Natural Language Processing in Finance: Applications and Opportunities

JEAN LEE

Doctor of Philosophy

Supervisor: Dr. Caren Han, Dr. Josiah Poon

A thesis submitted in fulfilment of
the requirements for the degree of
Doctor of Philosophy

This research reported in this thesis was supported by the award of a
Research Training Program scholarship to the PhD Candidate.

School of Computer Science
Faculty of Engineering
The University of Sydney
Australia

30 August 2024

Statement of Originality

This is to certify that to the best of my knowledge, the content of this thesis is my own work. This thesis has not been submitted for any degree or other purposes. I certify that the intellectual content of this thesis is the product of my own work and that all the assistance received in preparing this thesis and sources have been acknowledged.

Abstract

The research of Natural Language Processing (NLP) in Finance has experienced considerable development driven by academia and industry. Recently, the rapid growth of Large Language Models (LLMs) and their robust capabilities have accelerated real-world applications in finance. However, small benchmark datasets in financial NLP often yield lower performance in real-world scenarios. Additionally, data raises privacy concerns and often requires multimodal understanding. Moreover, financial NLP research typically focuses on single tasks, whereas real-world applications demand the integration of multiple tasks. To address these challenges, this research explores various financial NLP applications through three distinct segments. The first segment focuses on an interpretable multi-component NLP system, aiming to identify how NLP components can assist end-users in analyzing financial documents and news. This system integrates various NLP components into a web application, including sentiment analysis, topic modelling, prediction, explanation, and summarization. The second segment focuses on a novel emotion dataset in the stock market, extending beyond traditional financial sentiment classification and demonstrating the importance of emotions in market behaviour. The dataset's usability is shown through empirical analysis, sentiment/emotion classification, and its potential for market forecasting. The third segment focuses on a multimodal financial document understanding and document QA system. This complex system integrates several deep learning models to address tasks, such as intent classification and slot filling, layout analysis, key information extraction, and Retrieval Augmented Generation (RAG) using LLMs. This approach enables comprehensive financial document analysis that combines text, images, and tables. Additionally, this research provides a comprehensive review of LLMs in Finance (FinLLMs) and discusses opportunities and challenges in practical applications. By addressing the challenges of interpretability, multimodal data, and document understanding, this research aims to enhance financial decision-making processes through advanced NLP.

Acknowledgements

I would like to express my highest gratitude to my supervisor, Dr. Caren Han, who made this work possible. Her guidance, expert advice, and encouragement have been invaluable throughout my entire PhD journey.

I also would like to express my sincere appreciation to Dr. Josiah Poon for his care, constructive feedback, and valuable suggestions throughout the research process.

Deep gratitude goes to my parents and my grandmother for their unconditional love and unwavering support. Their prayers have been a great source of strength and wisdom.

Special thanks to my partner for his understanding, patience, and encouragement. Your love, kind words, and care packages helped me through some of my darkest days. Thank you from the bottom of my heart.

I am also thankful to my friends and the members of the USYD NLP/AD-NLP group for their support and encouragement through the highs and lows of this challenging endeavor.

To all who played a role, whether big or small, in this academic journey, I am sincerely grateful for your contributions.

Last but not least, I am truly thankful to God for His love, grace, and blessings throughout my entire life.

(This research reported in this thesis was supported by the award of a Research Training Program scholarship to the PhD Candidate.)

Authorship Attribution Statement

- **Chapter 2** of this thesis includes the publication material of [83], which is submitted to the Neural Computing and Applications and accepted with minor revision.
  I am the first author of this paper. I formulated the research aim and scope of the survey, collected the previous papers, summarized the previous work, analyzed the data and the experiment results, and wrote the whole paper.
- **Chapter 3** of this thesis is published as [84].
  I am the first author of this paper. I formulated the research aim, collected the data, designed the methodology, analyzed the data, conducted the experiments, and wrote the whole paper.
- **Chapter 4** of this thesis is published as [86].
  I am the first author of this paper. I formulated the research aim, collected the data, designed the methodology, analyzed the data, conducted the experiments, and wrote the whole paper.
- **Chapter 5** of this thesis is under review for submission to the AAAI 2025.
  I am the co-first author of this paper. I formulated the research aim, processed data annotation, checked the data quality, analyzed the data and the experiment results, and wrote the whole paper.
- **Appendix** of this thesis is published as [85].
  I am the first author of this paper. I formulated the research aim, reviewed the previous work, checked the data quality, analyzed the data and the experiment results, and wrote most of the paper.

In addition to the statements above, in cases where I am not the corresponding author of a published item, permission to include the published material has been granted by the corresponding author.

Name: Jean Lee
Date: 30 August 2024

As a supervisor for the candidature upon which this thesis is based, I can confirm that the authorship attribution statements above are correct.

Name: Dr. Caren Han
Date: 30 August 2024

Name: Dr. Josiah Poon
Date: 30 August 2024

Publication List

Lee, J., Youn, H. L., Stevens, N., Poon, J., & Han, S. C. (2021, July). "Fednlp: an interpretable nlp system to decode federal reserve communications." In Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval (pp. 2560-2564), **(SIGIR 2021 A* Rank)**.

Lee, J., Youn, H. L., Poon, J., & Han, S. C. (2023). "StockEmotions: Discover Investor Emotions for Financial Sentiment Analysis and Multivariate Time Series." **AAAI-23 Bridge** (AI for Financial Services).

Lee, J., Stevens, N., Han, S. C., & Song, M. (2024). "A Survey of Large Language Models in Finance (FinLLMs)." *accepted with minor revisions to* **Neural Computing and Applications** (H-Index 130).

Lee, J., Lim, T., Lee, H., Jo, B., Kim, Y., Yoon, H., & Han, S. C. (2022, October). "K-MHaS: A Multi-label Hate Speech Detection Dataset in Korean Online News Comment." In Proceedings of the 29th International Conference on Computational Linguistics. (pp. 3530-3538), **(COLING 2022 A Rank)**.

Weld, H., Huang, G., **Lee, J.**, Zhang, T., Wang, K., Guo, X., ... & Han, C. (2021, August). "CONDA: a CONtextual Dual-Annotated dataset for in-game toxicity understanding and detection." In Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021 (pp. 2406-2416), **(ACL 2021 A* Rank)**.

Contents

List of Figures

CHAPTER 1

Introduction

1.1 Background

Natural Language Processing (NLP) in Finance is a growing research area and has attracted considerable attention from both academics and industry. Researchers are exploring the adoption of NLP techniques to a diverse range of financial downstream tasks, including financial sentiment analysis, text classification, question answering, stock market prediction, and document understanding. Concurrently, the financial services industry is actively investigating the implementation of NLP applications to address real-world challenges.

In finance, the majority of market participants aim to accurately forecast asset values and analyse public sentiment to make value-generated investment decisions. However, this is a challenging problem, given that financial markets are typically volatile and influenced by numerous factors. To a large extent, prior research in finance primarily concentrated on the application of time-series modeling and prediction techniques utilizing historical numerical pricing data. With recent advancements in NLP, there is an increasing opportunity to leverage diverse sources of data, including unstructured textual data from financial news, reports, earnings call transcripts, and social media. More recently, deep learning methods for NLP have become more common, with numerous research highlighting the application of neural networks for text-driven stock market prediction tasks.

Furthermore, the emergence of Large Language Models (LLMs) has significantly transformed the research landscape and accelerated the development of LLM applications, such as ChatGPT (from OpenAI) or Gemini (from Google). LLMs have demonstrated robust capabilities

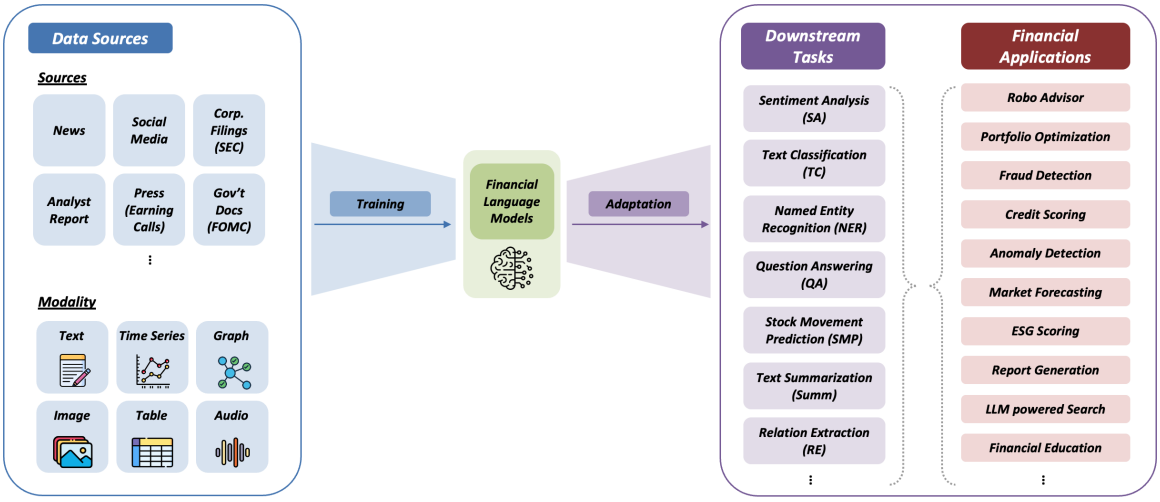


FIGURE 1.1: Language Models in Finance, trained on a diverse range of data sources and adapted for Financial NLP tasks and Financial applications.

across diverse NLP tasks and further facilitated the adaptation of various forms of multimodal data (e.g., text, images, speech, and tabular data) across AI and interdisciplinary research communities. The rapid progress of general-domain LLMs has stimulated the evolution of financial-domain LLMs, by employing methods such as mixed-domain LLMs with prompt engineering and instruction fine-tuned LLMs with prompt engineering. Nevertheless, the field of financial LLMs [95] is in its infancy, while general LLMs are extensively researched and reviewed [206, 194, 17, 97, 204].

The development of pre-training models and the multimodal capabilities of LLMs facilitates the implementation of complex financial applications, thereby expanding opportunities within the finance sector. For example, there is a growing interest in Intelligence Document Processing (IDP) services that can automatically extract data from business documents and process the relevant data into actionable information. Automated data entry in bookkeeping and extracting content from scanned receipts represent common instances of real-world NLP applications in finance.

1.2 Research Problem and Aim

Although research on NLP in finance has shown progress, the focus remains largely on conventional tasks with limited dataset availability. Moreover, as models are primarily derived from the general domain, their adaptability in the financial context lags behind the overall progress in general-domain NLP research. In addition, a substantial knowledge gap exists between NLP research and the financial industry in terms of datasets, tasks, and real-world applications.

