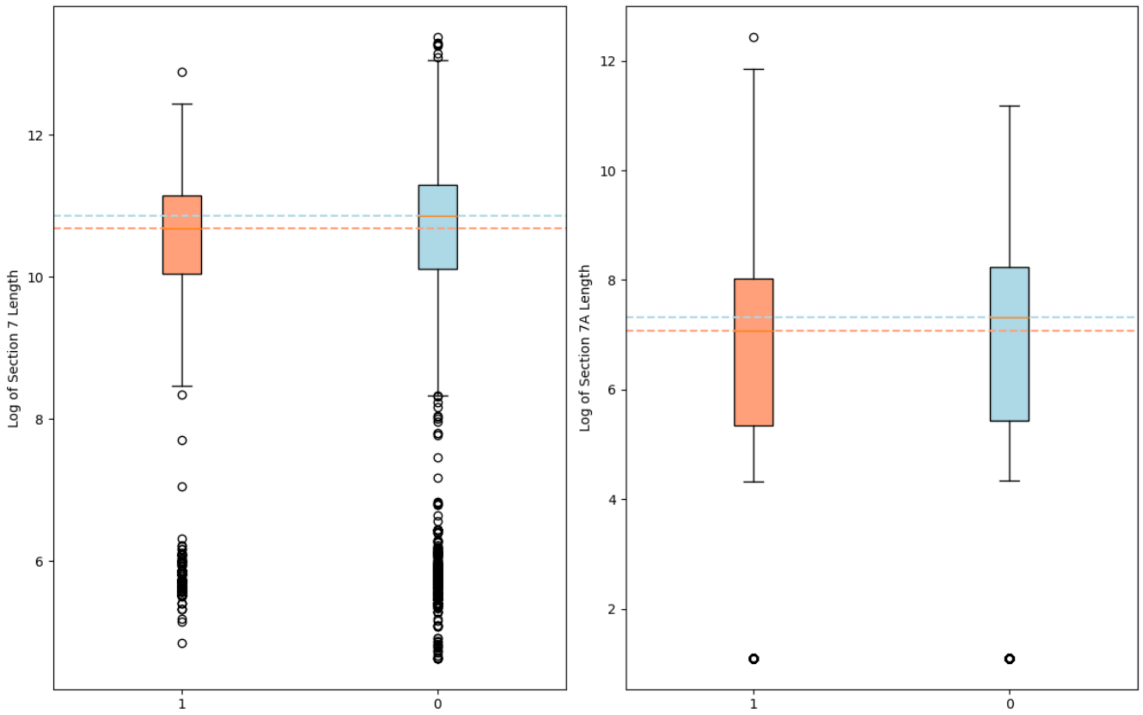

Figure B.1: Box plots of the log length of the 10-K sections 7 (left) and 7A (right).

Figure B.1 displays two box plots comparing the log of section 7 – MD&A – and section 7A – Quantitative and Qualitative Disclosures about Market Risk – lengths by class. For the MD&A section the median log length for class 1 is slightly lower than for class 0. Class 1 shows a wider spread, including more outliers on both the lower and upper ends. Lower outliers (which were calculated via Q1-1.5*IQR) mostly correspond to rows which did not contain the text but rather a mention that it should be "*incorporated herein by reference*".

In Section 7A both classes are nearly identical, showing substantial overlap. Class 1 appears to have fewer lower outliers, which were confirmed to correspond to Not a Number (NaN).

In another experiment, after removing the lower outliers (Q1-1.5*IQR) identified in both sections, the new MD&A section outliers (Q1-1.5*IQR) seemed to be relieved of errors, but on the other hand section 7A appeared to still display some incorrect cases. These appeared to be of two sorts, the item was considered as non-applicable (e.g. "*not applicable*", "*no disclosure is required under this item*", "*disclosures are not required at this time*") or the text was incorporated on distinct items across the form and not in the specific item 7A where it should be (e.g. "*for quantitative and qualitative information about market risk, refer to item 8, notes 1 and 2 of the notes to consolidated financial statements*", "*response to this item is included in "item 7 - management's discussion and analysis of financial condition and results of operations - market risk.*"").

The first occurrence stems from the fact that smaller reporting companies (companies that fulfill certain revenue or public float requirements) are not required to file some of the standard 10-K report items, including "*Quantitative and qualitative disclosures about market risk*" – Item 305 – (as can be seen from Table B.1).

Table B.1: Index of Scaled Disclosure Available to Smaller Reporting Companies – Adapted from Cornell Law School.[54]

Item 101	Description of business.
Item 201	Market price of and dividends on registrant's common equity and related stockholder matters.
Item 302	Supplementary financial information.
Item 305	Quantitative and qualitative disclosures about market risk.
Item 402	Executive compensation.
Item 404	Transactions with related persons, promoters and certain control persons.
Item 407	Corporate governance.
Item 503	Prospectus summary.
Item 504	Use of proceeds.
Item 601	Exhibits.

The second occurrence elucidates that the contents of section_7A could have already been stated in another section of the report. These two instances added doubts on whether it would be correct to use the text contents of this section in our study. Additionally, knowing whether prior research had made use of item 7A was not straightforward as no explicit mentions were made. However, Bhattacharya [18] states to have determined the end of the MD&A section by searching for the variations of "*Item 8. Consolidated Financial Statements*". Although it is not specifically stated, as item 7A comes immediately after item 7 and before item 8 this probably means that said section was used in their study. Because of all this, we chose to cut the section.