For financial NLP **datasets**, benchmark datasets are not comprehensively researched, have limited availability, and are mostly small in scale. Models developed on these benchmark datasets may show good performance, but the model capability tends to fall when applied to real-world data. Furthermore, dealing with real-world data demands extensive effort in data pre-processing. In the case of financial documents, which often contain sensitive information, careful handling is essential. Additionally, the financial jargon causes a barrier since most pre-trained datasets predominantly consist of general text, lacking specific financial terms.

For financial NLP **tasks**, current research often focuses on one single task, whereas real-world applications require the integration of multiple tasks within a system. In addition, while conventional tasks such as financial sentiment analysis or stock movement prediction are extensively researched, a broader spectrum of financial NLP tasks is yet to be thoroughly explored.

For financial NLP **applications**, it is rare to find pilot studies demonstrating the application of NLP techniques to solve real-world cases. Although numerous commercial services exist, the detailed implementations of NLP applications are not widely shared.

Given the relatively unexplored progress of financial NLP compared to general domain research, this study pioneered the research direction of employing **deep learning-based feature extraction and information fusion for finance NLP applications**. To address this objective, the thesis answers the following three research questions:

- What financial NLP applications can be developed? How can we design the system architecture to handle multiple tasks?
- For this, what datasets, conventional and advanced NLP tasks and techniques can be considered?
- What is the best multimodal feature extraction and integration (text, time series, tables, charts) approach for the financial visually-rich document understanding?

Specifically, this research works on three variants of financial NLP applications: 1) FedNLP Application, 2) StockEmotions Application, and 3) FinDoc Application. FedNLP Application is an interpretable multi-component NLP system that aims to decode Federal Reserve communications that assist end-users. StockEmotions Application is a new dataset for financial sentiment, emotion classification, and stock market forecasting. FinDoc Application is a financial document understanding and document QA system that integrates the capabilities of LLM and Retrieval-Augmented Generation (RAG). This research also explores a comprehensive review of Financial LLMs, including benchmark datasets, downstream tasks, and financial applications. In each chapter, three main research questions are addressed in detail, including inquiries as follows:

- What types of benchmark datasets exist, and are these datasets suitable for training models for real-world applications? What financial NLP tasks have been explored, and are there any advanced tasks worth considering? [Chapter 2]
- What financial applications can be developed to demonstrate the utility of various NLP components? What factors should be considered when building real-world applications in finance? [Chapter 3]
- What financial applications can be developed in conjunction with financial sentiment analysis and market prediction tasks? Can the combination of textual and emotional features enhance market prediction? [Chapter 4]
- How to develop financial applications that can handle the multimodal data in the internal financial data repository? What advanced techniques are required for financial document understanding tasks? [Chapter 5]

- What opportunities and challenges should be considered when using financial LLMs in practical applications? [Chapter 6]

1.3 Contribution

To investigate the research questions, this research breaks new ground in the field of deep learning-based natural language understanding/analysis for finance by exploring three financial NLP segments. Each segment leverages the collection of real-world multimodal data and demonstrates the application of conventional and advanced NLP tasks in real-world financial scenarios, aiming to enhance financial decision-making processes through advanced NLP.

(1) **LLMs in finance (FinLLMs)**:
   (a) provides a comprehensive review of FinLLMs that explores the evolution from general-domain LMs to financial-domain LMs;
   (b) compares five techniques used across four FinPLMs and four FinLLMs, including training methods and data, and instruction fine-tuning methods;
   (c) summarizes the performance evaluation of benchmark tasks and datasets between different models and provides advanced financial NLP tasks and datasets.

(2) **Interpretable NLP System in Finance (FedNLP Application)**:
   (a) proposes FedNLP, the first interpretable multi-component NLP system for decoding Federal Reserve communications that assist end-users;
   (b) conducts an extensive study to predict the changes in the target Federal Funds Rate (FFR) using several models;
   (c) conducts a pilot study of human evaluation to determine which system components assist end-users in understanding the Fed documents;
   (d) publishes a demo to facilitate the development of the system.

(3) **Beyond Financial Sentiment Analysis (StockEmotions Application)**:
   (a) introduces StockEmotions, a financial-domain-focused dataset for financial sentiment/emotion classification and stock market time series prediction;

(b) applies a multi-step annotation pipeline that brings the collaboration of human and pre-trained language model;

(c) demonstrates the dataset usability through downstream tasks and the impact of investor emotions for time series forecasting in particular.

(4) **Financial Document Understanding (FinDoc Application)**:

(a) introduces the first Financial Document Understanding and DocQA system that integrates the capabilities of LLM and Retrieval-Augmented Generation (RAG);

(b) demonstrates advanced techniques implemented throughout the entire system architecture, including Intent Detection and Slot Filling, Top-k Search Algorithms, Layout Analysis, and Key Information Extraction;

(c) conducts a multimodal model that enables handling text, layout, and visual features using a cross-modality encoder, managing unstructured documents in PDF format.

1.4 Thesis Outline and Research Workflow

Chapter 2 is an extended work published as ***A Survey of Large Language Models in Finance (FinLLMs)*** [83]. This work was submitted to Neural Computing and Applications and accepted with minor revisions. This literature review includes the first comprehensive survey of Large Language Models in Finance that explores the evolution from general-domain LMs to financial-domain LMs. This work summarizes the performance evaluation of six benchmark tasks and provides eight advanced financial NLP tasks and over 30 datasets for the development of advanced models in finance.

Chapter 3 is an extended work published as ***FedNLP: an Interpretable NLP System to Decode Federal Reserve Communications*** [84], accepted to the ACM SIGIR 2021. This work focuses on an interpretable multi-component NLP system. The objective is to identify how NLP components can assist end-users in analyzing documents and speeches related to the Federal Open Market Committee (FOMC). In this demo system, various NLP components,

such as sentiment analysis, topic modelling, prediction, explanation, and summarization, are integrated into a single application.

Chapter 4 is an extended work published as ***StockEmotions: Discover Investor Emotions for Financial Sentiment Analysis and Multivariate Time Series*** [86], accepted to AAAI-23 Bridge. This work focuses on emotion classification in the stock market, extending beyond conventional financial sentiment classification. Inspired by behavioral finance, this work presents a dataset comprising 10,000 sentences collected from StockTwits. It includes two financial sentiment classes, and twelve emotion classes by leveraging both a pre-trained language model (PLM) and finance experts. This work demonstrates the dataset usability through the Sentiment/Emotion classification and Market Forecasting tasks.

Chapter 5 is the work being prepared as ***FinDoc: Financial Document Understanding and Document QA System***. This work is under review for submission to the AAAI 2025. This chapter focuses on a multimodal Financial Document Understanding and Document Question Answering (DocQA) system by exploring the best multimodal feature extraction and integration (text, time series, tables, charts) approach for the financial visually-rich document understanding. This complex system integrates several distinct deep learning models to address a variety of tasks, such as Intent Classification and Slot Filling, Layout Analysis, Key Information Extraction, and Retrieval Augmented Generation (RAG) using LLMs. This work demonstrates the system in a web environment, offering a replicable financial application in Intelligent Document Processing.

Chapter 6 concludes the findings of this research and discusses future works. This chapter discusses the potential opportunities and challenges of NLP in finance. It addresses key concerns such as hallucinations, privacy implications, and efficiency limitations to foster a comprehensive understanding of the practical deployment of financial NLP research.

CHAPTER 2

Literature review

This chapter presents a literature review of financial NLP research, emphasizing LLMs in finance and encompassing a wide range of financial NLP tasks and datasets. This literature review is an extension of the work ***A Survey of Large Language Models in Finance*** [83], which is accepted with minor revisions at Neural Computing and Applications. I formulated the research aim and scope of the survey, collected the previous papers, summarised the previous work, analyzed the data and the experiment results, and wrote the whole paper.

Large language models (LLMs) such as ChatGPT have demonstrated remarkable capabilities in handling various Natural Language Processing (NLP) tasks, attracting significant attention across diverse domains, including financial services. Despite the extensive research into general-domain LLMs and their immense potential in finance, Financial LLMs (FinLLMs) research remains limited. This research provides a comprehensive review of Large Language Models in Finance, including their history, underlying techniques, associated downstream tasks and datasets, evaluations, and prospective directions for researchers and practitioners. In addition to FinLLMs, this chapter explores a broader range of NLP tasks within the financial domain. Firstly, this research presents a chronological overview of general-domain Pre-trained Language Models (PLMs) through current FinLLMs, including the GPT series, selected open-source LLMs, and financial LMs. Secondly, this study compares five techniques used across eight financial LMs, including training methods, training data, and fine-tuning methods. Thirdly, this research summarizes the performance evaluations of six benchmark tasks and datasets and provides eight advanced financial NLP tasks and datasets for developing more sophisticated FinLLMs.

2.1 Overview

Research into Large Language Models (LLMs) has developed rapidly in both academia and industry, with notable attention to LLM applications such as ChatGPT. Inspired by Pre-trained Language Models (PLMs) [36, 136], LLMs are empowered by transfer learning and built upon the Transformer [168] architecture using large-scale textual corpora. They have demonstrated robust capabilities in diverse Natural Language Processing (NLP) tasks, even suggesting potential for Artificial General Intelligence (AGI) [13]. Researchers have discovered that scaling models [137] to sufficient sizes not only enhances model capacity but also enables surprising emergent properties such as in-context learning [12]. These properties were neither specifically trained for nor observed in small-scale language models (e.g., BERT [36], GPT-1 [136]). Following the research community's consensus [206, 17], this research adopts the common definition of Large Language Models (LLMs) as Pre-trained Language Models (PLMs) with over 7 billion parameters, while those with fewer parameters continue to be referred to as PLMs. However, I note that the specific parameter threshold can vary across the research community. The technical evolution of LLMs has resulted in a remarkable level of homogenization [11], which means that a single model could yield strong performance across a wide range of tasks. The capability of LLMs has facilitated the adaptation of various forms of multimodal data (e.g., text, images, speech, and tabular data) and multimodal models across AI and interdisciplinary research communities.

In the financial domain, there has been growing interest in leveraging NLP technologies for various financial tasks, including sentiment analysis, question answering, and stock market prediction. The rapid advancement of general-domain LLMs has spurred investigations into financial LLMs (FinLLMs), employing methods such as mixed-domain LLMs with prompt engineering and instruction fine-tuned LLMs with prompt engineering.