MD&A section length Analysis

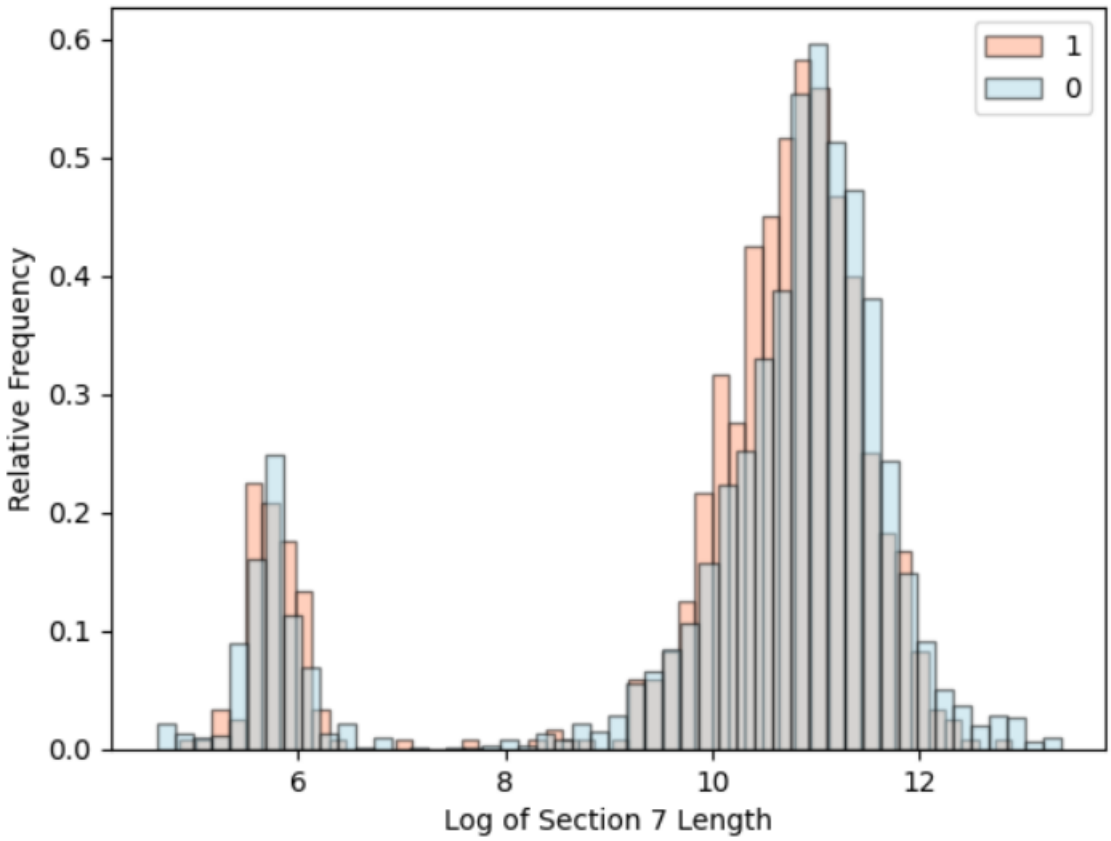

Figure B.2: Box plots of the log length of the 10-K sections 7 and 7A.

To better understand the structure of the data, we also analyzed the frequency distribution of the logarithm of the length of the MD&A section in the two classes. Figure B.2 appeared to reveal no significant differences between the distributions of the classes, although the Kolmogorov-Smirnov test for a 95% significance level resulted in the rejection of the null hypothesis regarding the similarity of distributions. On the one hand, there seems to be statistical evidence that the distributions between classes in the same set are different, the same was also observed between the training and test sets.

Appendix C – Implementation Details

This project was conducted in two distinct environments, one local and another corresponding to *Google Colab* since some tasks took too long to complete locally.

Locally, the system was equipped with an 11th Generation Intel Core i7- 11800H processor, complemented by 16 GB of RAM and featured a Nvidia RTX 3050 Ti Graphics card, running on a 64-Bit operating system. The software used to execute the code and implement all the models was the Windows 11 version of Visual Studio Code.

On *Google Colab*, different types of Virtual Machines were used as different processing needs were required by each of the methods. L4 GPU (22.5 GB of GPU and 53 GB of system RAM) was used for the more demanding LLMs.

Appendix D – Modeling

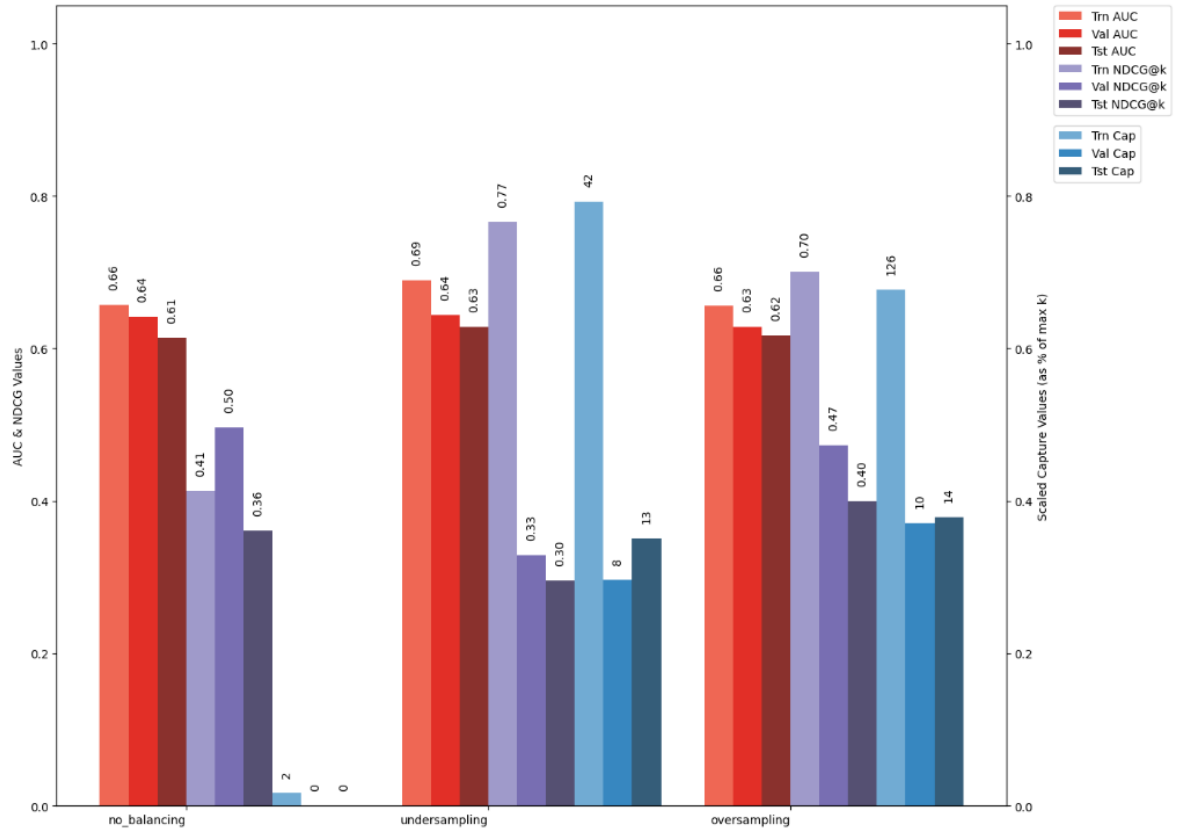

Figure D.1: Logistic Regression results using exclusively financial variables.