While general LLMs are extensively researched and reviewed [206, 194, 204], the field of financial LLMs [95] is at an early stage. Considering the immense potential of LLMs in finance, this survey provides a holistic overview of FinLLMs and discusses future directions that

can stimulate interdisciplinary studies. The key contributions of this chapter are summarized below.

- To the best of my knowledge, this is the first comprehensive survey of FinLLMs that explores the evolution from general-domain LMs to financial-domain LMs.
- This research compares five techniques used across financial PLMs and financial LLMs, including training methods, data, and instruction fine-tuning methods.
- This research summarizes the performance evaluation of six benchmark tasks and datasets between different models. For the development of advanced FinLLMs, this chapter provides eight advanced financial NLP tasks over 30 datasets.

2.1.1 Survey Methodology

To ensure a comprehensive identification of available Large Language Models in finance, this study utilized interdisciplinary databases and extensive search strings. The search covered works from Google Scholar, Scopus, DBLP, and refereed journals such as Elsevier, Springer, Wiley, MDPI, Frontiers, and Sage. I employed relevant keywords, including "Large Language Models", "Language Models" AND "Finance" or "financial", spanning from January 2018 to current. The search string used for Scopus was: (TITLE-ABS-KEY ("Large Language Models") OR TITLE-ABS-KEY ("Language Models") OR TITLE-ABS-KEY ("GPT") AND TITLE-ABS-KEY ("financ*") AND PUBYEAR > 2018 AND PUBYEAR < current).

This research initiated the exploration by prioritizing peer-reviewed articles from reputable publishers. The selection criteria gave preference to the latest state-of-the-art algorithms relevant to this study, which led to the inclusion of specific preprint papers. Given the primary focus of this chapter on a comprehensive survey of FinLLMs, I thoughtfully integrated general LLMs to ensure a holistic view. To ensure thorough coverage, this study analyzed existing research on general LLMs and included publications that were significantly cited in recent studies. While this chapter is about LLMs in finance, it's important to note that specific financial research related to blockchain is beyond the scope of this chapter. An overview of the literature review is presented in Figure 2.1.

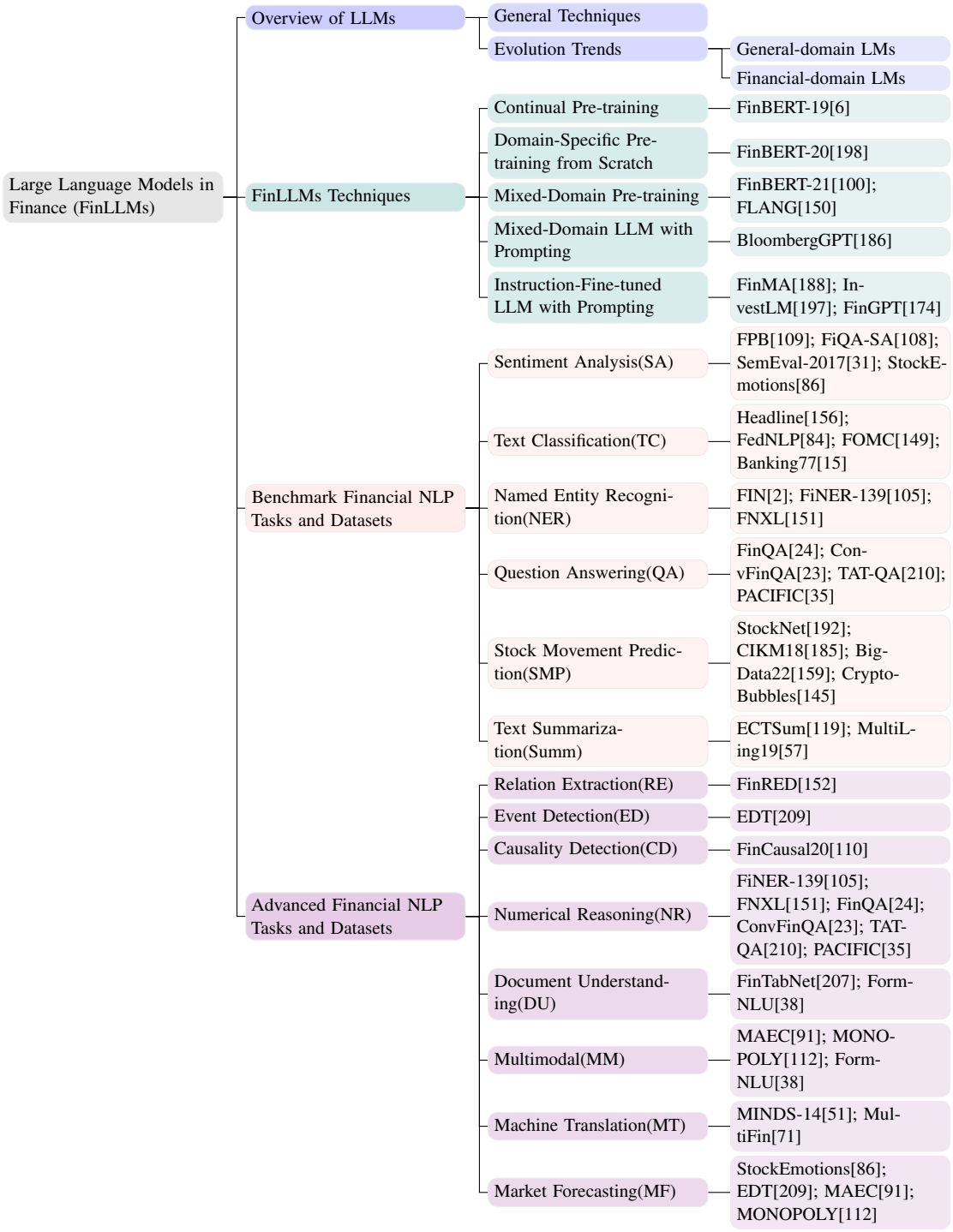


FIGURE 2.1: Overview of the literature review

2.2 Background of LLMs

What are Large Language Models? Large Language Models (LLMs) are a type of artificial intelligence (AI) algorithm that contains a billion to a trillion parameters, which are typically pre-trained on massive text data through self-supervised learning. LLMs exhibit strong capacities to understand natural language and excel in a diverse range of tasks involving generative abilities and human interaction. Notable examples include OpenAI's GPT models (e.g., GPT-3.5 used in ChatGPT), Google's PaLM (used in Gemini), and Meta's LLaMa.

How do they work? On a technical level, LLMs are enabled by transfer learning and scale. The idea of transfer learning is to take the knowledge learned from one task and apply it to another task. Within deep learning, pre-training is the dominant approach to transfer learning: a model is trained on a surrogate task and then adapted to the downstream task of interest via fine-tuning. In self-supervised learning, the pre-training task is derived automatically from unannotated data. For example, the language model works by taking an input text to predict the next token or word in a sentence, given its surrounding context.

How do they evolve? Significant advancements have been made in the domain of self-supervised learning. In 2017, the Google Brain team proposed the Transformer architecture, which heavily relies on the parallel multi-head attention mechanism, enabling the scaling up of models to larger sizes. This breakthrough has paved the way for the development of pre-trained language models (PLMs), including the generative pre-trained transformers [136] (GPTs, introduced in 2018) and BERT [36] (Bidirectional Encoder Representations from Transformers, introduced in 2019). After 2019, self-supervised learning with language models solidified its position in NLP. Researchers have discovered that scaling models to sufficient sizes not only enhances model capacity but also enables surprising emergent properties such as in-context learning. These properties were neither specifically trained for nor observed in small-scale language models (e.g., BERT). To categorise language models based on their parameter scales, the research community has named the term "Large Language Models (LLM)" for the PLMs of substantial size, typically exceeding 7 billion parameters.

LLMs are currently applied not only in NLP but also in computer vision, audio and multi-modal processing (e.g., text, images, speech, and tabular data). In 2022, researchers introduced the concept of "Foundation Models" [11] to describe AI models that surpass language-centric capabilities. These models are trained on extensive multi-modal datasets, enabling them to proficiently handle diverse downstream tasks and deployment.

Why is understanding LLMs important? LLMs have demonstrated remarkable capabilities, yet we are still in the early days of exploring their full potential. Grasping their mechanisms is crucial for harnessing their potential across a wide array of applications and domains. In the context of deploying LLMs in practical settings, a profound understanding of LLMs allows for effective customization and fine-tuning, potentially enhancing their performance in specific domains or tasks. With the continued evolution of LLMs, comprehending their underlying technology is vital not only for researchers but also for interdisciplinary practitioners. To have a quick understanding, the following section introduces the foundational background of LLMs, encompassing key techniques and the evolutionary trajectory from general LLMs to financial LLMs.

2.2.1 Scaling Laws

The scaling laws in AI indicate that scaling up language models improves the model performance, or to put it simply: a larger model is better. In January 2020, the OpenAI team [75] conducted empirical research on scaling laws for models based on the Transformer architecture. This study showed that the model's performance is mostly impacted by scale and comprises three factors: model size (N: Number of parameters), training Dataset size (D), and computing Cost of training (C). When each individual factor is not bottlenecked by the other two, the evaluation of performance after training (L: cross-entropy loss) exhibits a power-law relationship, and this relationship spans more than six orders of magnitude.

The application of Scaling Laws led to the introduction of GPT-3 in July 2020 by the OpenAI team [12] with a more detailed explanation provided in the subsequent "In-Context Learning" section. In March 2022, the DeepMind team [60] conducted further investigations into scaling

laws. Their research showed that to achieve compute-optimal LLMs, the model size (N) and training tokens (D) should be scaled equally under a given compute budget (C). These results differ from OpenAI's earlier research, which indicated that model size (N) should be scaled faster than the dataset size (D).

2.2.2 Emergence and Homogenization

Scaling laws continue to drive the progress of LLMs, and have shown two major characteristics: emergence and homogenization [11]. Emergence refers to the unforeseen and implicitly induced capabilities that can appear, which were not explicitly trained but emerge as models are scaled up. Examples of emergent properties include a deeper understanding of context, enhanced language generation, and even reasoning or problem-solving abilities. Homogenization refers to the model's consistent and robust performance across a wide variety of tasks and domains, demonstrating a certain level of universality in its application. This can simplify deployment and usage since the same model can be used across a diverse range of tasks with some task-specific modifications using fine-tuning or prompt tuning.