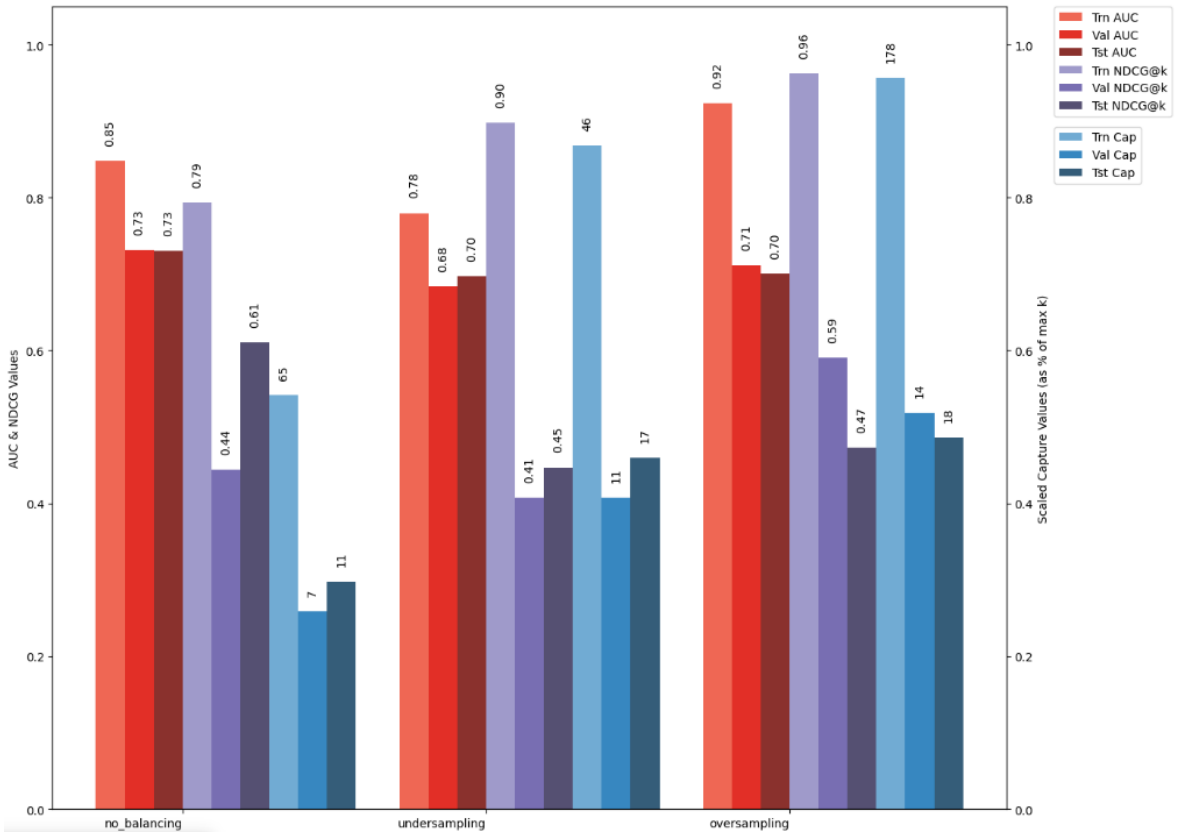

Figure D.2: Logistic Regression results using exclusively textual variables.

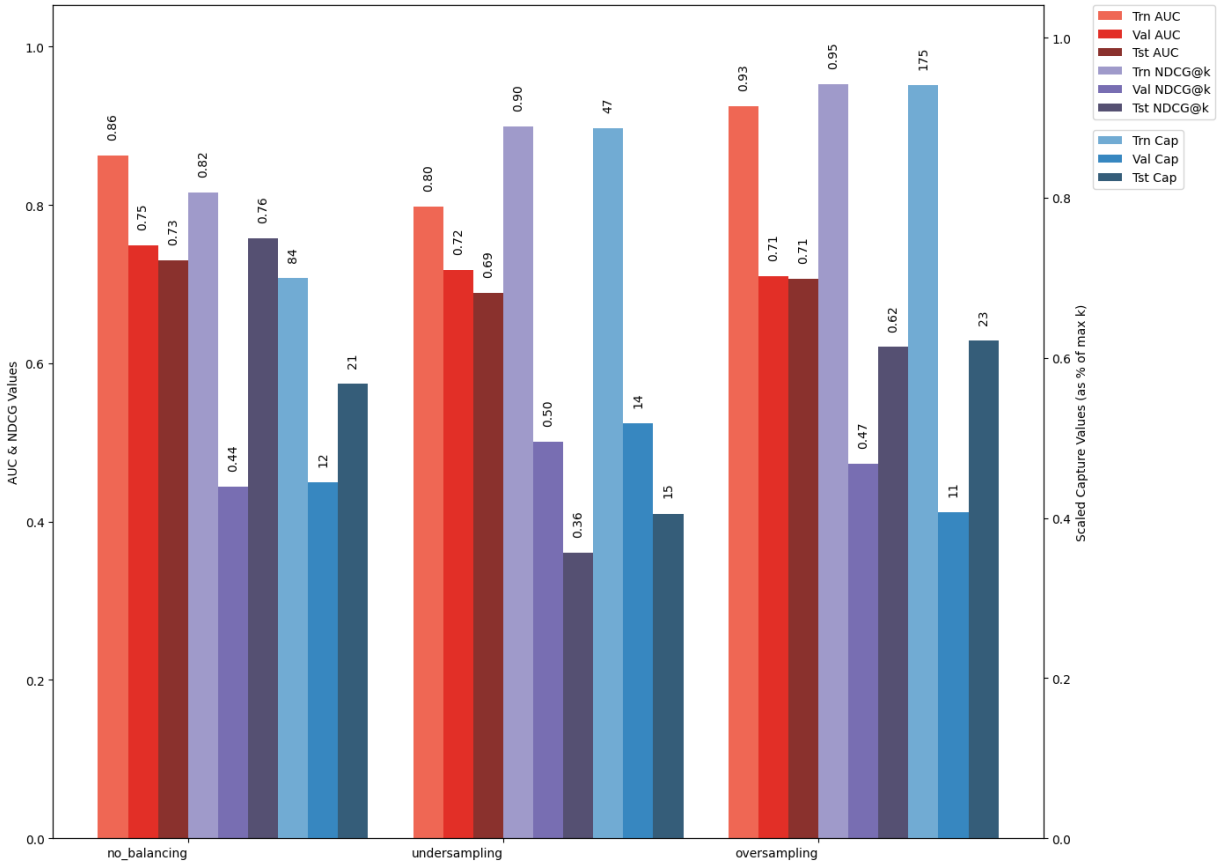

Figure D.3: Logistic Regression results using both financial and textual variables.

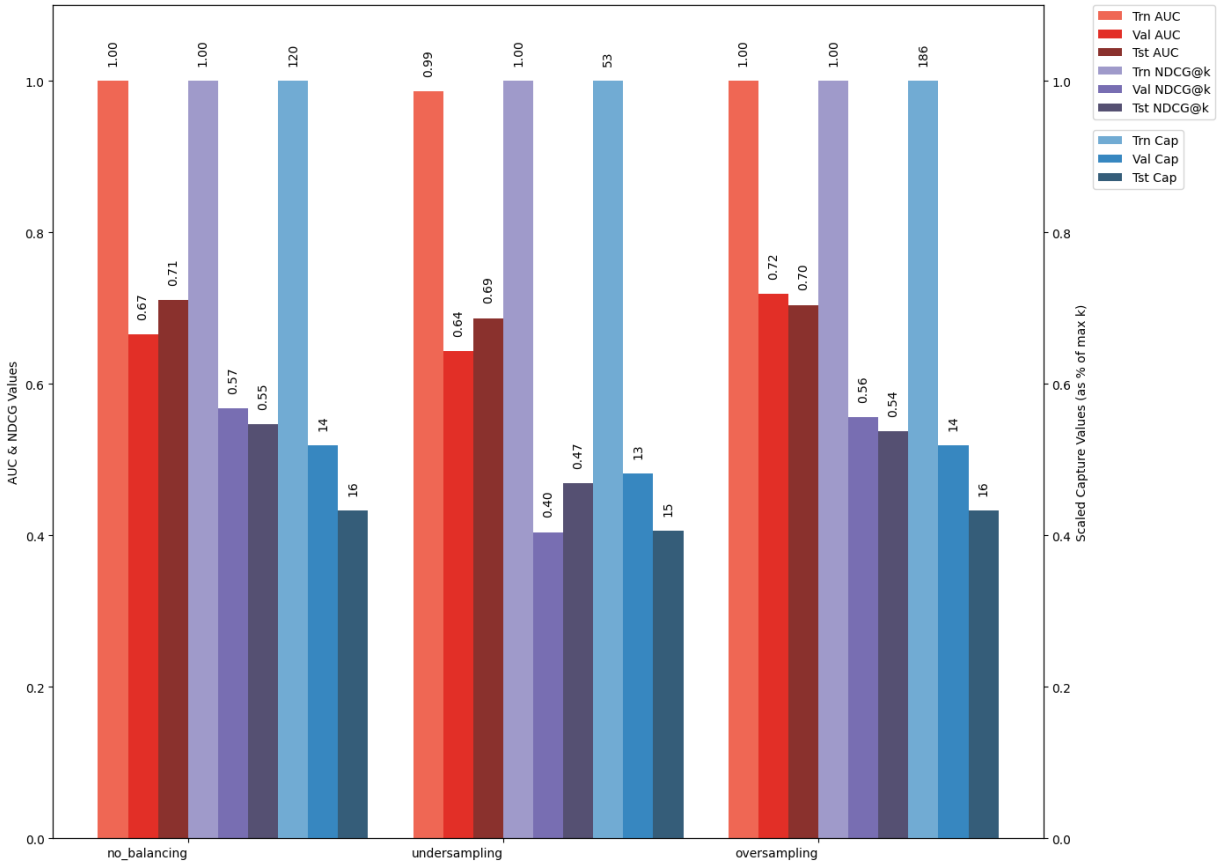

Figure D.4: XGBoost results using exclusively financial variables.

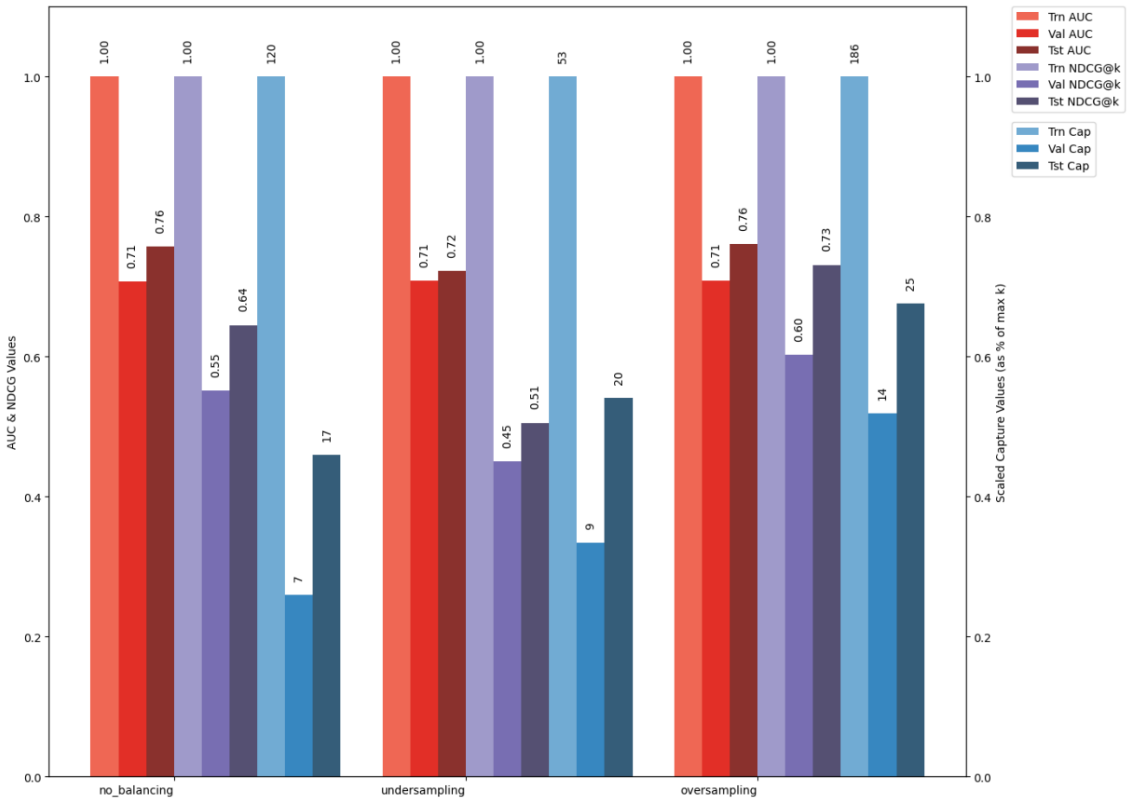

Figure D.5: XGBoost results using exclusively textual variables.

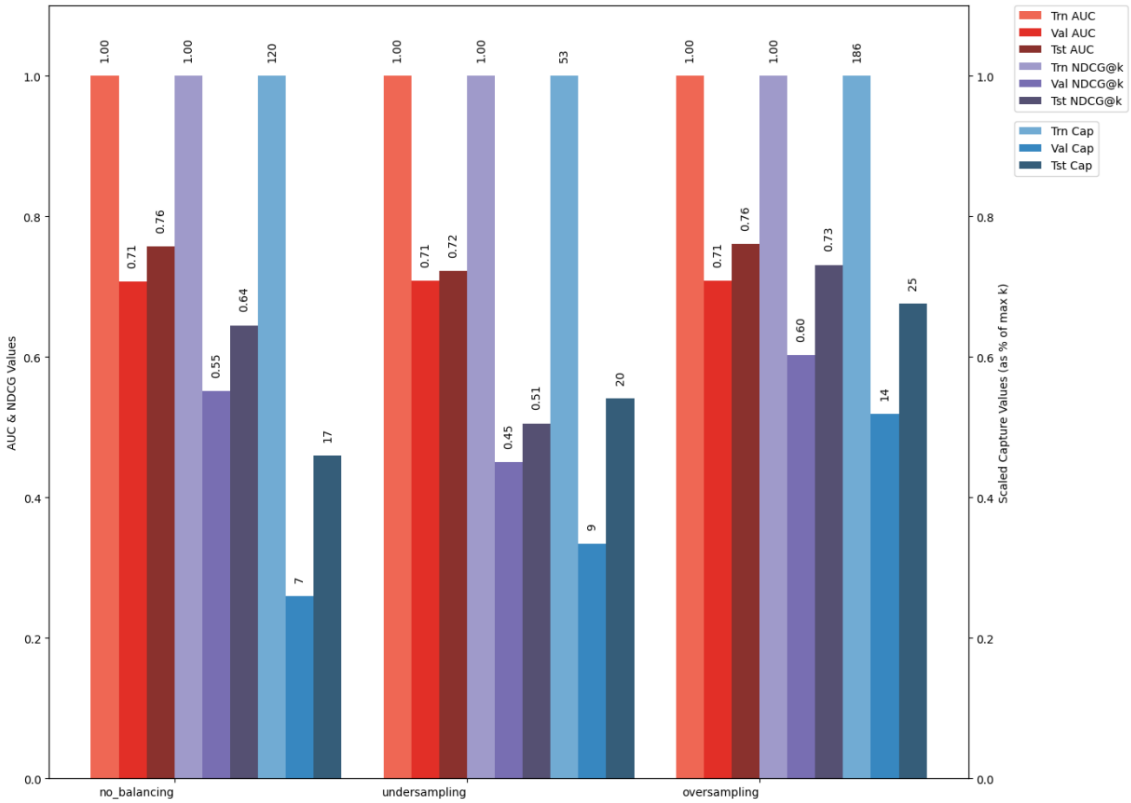

Figure D.6: XGBoost results for using both financial and textual variables.