Homogenization and emergence combine in a potentially problematic way. Homogenization has the potential to yield significant benefits across various domains, particularly those with limited task-specific data like finance or healthcare. However, this approach also results in blindly inheriting any flaws present in the model across all adapted models. Given that the emergence of LLMs comes from implicit capabilities rather than explicit training, understanding the nature or quality of LLMs is challenging, and they may exhibit unexpected failure modes. Emergence introduces considerable uncertainty regarding both the capabilities and imperfections of LLMs. As a result, deploying LLMs through aggressive homogenization needs significantly more rigorous testing and scrutiny.

2.3 General LLM Techniques

LLMs can be assumed to be a "generalist" or an "intermediate" model. In order to meet the specific needs, the model needs to be "tuned". This section discusses several tuning methods,

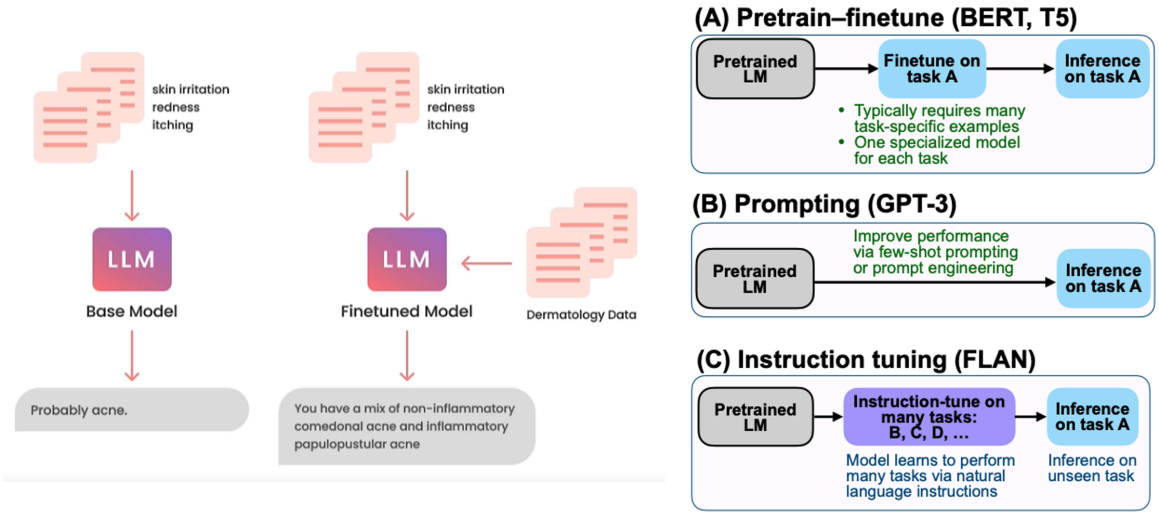


FIGURE 2.2: Comparison of the Pre-train and fine-tune, Prompting, and Instruction tuning Techniques [180].

such as (full) fine-tuning, Parameter-Efficient Fine-Tuning (PEFT), Prompting, Instruction Tuning, Reinforcement Learning from Human Feedback (RLHF), and Retrieval-Augmented Generation (RAG).

2.3.1 Fine-tuning

Fine-tuning is a process that involves further training an existing pre-trained model on a smaller, task-specific, labelled dataset. In this way, it is possible to adjust some of the model parameters to optimize its performance for a particular task or set of tasks. In full fine-tuning, all the model parameters are updated, making it similar to pre-training but with a much smaller and labelled dataset.

Advantages of full fine-tuning: Full fine-tuning can be effective, even with relatively small task-specific datasets. The fine-tuning process primarily involves adjusting the model's knowledge to the specifics of the new data. This is especially vital in areas with specialized jargon, concepts, or structures, such as legal documents, medical texts, or financial reports. As a result, when encountering unseen examples from a specific domain or task, the fine-tuned model is likely to make predictions or generate outputs with higher accuracy and robustness.

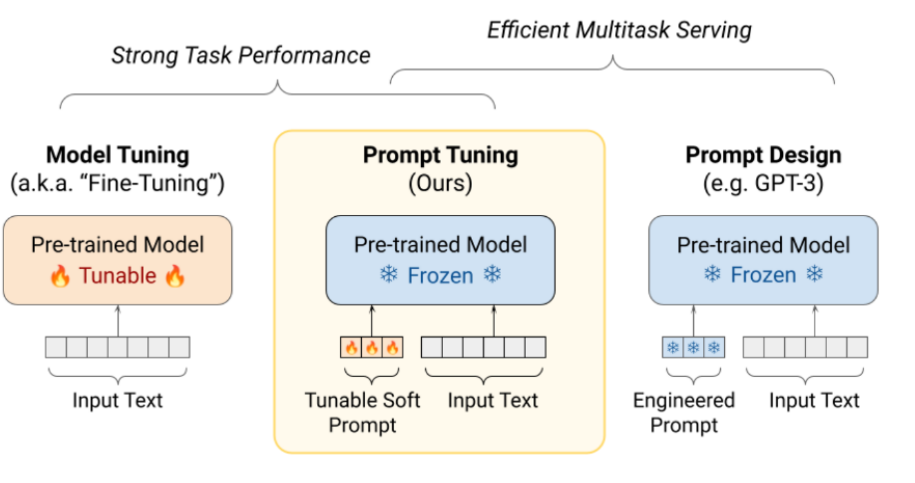


FIGURE 2.3: Comparison of Prompting. [1]

Disadvantages of full fine-tuning: Full fine-tuning involves updating all the parameters of a large model, demanding substantial computational resources. Even with relatively small fine-tuning datasets, the number of tokens can be huge and expensive to compute. Working with large models can require specialized hardware, such as high-performance GPUs or TPUs, with substantial memory capacities. Depending on the model's size and the dataset, the duration of the fine-tuning process can vary widely, ranging from hours to weeks.

2.3.2 Prompt Engineering

Training large language models is very time-consuming and compute-intensive. As models continue to scale in size, there is growing interest in more efficient training methods such as prompting [97]. Prompting involves a frozen pre-trained model for a specific downstream task by including a text prompt that describes the desired task or demonstrates an example of the task. With prompting, it is possible to eliminate the need to train a separate model for each downstream task, allowing for the reuse of a single, frozen pre-trained model. This is not only simpler but also significantly more efficient, as training and storing a smaller set of prompt parameters is far less resource-intensive than training and storing all the model's parameters. There are two categories of prompting methods:

[1] https://blog.research.google/2022/02/guiding-frozen-language-models-with.html

- **hard prompts** (a.k.a. **prompt engineering**) are manually handcrafted text prompts with discrete input tokens; the downside is that creating good prompts requires a lot of effort. (e.g. In-context Learning [12], Chain-of-Thought [179])
- **soft prompts** (a.k.a. **prompt tuning**) are learnable tensors concatenated with the input embeddings that can be optimized to a dataset; the downside is that these tensors are not human readable because these "virtual tokens" are not matched to the embeddings of a real word. (e.g. prefix tuning [93], P-tuning [98])

In-context learning (Few-shot Prompting)

The standard few-shot prompting strategy was introduced with GPT-3 [12]. This approach involves crafting a prompt that includes a few text-based demonstrations illustrating the desired task, while zero-shot prompting typically relies on an instruction describing the task without including any additional examples. Few-shot prompting demonstrations are typically encoded as input–output pairs. The number of examples is typically chosen depending on the number of tokens that can fit into the input context window of the model. Few-shot performance appears to be an emergent ability for many tasks.

Chain-of-Thought (CoT)

Chain of Thought (CoT)[179] involves augmenting each few-shot example in a prompt with a step-by-step breakdown of the reasoning process that leads to the final answer. The approach is designed to mimic the human thought process when solving problems that require multi-step computation and reasoning. By incorporating CoT prompting, sufficiently large LLMs can demonstrate enhanced reasoning capabilities, significantly improving performance on tasks such as mathematical problems. Moreover, the emergence of CoT reasoning appears to be an emergent ability of LLMs. CoT prompting has been used to achieve state-of-the-art LLM performance on several benchmarks.

Prompt Tuning (or Soft Prompt)

Soft prompting [88] is a straightforward and computationally efficient method to adapt LLMs to specific downstream tasks, particularly tasks with limited data. The approach involves the learning of **soft prompt** vectors through backpropagation while maintaining the rest of the LLM parameters frozen. This enables the flexible reuse of a single model across several

tasks. Soft prompts can be contrasted with the discrete, text-based('hard') few-shot prompts commonly associated with LLMs like GPT-3.5. While soft prompting can benefit from any number of labelled examples, typically, only a few examples are needed to achieve satisfactory performance. Moreover, it was demonstrated that soft-prompt-tuned model performance could attain performance levels comparable to end-to-end fine-tuning as the model size increases.

2.3.3 Instruction Tuning

Instruction Tuning has been introduced to improve the capabilities and controllability of LLMs. This approach involves additional training of LLMs using explicit instructions, resulting in a notable zero-shot learning performance across unseen tasks [180]. Research on instruction tuning can be broadly classified into two areas [204]: 1) construction of instruction datasets and 2) generation of fine-tuned LLMs using these datasets and existing LLMs.

Instruction Datasets

An instruction dataset can be constructed by combining existing annotated NLP datasets. Datasets such as Flan [180] are constructed based on the data integration strategy. The fundamental elements are "Instruction, Output" pairs, with Context and Inputs being optional components.

- **Instruction**: A natural language text sequence that defines a specific task. (e.g. "determine the sentiment of the sentence.")
- **Context**: Additional information or external context that can guide the model toward better responses. (e.g. positive example, negative example, definition, and constraints)
- **Input**: The specific question or input for which a response is sought (e.g. "He likes the cat." Is positive or negative?)
- **Output**: The expected type or format of the anticipated output based on the instruction and input. (e.g. "Positive")

An alternate way to rapidly generate desired outputs for given instructions is to employ LLMs such as GPT-3.5-Turbo or GPT4, rather than manually collecting the outputs. Instructions

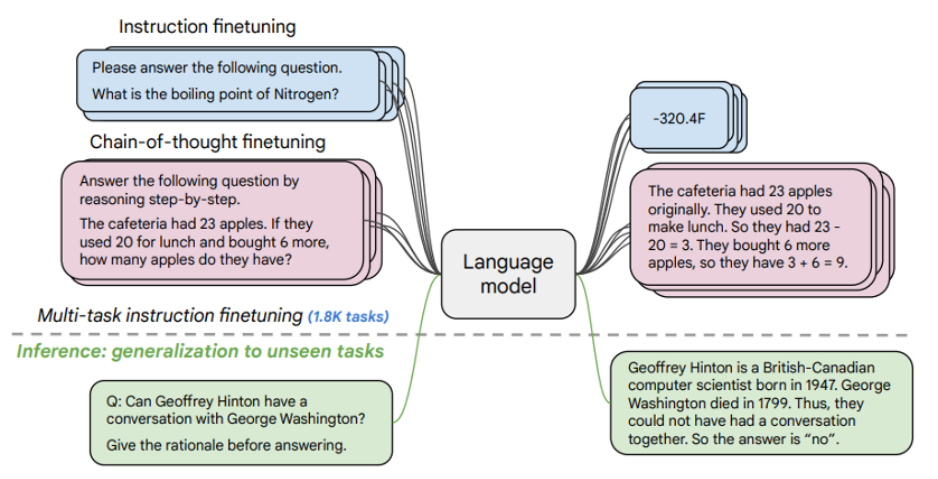


FIGURE 2.4: An Overview Framework of Instruction fine-tuned LLM [27]

can be either manually collected or expanded from a small set of handwritten seed instructions using LLMs. Subsequently, the collected instructions are fed to LLMs to obtain the corresponding outputs.