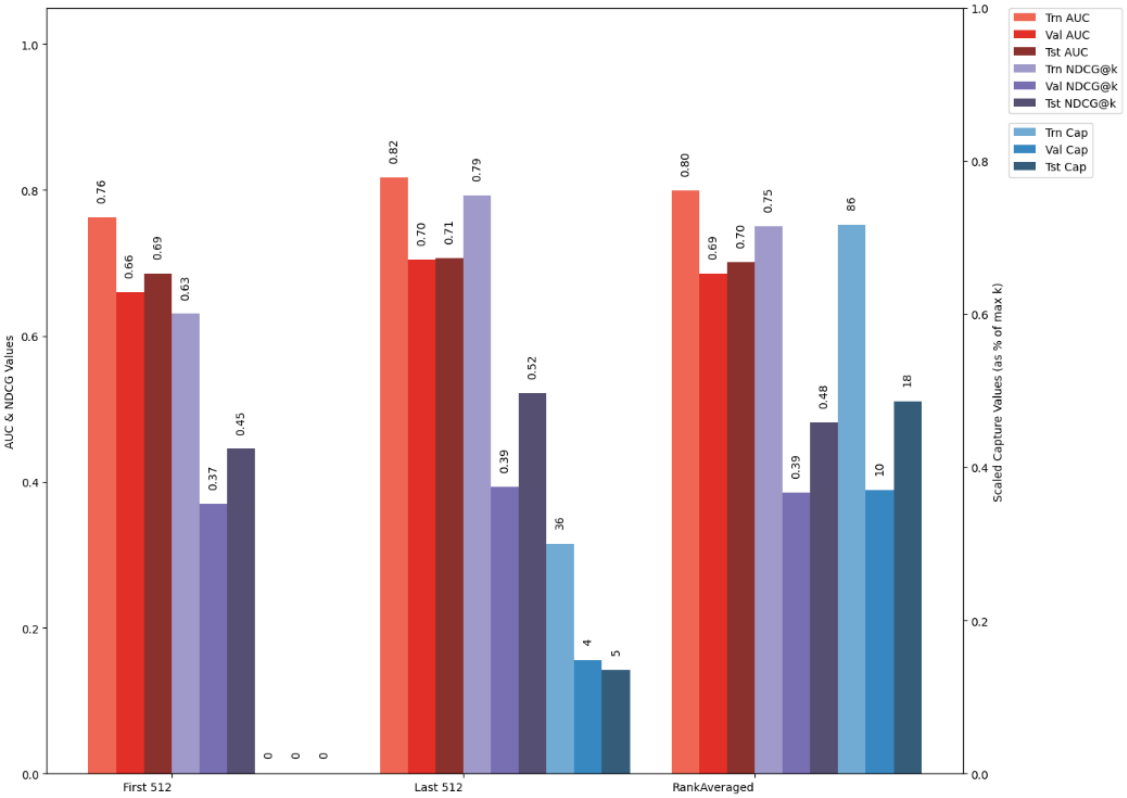

Figure D.7: BERT (first), (last) and (first+last) rank-averaged results using exclusively textual variables over the train, validation and test sets with no balancing.

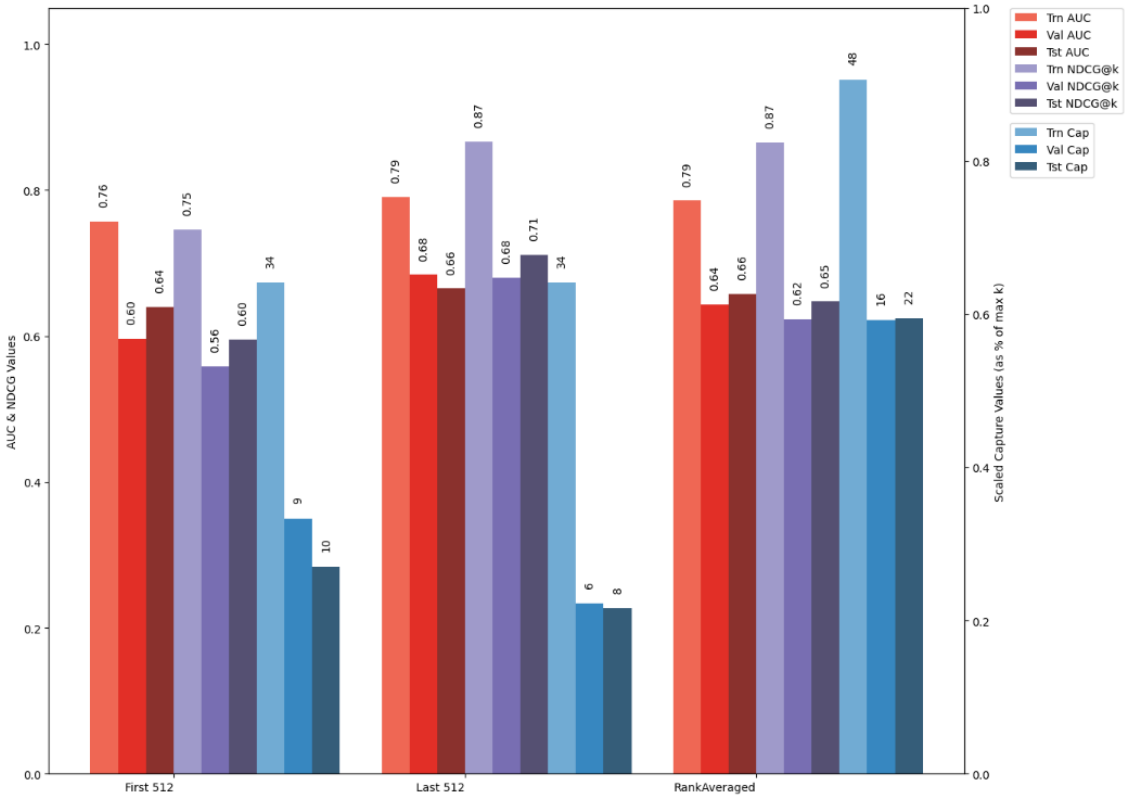

Figure D.8: BERT (first), (last) and (first+last) rank-averaged results using exclusively textual variables over the train, validation and test sets with undersampling.