Instruction fine-tuned LLMs

By leveraging the collected Instruction Tuning dataset, a pre-trained model can be directly fine-tuned with full supervision. In this process, the model is trained to predict each token in the output sequentially, based on the given instruction and input. Notable instruction fine-tuned LLMs include InstructGPT [127](developed by OpenAI), BLOOMZ [118], Flan-T5 [27] (by Google), Alpaca (by Stanford University), and Vicuna 13B (by UC Berkeley). From a technical perspective, Reinforcement Learning with Human Feedback (RLHF) is applied in InstructGPT, which uses GPT-3 as its foundational model. Overall, InstructGPT outperforms GPT-3, showing the effectiveness of RLHF techniques on instruction fine-tuned LLMs.

2.3.4 Reinforcement Learning from Human Feedback (RLHF)

Reinforcement Learning with Human Feedback (RLHF) [127] enhances traditional Reinforcement Learning (RL) techniques by integrating human feedback to guide the reward structure. In traditional RL, a language model learns decision-making by interacting with an environment, proposing actions, and receiving feedback in the form of rewards or punishments.

However, in many real-world scenarios, it can be challenging or too expensive to define a precise reward structure that aligns with the task's goals. Human feedback (HF) can help address some of these challenges. RLHF involves obtaining feedback from human evaluators to guide the model's learning process. The process typically involves the following steps:

(1) **Supervised fine-tuning**: An initial model is trained on the human-filtered instruction dataset using supervised learning or another available method. The model then interacts with the environment, and for each interaction, a set of actions is proposed.
(2) **Training a reward model based on human feedback**: By sampling multiple responses for one instruction, human evaluators rank them from the best to the worst.
(3) **Updating Model**: The feedback is used to update the model via fine-tuning the model's parameters with new instructions (step 1) to align with the human-provided feedback (step2). Parameters can be updated using the proximal policy optimization (PPO) method, a policy gradient reinforcement learning method.
(4) **Iteration**: Steps 1 to 3 are repeated multiple times until the model performance does not significantly improve.

2.3.5 Parameter-efficient Fine-tuning (PEFT)

Parameter-efficient fine-tuning (PEFT) [96] employs techniques to further tune a pre-trained model by updating only a small number of its total parameters, rather than fine-tuning all the model's parameters. PEFT methods significantly reduce computational and storage costs by focusing on a small number of model parameters, which may even include additional parameters. Recent state-of-the-art PEFT techniques achieve performance comparable to full fine-tuning. PEFT methods vary in their strategies for determining which components of the model are trainable. Some approaches prioritize training specific parts of the original model's parameters, while others integrate and train smaller additional components without modifying the original model structure, such as Low-Rank Adaptation (LoRA).

LoRA [64] is a technique that accelerates the fine-tuning of large models while reducing memory consumption. To make fine-tuning more efficient, LoRA represents weight updates

with two new smaller matrices, known as update matrices, through low-rank decomposition. These new matrices can be trained to adapt to new data while minimizing the overall number of changes. The original weight matrix remains frozen and does not undergo any further adjustments. The final results are obtained by combining the original and adapted weights. The advantages of LoRA include:

- **Task switching efficiency**: Creating different versions of the model for specific tasks becomes easier. A single copy of the pre-trained weights is stored, while multiple small LoRA matrices are created. When switching between tasks, only the matrices need to be replaced, significantly reducing storage requirements.
- **Fewer GPUs required**: LoRA reduces GPU memory requirements by up to 3x, as gradients do not need to be retrained for most parameters.
- **High accuracy**: On a range of evaluation benchmarks, LoRA has demonstrated performance nearly equivalent to full fine-tuning.

2.3.6 Retrieval-Augmented Generation (RAG)

For more complex and knowledge-intensive tasks, Retrieval Augmented Generation (RAG) [89] is proposed, which combines prompt engineering with context retrieval from external knowledge sources to improve the performance and relevance of LLMs. RAG integrates information retrieval mechanisms with text generation models. The information retrieval component extracts relevant contextual information from an external data source, such as a database, and the text generation model uses this added context to produce a more accurate and "knowledgeable" response. Implementing RAG in an LLM-based system has two main benefits:

- **Reliable source data:** This ensures that the model has access to the most current and accurate information.
- **Trustworthiness:** This provides end-users with access to the model's sources, allowing them to verify the accuracy of its claims and build trust in its output.

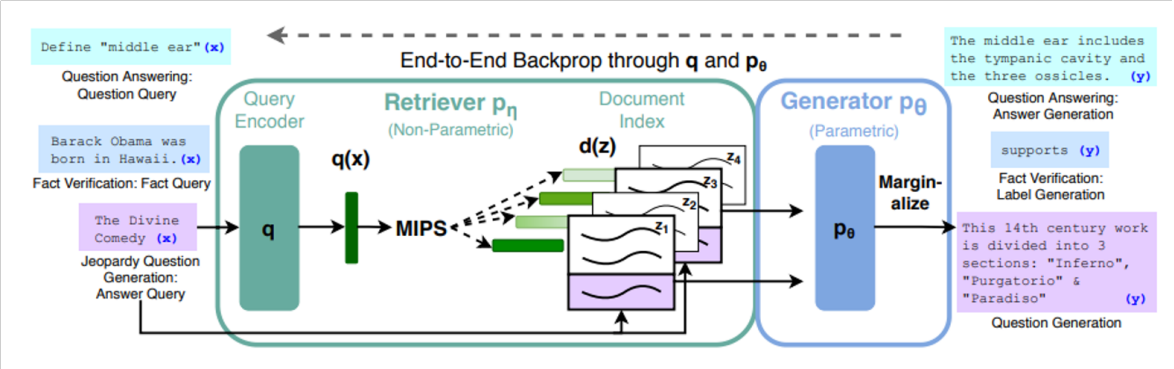


FIGURE 2.5: An Overview of Retrieval-Augmented Generation (RAG) Framework [89].

In recent developments, retriever-based approaches have gained prominence and are increasingly integrated with popular LLMs like ChatGPT to enhance capabilities and ensure factual consistency.

2.4 Evolution Trends: from General LMs to Financial LM

2.4.1 General-domain LMs

Since the introduction of the Transformer [168] architecture by Google in 2017, Language Models (LMs) are generally pre-trained with either discriminative or generative objectives. Discriminative pre-training, which involves predicting the next sentence using a masked language model, typically employs an encoder-only or encoder-decoder architecture. The Bidirectional Encoder Representations from Transformers (**BERT**)[36] is a pre-trained language model that employs a discriminative objective, establishing a foundational role in LM research. Generative pre-training, on the other hand, involves predicting the next token using autoregressive language modelling and typically features a decoder-only architecture. OpenAI has been at the forefront of research in this area, particularly with its Generative Pre-trained Transformer (**GPT**)[136] family of models.

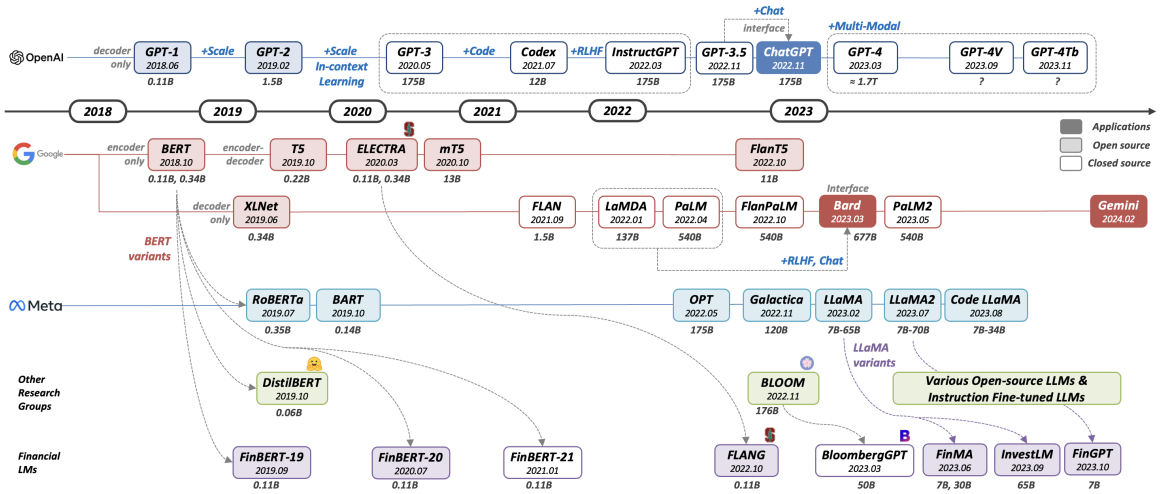


FIGURE 2.6: Timeline showing the evolution of selected PLMs/LLMs releases from the general domain to the financial domain [83].

Figure 2.6 illustrates LMs' evolutionary trends in chronological order, displaying foundational models ranging from PLMs to LLMs and transitioning from general-domain LMs to financial-domain LMs. Given this chapter's focus on financial LLMs, I include a few open-sourced LMs used as backbone models for financial models. For details on other general-purpose LLMs, I refer to various recent survey papers [206, 194].

GPT-Series

The Generative Pre-trained Transformer (GPT) series of models started with **GPT-1** (110M, 2018) [136]. This work shows the fine-tuning results of each specific generative task. Since then, the OpenAI team has focused on scaling the model, and **GPT-2** (1.5B) [137] was released in 2019. GPT-2 identified the power of scaling and a probabilistic approach for multi-task problem-solving. In 2020, **GPT-3** with 175B parameters was released [12]. This was a significant milestone for LLMs, as it introduced an emergent capability of LLMs; in-context learning. In-context learning refers to the model acquiring capabilities that were not explicitly trained, allowing language models to understand human language and produce outcomes beyond their original pre-training objectives. In this research, experiments on zero-shot, one-shot, and few-shot learning were reported, demonstrating that LLMs can successfully perform tasks without any examples (zero-shot) and show improved performance with some examples (few-shot). An important aspect of LLMs is that the natural language prompt does

not update the model's weights (parameters), whereas fine-tuning methods require updating weights for specific purposes [97].

Ongoing efforts to improve LLMs have resulted in the introduction of the LLM-based dialogue application, **ChatGPT**, in November 2022. This application combines GPT-3 (In-context learning), Codex (LLMs for code), and InstructGPT (Reinforcement Learning with Human Feedback, RLHF). Codex [21] is an LLM for programming languages, an extension beyond human language. The Codex model utilises 159 GB of code, enabling code generation. InstructGPT [127] introduces the human alignment method known as Reinforcement Learning with Human Feedback (RLHF), which integrates reinforcement learning (RL) models into the training of LLMs. The answers produced by LLMs are ranked based on human feedback (HF), and this ranking serves as input for reinforcement learning models to enhance the accuracy of the LLM's responses. This additional process to align the model with human feedback is implemented to restrict the generated answers and develop a more responsible AI.

The success of ChatGPT has raised mainstream awareness of the AI capabilities and led to further development of significantly larger models, including **GPT-4** (estimated 1.7T parameters). GPT-4 demonstrates human-level performance, capable of passing the bar and medical exams, handling multi-modal data, including images, videos, and sounds, as well as processing multilingual data. GPT-4 was released in March 2023, followed by GPT-4V in September 2023, GPT-4V Turbo in November 2023, and GPT store in January 2024.

OpenAI continues to develop exceptionally large-sized language models, releasing closed-source LLMs (a.k.a proprietary models) and providing APIs to foster the development of real-world applications. Their focus has a notable impact on both the research community and various industries, including healthcare, education, and finance. In the financial domain, researchers are conducting experiments comparing the performance of financial LLMs built on open-source models to that of ChatGPT and GPT-4. However, the real-world implementation of financial applications utilizing their APIs remains relatively unexplored.

Open-source LLMs

Prior to the era of LLMs, the research community often released open-source PLMs such

as BERT (base-110M parameters) [36]. BERT and its variants subsequently became the dominant PLM research. Other models released by Google Research include open-source PLMs such as **T5** [138] and **ELECTRA** [29], which have encoder-decoder architecture, and **XLNet** [199] which has decoder-only architecture with generative objectives similar to the GPT-series. While generative objective models from Google Research followed the closed-source trend, some open-source LLMs (e.g. FLAN T5 [27]) and datasets (e.g. FLAN collection [102]) still continued to be released. In March 2023, Google released a dialogue application system, **BARD**, to compete with ChatGPT. This application was built based on the closed-source LLMs, LaMDA (137B) [166] and PaLM (540B) [26]. In February 2024, the BARD application was renamed to **Gemini**, focusing on multimodal capability.

Since OpenAI shifted from open-source to closed-source LLMs, the trend across LLM research is a reduction in the release of open-source models. However, in February 2023, Meta AI released the open-source LLM, **LLaMA** (7B, 13B, 33B, 65B parameters) [167], and this encouraged the development of diverse LLMs using LLaMA. Similar to BERT variants, LLaMA variants quickly proliferated by adopting various techniques such as Instruction Fine-Tuning (IFT) [204] and Chain-of-Thought (CoT) Prompting [179]. In the research community, there have also been significant efforts to generate open-source LLMs to reduce the reliance on corporate research and proprietary models. **BLOOM**(176B) [147] was built by a collaboration of hundreds of researchers from the BigScience Workshop. This open-source LLM was trained on 46 natural languages and 13 programming languages.

2.4.2 Financial-domain LMs

Domain-specific LMs, such as financial-domain LMs, are commonly built using general-domain LMs. In finance, there are predominantly four financial PLMs (FinPLMs) and four financial LLMs (FinLLMs). Within the four FinPLMs, **FinBERT-19**[6], **FinBERT-20**[198], and **FinBERT-21** [100] are all based on BERT, while **FLANG** [150] is based on ELECTRA [29]. Within the four FinLLMs, **FinMA** [188], **InvestLM** [197], and **FinGPT** [174] are based on LLaMA or other open-source-based models, while **BloombergGPT** [186] is a BLOOM-style closed-source model.

2.5 FinLLMs Techniques: from FinPLMs to FinLLMs

While this chapter focuses on FinLLMs, it is important to acknowledge that previous studies on FinPLMs as they formed the groundwork for FinLLM development. I reviewed three techniques used by the four FinPLMs and two techniques used by the four FinLLMs and summarized them in Table 2.1, including their pre-training techniques and fine-tuning details.

Within the four FinPLMs, three of the models are from the BERT family of financial PLMs. Despite sharing the same name, each FinBERT model employs a different pre-training technique, pre-training data, and evaluation dataset. The fourth FinPLM is FLANG, which uses ELECTRA as its backbone architecture. The first technique is **continual pre-training** and is used by FinBERT-19 [6]. The second technique is financial **domain-specific pre-training from scratch** and is used by FinBERT-20 [198]. The third technique is **mixed-domain pre-training** and is used by both FinBERT-21 [100] and FLANG [150]. The techniques employed by FinPLMs are all based on weight updates (fine-tuning) of the underlying model. In contrast, LLMs use prompt engineering from the human language of text instruction and input. Additionally, prompt engineering does not require weight updates to the underlying model. The fourth technique reviewed in this research is **mixed-domain LLM with prompt engineering**, used by BloombergGPT [186]. The fifth technique is **instruction fine-tuned LLM with prompt engineering**, used by FinMA [188], InvestLM [197] and FinGPT [174].

2.5.1 Continual Pre-training

Continual Pre-training of LMs aims to train an existing LM with new data on an incremental sequence of tasks [76, 162]. This technique includes continual domain-adaptive pre-training,

[2]The abbreviations correspond to Paras.= Model Parameter Size (Billions); (in Category) PLM = Pretrained Language Model, Disc. = Discriminative, Gen. = Generative; Post-PT = Post-Pre-training, PT = Pre-training, FT = Fine-Tuning, PE = Prompt Engineering, IFT = Instruction Fine-Tuning, PEFT = Parameter Efficient Fine-Tuning; (G) = General domain, (F) = Financial domain; (in Evaluation) [SA] Sentiment Analysis, [TC] Text Classification, [SBD] Structure Boundary Detection, [NER] Named Entity Recognition, [QA] Question Answering, [SMP] Stock Movement Prediction, [Summ] Text Summarization, [RE] Relation Extraction; (in Venue) (S) = Special Track, (D) = Datasets and Benchmarks Track, (W) = Workshop. In open source, it is marked as Y if it is publicly accessible as of Dec 2023.

					Pre-training (PT)		Evaluation		Open Source			
Category	**Model**	**Backbone**	**Paras.**	**Techniques**	**PT Data**	**PT Data Size**	**Task**	**Dataset**	**Model**	**PT**	**IFT**	**Venue**
FinPLM (Disc.)	FinBERT-19 [6]	BERT	0.11B	Post-PT, FT	(G) Wikipedia, BookCorpus (F) Reuters	(G) 3.3B words + [Post-PT] (F) 29M words	[SA]	FPB FiQA-SA	Y	N	N	ArXiv Aug 2019
	FinBERT-20 [198]	BERT	0.11B	PT, FT	(F) SEC, Investext, SeekingAlpha	(F) 4.9B tokens	[SA]	FPB FiQA-SA AnalystTone	Y	Y	N	ArXiv Jul 2020
	FinBERT-21 [100]	BERT	0.11B	PT, FT	(G) Wikipedia, Book. (F) FinancialWeb, YahooFinance, Reddit	(G) 3.3B words + (F) 12B words	[SA] [QA] [SBD]	FPB FiQA-SA FiQA-QA FinSBD19	N	N	N	IJCAI (S) Jan 2021
	FLANG [150]	ELECTRA	0.11B	PT, FT	(G) Wikipedia, Book. (F) SEC, Reuters, Bloomberg, SeekingAlpha, Investopia	(G) 3.3B words + (F) 696k docs	[SA] [TC] [NER] [QA] [SBD]	FPB FiQA-SA Headline FIN FiQA-QA FinSBD21-3	Y	Y	N	EMNLP Oct 2022
FinLLM (Gen.)	BloombergGPT [186]	BLOOM	50B	PT, PE	(G) C4, Wikipedia Github, Pile, etc (F) SEC, Web, News, Filings, Press, Bloomberg (exclusive)	(G) 345B tokens + (F) 363B tokens	[SA] [TC] [NER] [QA]	FPB FiQA-SA Headline FIN ConvFinQA	N	N	N	ArXiv Mar 2023
	FinMA [188]	LLaMA	7B, 30B	IFT, PE	(G) CommonCrawl, C4, Github, Books, Wikipedia, ArXiv, StackExchange	(G) 1T tokens	[SA] [TC] [NER] [QA] [SMP]	FPB FiQA-SA Headline FIN FinQA ConvFinQA StockNet CIKM18 BigData22	Y	Y	Y	NIPS (D) Jun 2023
	InvestLM [197]	LLaMA	65B	IFT, PE	(G) CommonCrawl, C4, Github, Books, Wikipedia, ArXiv, StackExchange	(G) 1.4T tokens	[SA] [TC] [QA] [Summ]	FPB FiQA-SA FOMC FinQA ECTSum	Y	N	N	ArXiv Sep 2023
	FinGPT [174]	6 open-source LLMs	7B	IFT, PE PEFT	(G) Public Datasets (refer to 6 LLMs)	(G) 2T tokens (e.g. LLaMA2)	[SA] [TC] [NER] [RE]	FPB FiQA-SA Headline FIN FinRED	Y	Y	Y	NIPS (W) Oct 2023

TABLE 2.1: A Summary of Financial PLMs and LLMs.[2]

which is also known as "further pre-training", "post-pre-training", or "pre-fine-tuning". It refers to the use of an initial general LM, then using continual pre-training with a domain-specific corpus in order to adapt its knowledge for domain-specific models. While AI researchers in biomedical domains [55] have extensively explored this technique to develop contextual LMs, its application in the financial domain remains relatively understudied.