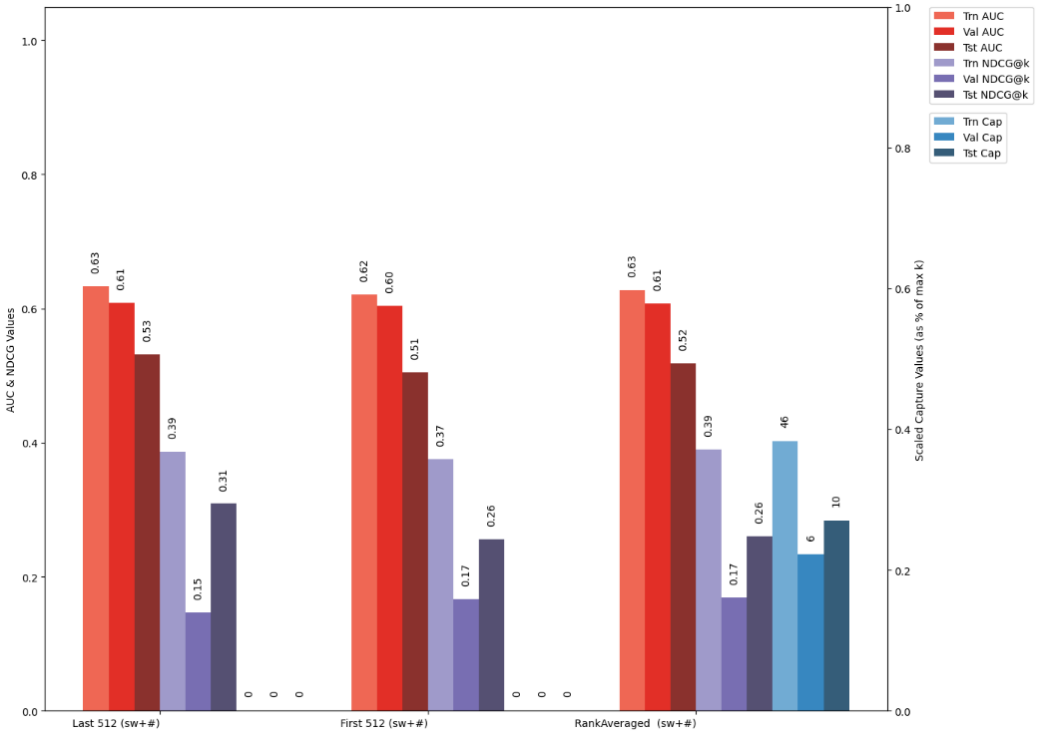

Figure D.9: BERT (first), (last) and (first+last) rank-averaged results using exclusively textual variables obtained from text where stopwords and numbers were removed over the train, validation and test sets with no balancing.

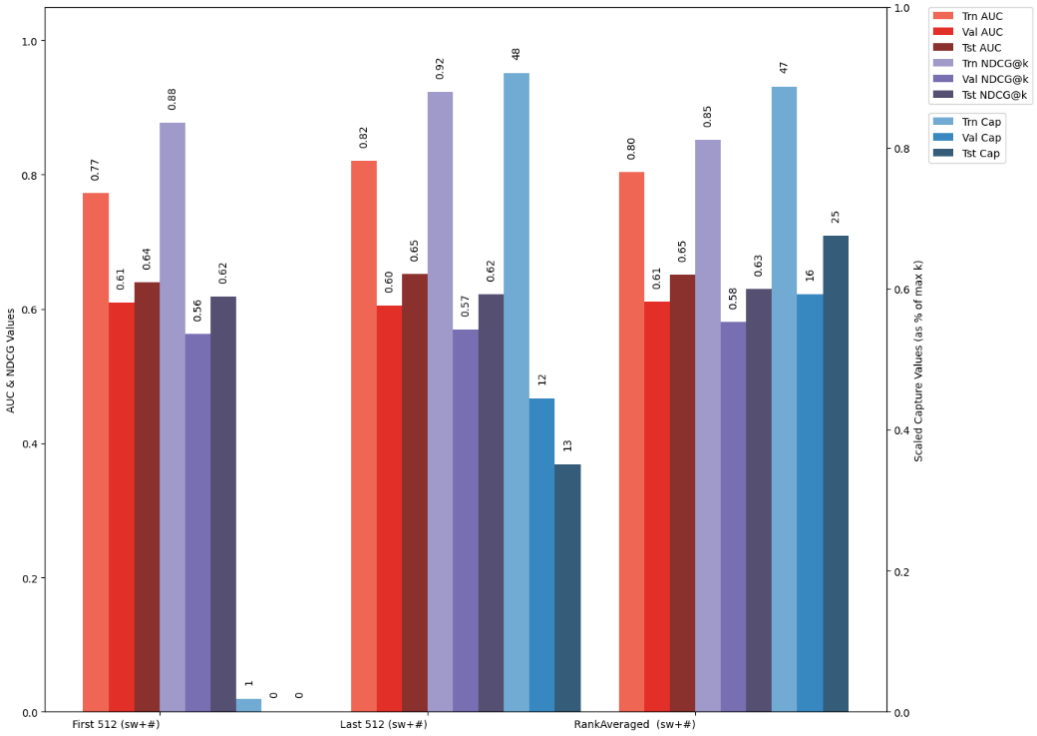

Figure D.10: BERT (first), (last) and (first+last) rank-averaged results using exclusively textual variables obtained from text where stopwords and numbers were removed over the train, validation and test sets with undersampling.