FinBERT-19 [6] [3] is the first FinBERT model released for financial sentiment analysis. This model implements three steps: 1) the initialisation of the general-domain PLM, 2) continual

[3] https://huggingface.co/ProsusAI/finbert

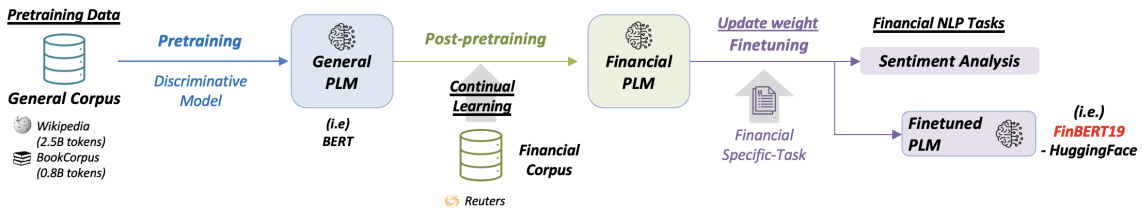


FIGURE 2.7: Continual Pre-training (e.g. FinBERT-19)

pre-training on a financial-domain corpus, and 3) fine-tuning on financial domain-specific NLP tasks. The original BERT model pre-trained on Wikipedia and BookCorpus (3.3B tokens) is initialised, and the Reuters TRC2 dataset is used for continual pre-training for the financial-domain corpus. This dataset contains 1.8M financial news items between 2008 and 2010, and the author filtered out 46,143 docs with 29M words. This further pre-trained model is fine-tuned for the financial sentiment analysis task with the Financial PhraseBank [109] and FiQA18 [108] datasets. The fine-tuned financial LM is released on HuggingFace, and this FinBERT-19 is a task-dependent model for financial sentiment analysis tasks.

2.5.2 Domain-Specific Pre-training from Scratch

The domain-specific pre-training approach involves training a model exclusively on an unlabeled domain-specific corpus while following the original architecture and training objective.

FinBERT-20 [198] [4] is a finance domain-specific BERT model, pre-trained on a large-scale financial communication corpus. While BERT is pre-trained on a general corpus with 3.3B tokens, FinBERT-20 compiles 4.9B tokens from corporate reports (the Securities Exchange Commission website), earnings conference call transcripts (Seeking Alpha) and analyst reports (Investext). To provide a foundation for further research, the author released not only the FinBERT model but also FinVocab uncased/cased, which has a similar token size to the original BERT model. FinBERT-20 also conducted a sentiment analysis task for fine-tuning experiments on the same dataset of FinBERT-19.

[4] https://github.com/yya518/FinBERT

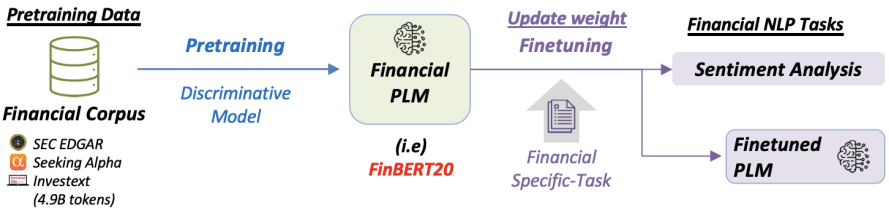


FIGURE 2.8: Domain-specific Pre-training (e.g. FinBERT-20)

2.5.3 Mixed-Domain Pre-training

The mixed-domain pre-training approach involves training a model using both a general-domain corpus and a domain-specific corpus. The assumption is that general-domain text remains relevant, while the financial domain data provides knowledge and adaptation during the pre-training process.

FinBERT-21 [100] [5] is another BERT-based PLM designed for financial text mining. It is trained simultaneously on a general corpus and a financial domain corpus, which is collected from the Financial Web (6.4B words), Yahoo Finance (4.7B words), and RedditFinanceQA (1.62B words). FinBERT-21 employs multi-task learning across six self-supervised pre-training tasks, enabling it to efficiently capture language knowledge and semantic information. The six pre-training tasks are Span Replace Prediction, Capitalization Prediction, Token-Passage Prediction, Sentence Deshuffling, Sentence Distance, and Dialogue Relation. For fine-tuning the model on financial NLP tasks, FinBERT-21 conducted experiments on sentiment analysis, similar to the other two FinBERT models. In addition, it shows the experiment results for two additional tasks; Sentence Boundary Detection and Question Answering.

FLANG [150] [6] is a domain-specific model using financial keywords and phrases for masking and follows the training strategy of ELECTRA [29]. The training process involves a span boundary objective on the generator, which predicts masked financial multi-word representations, and an in-filling objective on the discriminator to assess whether a token is original or replaced. The generator and discriminator are trained end-to-end, and the final discriminator

[5] As of Dec 2023, the repository link mentioned in the paper does not exist.
[6] https://github.com/SALT-NLP/FLANG/

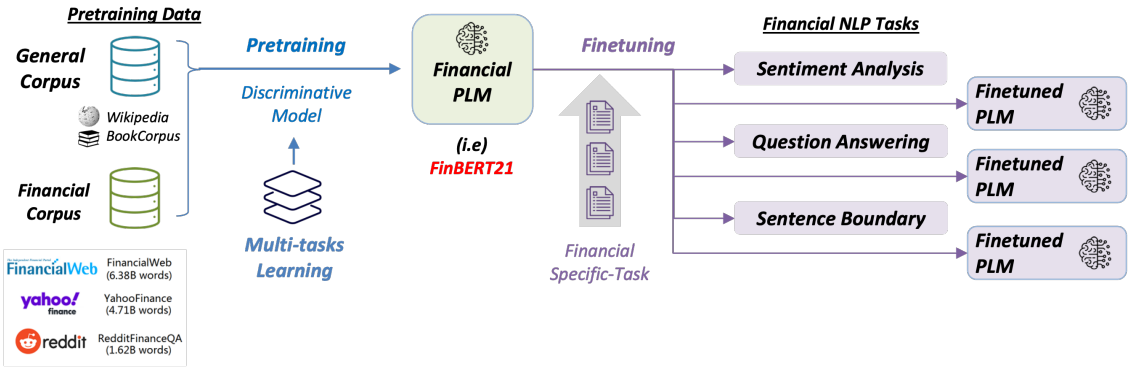


FIGURE 2.9: Mixed-domain Pre-training (e.g. FinBERT-21)

layer is then fine-tuned for specific downstream tasks. This research first introduces Financial Language Understanding Evaluation (**FLUE**), a collection of five financial NLP benchmark tasks. The tasks include Sentiment Analysis, Headline Text Classification, Named Entity Recognition, Structure Boundary Detection, and Question Answering.

2.5.4 Mixed-Domain LLM with Prompt Engineering

Mixed-domain LLMs are trained on both a large general corpus and a large domain-specific corpus. Then, users describe the task and optionally provide a set of examples in human language. This technique is called Prompt Engineering and uses the same frozen LLM for several downstream tasks with no weight updates.

BloombergGPT [186] [7] is the first FinLLM that utilizes the BLOOM model [147] as a backbone structure. It is trained on a large general corpus (345B tokens, 48.73% of training data) and a large financial corpus (363B tokens, 51.27% of training data). Compared to the pre-training data on BERT (3.3B tokens), the general corpus size is over 100 times larger. The financial corpus, FinPile, contains data collected from the web (42%), news, filings, press, and Bloomberg's proprietary data (0.7%). The authors conducted financial NLP tasks (5 benchmark tasks and 12 internal tasks) as well as 42 general-purpose NLP tasks and compared the evaluation results of BloombergGPT (50B), GPT-NeoX (20B), OPT (66B), and BLOOM (176B) models.

[7] This model and its associated data are closed-source.

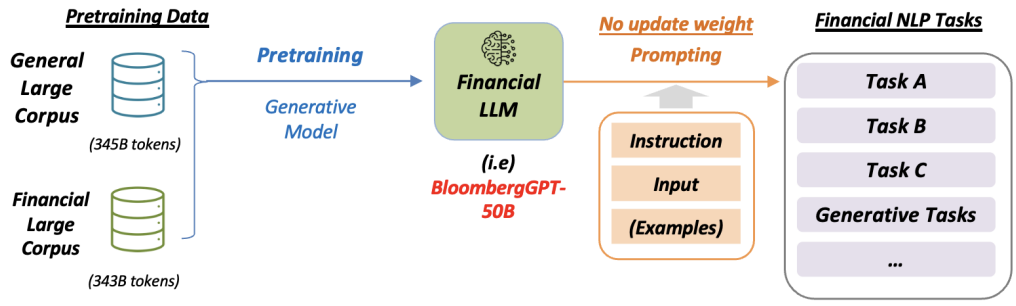


FIGURE 2.10: Mixed-domain LLM with Prompt Engineering (e.g. BloombergGPT)

2.5.5 Instruction-Finetuned LLM with Prompt Engineering

The primary objective of training LLMs is to minimize errors in contextual word prediction on extensive corpora. Instruction Tuning involves further training LLMs using explicit text instructions, leading to notable improvements in zero-shot learning performance across unseen tasks. In finance, researchers have started transforming existing financial datasets into instruction datasets and subsequently using these datasets for fine-tuning LLMs. Three representative FinLLMs are presented below.

FinMA [188] [8] (introduced in the PIXIU framework) consists of two fine-tuned LLaMA models (7B and 30B) [167] that use financial instruction datasets for financial tasks. It is constructed from a large-scale multi-task instruction dataset called Financial Instruction Tuning (FIT, 136k samples) by collecting nine publicly released financial datasets used across five different tasks. Then, the LLaMA model is fine-tuned on the FIT dataset. In addition to the five FLUE benchmark tasks, the Stock Movement Prediction task is also included.