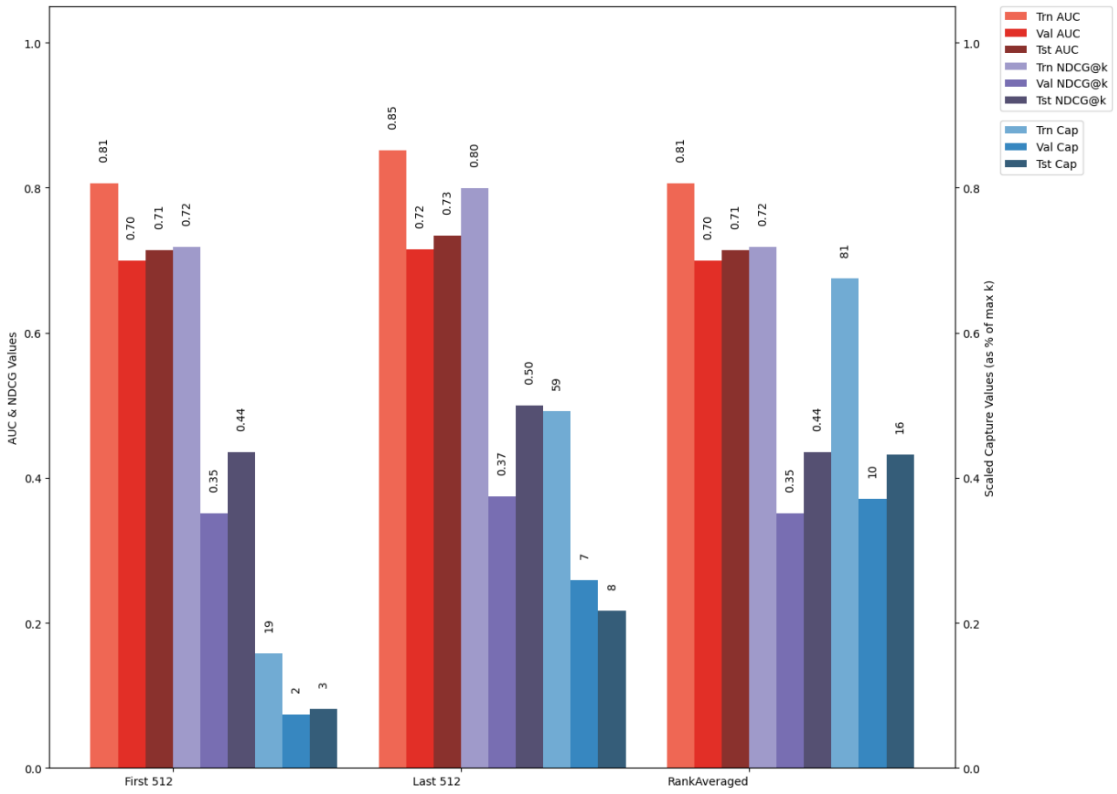

Figure D.11: FinBERT (first), (last) and (first+last) rank-averaged results using exclusively textual variables with no balancing.

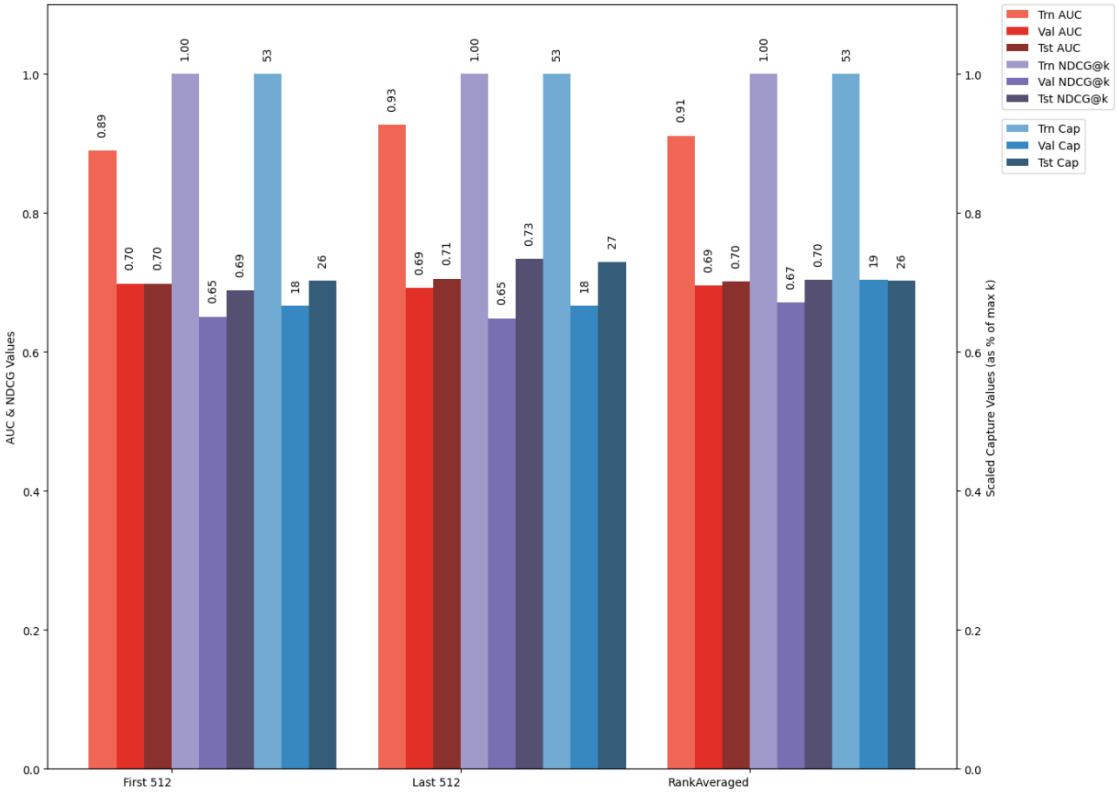

Figure D.12: FinBERT (first), (last) and (first+last) rank-averaged results using exclusively textual variables with undersampling.

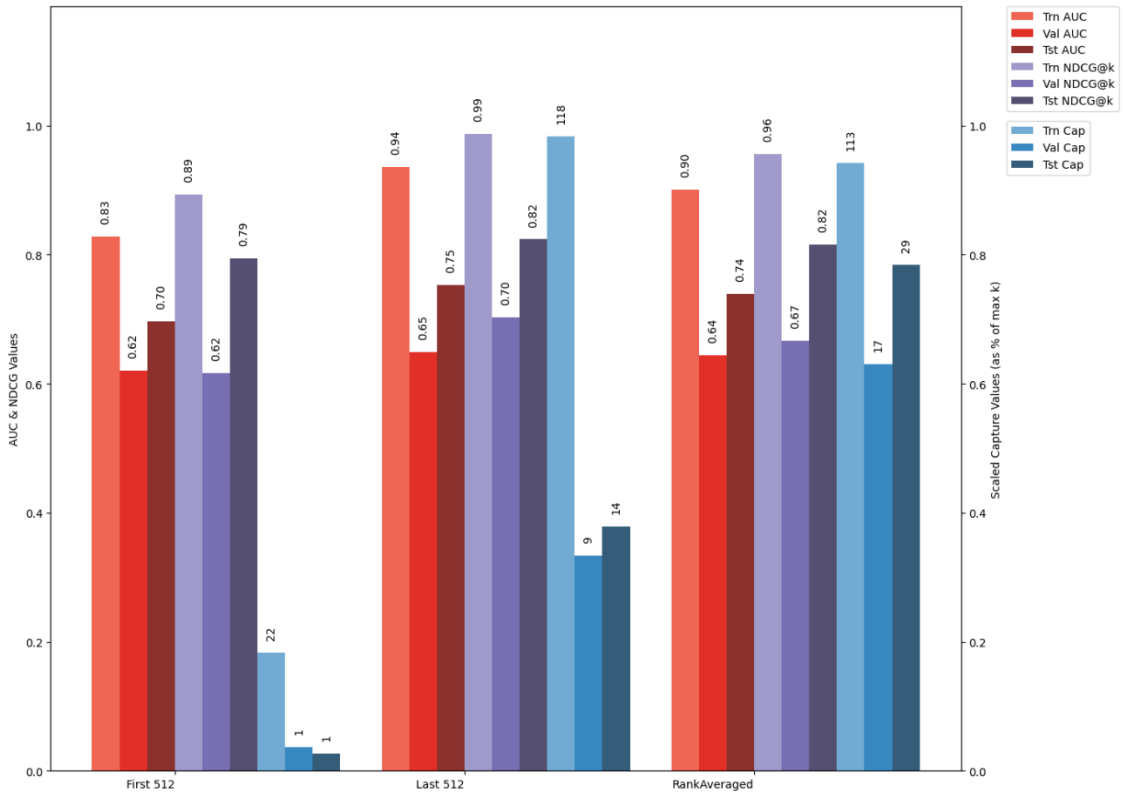

Figure D.13: FinBERT (first), (last) and (first+last) rank-averaged results using exclusively textual variables where stopwords and numbers were removed with no balancing.

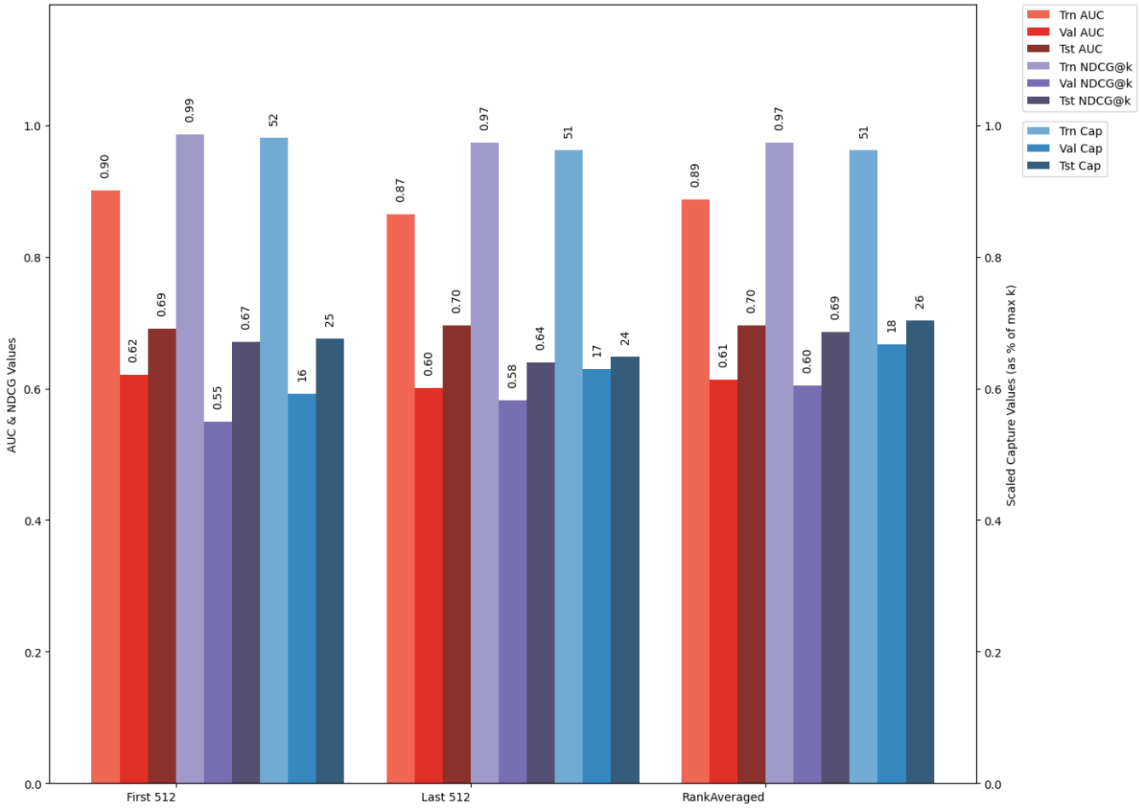

Figure D.14: FinBERT (first), (last) and (first+last) rank-averaged results using exclusively textual variables where stopwords and numbers were removed with undersampling.

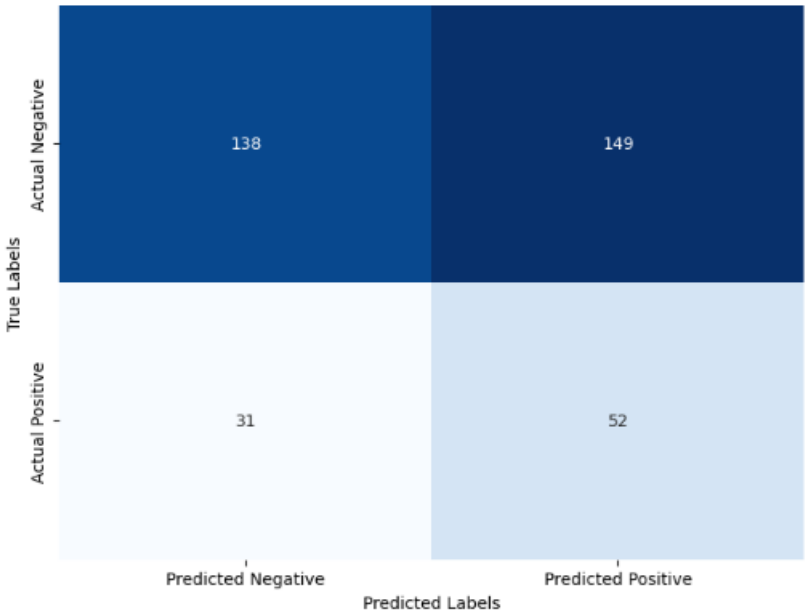

Figure D.15: Confusion Matrix of the rank-averaged BERT (first+last) + LR model with no balancing over the test set.

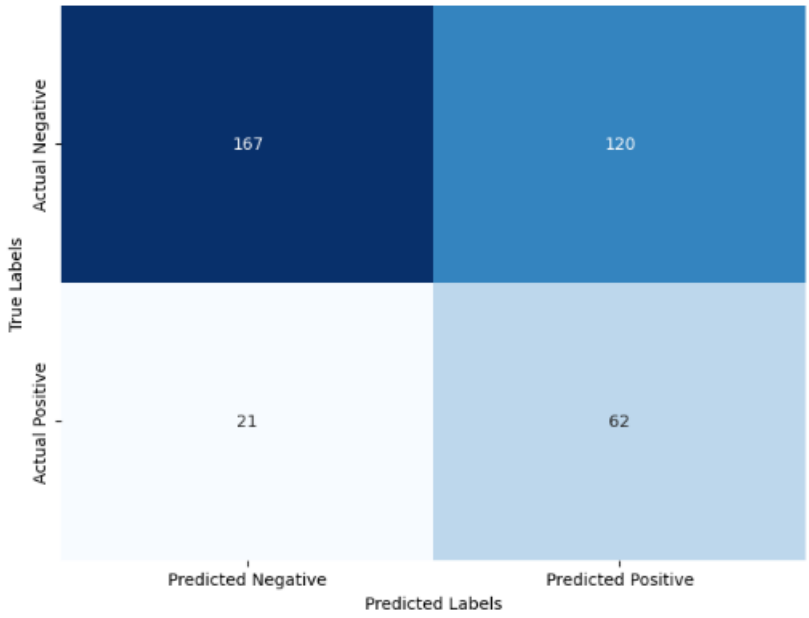

Figure D.16: Confusion Matrix of the rank-averaged FinBERT (first+last) + LR model with no balancing over the test sets.