InvestLM [197] [9] is a fine-tuned LLaMA-65B model, similar to FinMA, using a manually curated financial domain instruction dataset. The dataset covers Chartered Financial Analyst (CFA) exam questions, SEC filings, StackExchange quantitative finance discussions, Academic Journals, Finance Textbooks, Financial NLP tasks, and Investment questions. The LLaMA-65B model is trained on this instruction dataset using the Low-Rank Adaptation

[8] https://github.com/chancefocus/PIXIU

[9] The instruction dataset is not publicly available, but the model has been released on their GitHub repository. https://github.com/AbaciNLP/InvestLM

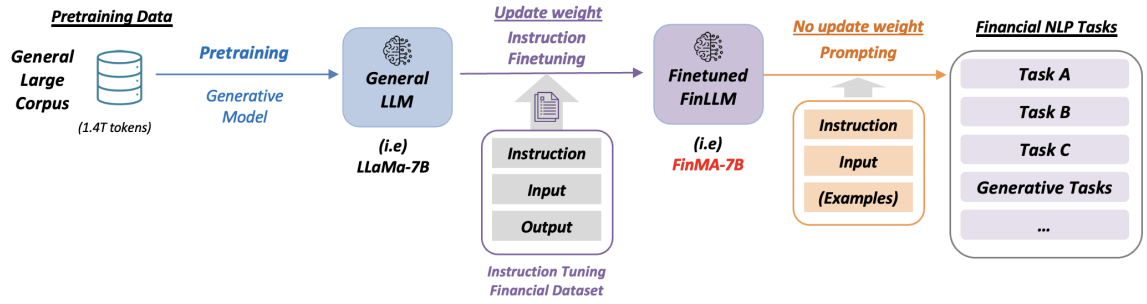


FIGURE 2.11: Instruction-finetuned LLM with Prompt Engineering (e.g. FinMA)

(LoRA) [64] method, which enables efficient tuning of the model parameters. This fine-tuned LLM, InvestLM, is evaluated alongside other commercial LLMs, including GPT-3.5, GPT-4, and Claude-2. The downstream tasks are similar to FinMA but also include a financial text summarization task.

FinGPT [193] [10] is an open-sourced and data-centric framework, which provides; a suite of APIs for financial data sources, an instruction dataset for various financial tasks similar to FinMA, and several fine-tuned financial LLMs. The authors have released several similar papers that describe the framework and a model experiment paper [174] on the instruction fine-tuned LLMs using 6 open-source LLMs including LLaMA2, Falcon, MPT, BLOOM, ChatGLM2, and Qwen.

2.6 Evaluation: Benchmark Tasks and Datasets

As LLMs gain significant attention, evaluating them becomes increasingly critical. Benchmark evaluations typically include specific tasks, datasets for each task, and evaluation metrics [17]. In financial NLP research, FLANG [150] first introduces a set of financial benchmark tasks called Financial Language Understanding Evaluation (**FLUE**). Since then, a few new tasks have been added to the evaluation of FinLLMs. In this section, I summarize six financial NLP benchmark tasks and datasets and review the evaluation results of models, including FinPLMs, FinLLMs, ChatGPT, GPT-4, and task-specific State-of-the-Art (SOTA) models.

[10] https://github.com/AI4Finance-Foundation/FinGPT

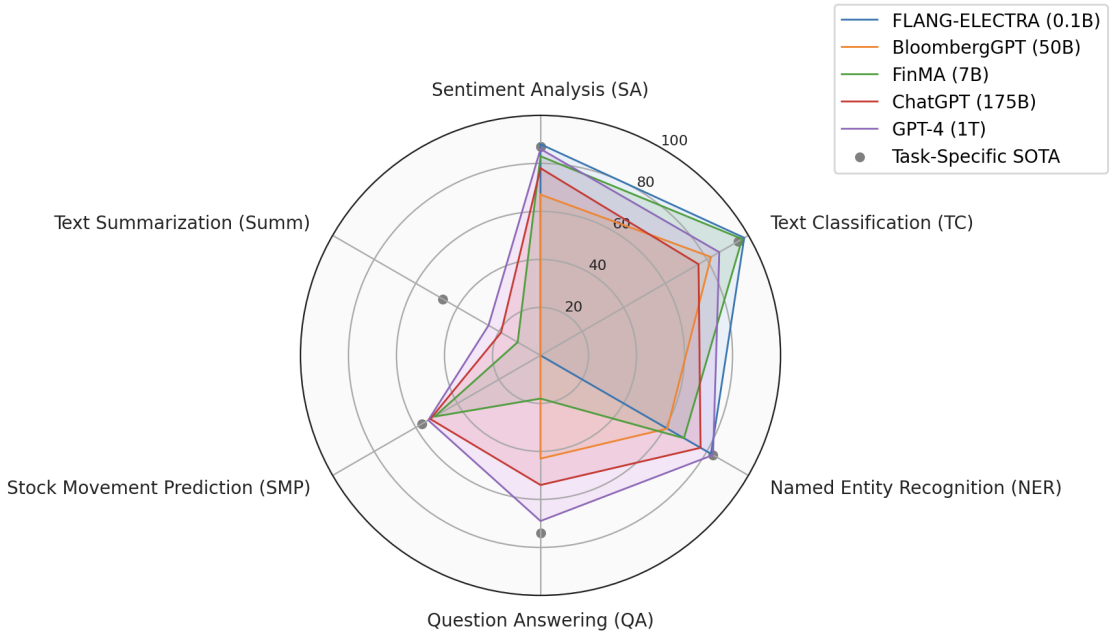


FIGURE 2.12: Evaluation of six LLMs, including General-domain LLMs and FinLLMs.

Figure 2.12 shows the normalized summary of FLANG-ELECTRA -110M, BloombergGPT -50B, FinMA-7B, ChatGPT-175B, GPT4-1T, and task-specific SOTA across six financial NLP tasks. The six financial NLP tasks are Sentiment Analysis (SA), Text Classification (TC), Named Entity Recognition (NER), Question Answering (QA), Stock Movement Prediction (SMP), and Text Summarisation (Summ). FinPLMs did not perform experiments on the more complex tasks such as SMP or Summ; hence, the results are not reported. Additionally, the results of ChatGPT and GPT-4 are referenced from original research or analysis research [94]. SOTA results are collected from task-specific models or the best performance results presented on datasets.

2.6.1 Sentiment Analysis (SA)

The Sentiment Analysis (SA) task aims to analyze sentiment information from input text, including financial news and microblog posts. Most FinPLMs and FinLLMs report the evaluation results of this task using **Financial PhraseBank (FPB)** and **FiQA-SA** dataset.

The **FPB** [109] dataset consists of 4,845 randomly selected English financial news articles about companies in OMX Helsinki. Domain experts annotated each sentence with one of three sentiment labels: positive, negative, or neutral. The **FiQA-SA** [108] dataset (task-1: Aspect-based financial sentiment analysis) consists of 1,173 posts from both headlines and microblogs. The sentiment scores are on a scale of [-1, 1], where -1 is the most negative, and 1 is the most positive sentiment. Recent FinLLMs have converted this score into a classification task and reported their results.

Category	Model	Size	FPB (F1)	FiQA-SA (F1)
PLM	BERT-base	0.11B	0.86	-
FinPLM	FinBERT20	0.11B	0.87	-
	FLANG-ELECTRA	0.11B	**0.92**	-
FinLLM	BloombergGPT	50B	0.51	0.75
IFT_FinLLM	FinMA 7B-Full	7B	0.87	0.79
	FinMA 30B	30B	<u>0.88</u>	<u>0.87</u>
LLM	ChatGPT	175B	0.75	0.78
	GPT4	1T	0.86	**0.88**

TABLE 2.2: Sentiment Analysis evaluation results on Financial PhraseBank (FPB) and FIQA-SA datasets. The table reports the 5-shot performance of all LLM baseline methods. Bold indicates the best performance metric, and underline indicates the second-best performance metric.

Table 2.2 presents comparison results of sentiment analysis on both FPB and FiQA SA datasets. Overall, **FLANG-ELECTRA** (mixed-domain PLM) achieved the best results (92% on F1), while FinMA-30B (IFT FinLLM) and GPT-4 achieved similar results (87% on F1) with a 5-shot prompting. It suggests that instruction fine-tuning on financial-specific data represents a practical approach in terms of efficiency and cost-effectiveness.

For further evaluation of SA, three open-released datasets can be considered: SemEval-2017 (Task 5), FinEntity, and StockEmotions. The **SemEval-2017** [31] dataset comprises 4,157 sentences collected from both headlines and microblogs. Similar to FiQA SA, the sentiment scores are on a scale of [-1, 1]. **FinEntity** [164] is an entity-level sentiment classification dataset containing 979 financial news paragraphs that annotate financial entity spans (a total of 2,131 entities) and their sentiment (positive, neutral, and negative). The **StockEmotions** [86] dataset consists of 10,000 sentences collected from microblogs that annotate binary

sentiment and 12 fine-grained emotion classes that span the multi-dimensional range of investor emotions.

2.6.2 Text Classification (TC)

Text Classification (TC) is classifying a given text or document into predefined labels based on its content. In financial text, there are often multiple dimensions of information beyond sentiment, such as price directions or interest rate directions. FLUE includes the gold news **Headline** [156] dataset for text classification. This dataset comprises 11,412 news headlines, labelled with a binary classification across nine labels such as "price up" or "price down".

Category	**Model**	**Size**	**Headline-Gold News (Avg. F1)**
PLM	BERT-base	0.11B	0.97
FinPLM	FinBERT20	0.11B	0.97
	FLANG-ELECTRA	0.11B	**0.98**
FinLLM	BloombergGPT	50B	0.82
IFT_FinLLM	FinMA 7B-Full	7B	0.97
	FinMA 30B	30B	**0.98**
LLM	ChatGPT	175B	0.77
	GPT4	1T	0.86

TABLE 2.3: Text Classification evaluation results on Headline dataset. The table reports the 5-shot performance of all LLM baseline methods.

Table 2.3 shows the average F1-score from representative FinPLMs/FinLLMs on the gold news headline dataset. Similar to Sentiment Analysis, **FLANG-ELECTRA** (mixed-domain PLM) and **FinMA-30B** (IFT FinLLM) with a 5-shot prompting achieved the best results (98% on Avg. F1). When examining the results of BERT (general-domain PLM) and FinBERT20 (FinPLM), the performance is also noteworthy (97% on Avg. F1), suggesting that using a PLM with fine-tuning for a specific task can be a practical approach depending on the task complexity.

As Text Classification is a broad task depending on the dataset and its predefined labels, four open-released financial TC datasets are included for further research: M&A, FedNLP, FOMC, and Banking77. The **M&A** [195] dataset contains 4,098 paragraphs concerning

merger and acquisition deals from financial news and Twitter, associated with four labels: complete, rumour, pending, and cancelled. In their repository, 1,000 paragraphs are publicly available, while the training data is currently undisclosed. The **FedNLP** [84] dataset comprises 1,422 documents sourced from various Federal Open Market Committee (FOMC) materials, including speeches, meeting minutes, and press conferences, in addition to financial news. The dataset is annotated with labels as Up, Maintain, or Down based on the Federal Reserve's Federal Funds Rate decision for the subsequent period. Similarly, the **FOMC** [149] dataset is a collection of 496 FOMC documents with the labels categorizing the documents as Dovish, Hawkish, or Neutral, reflecting the prevailing sentiment conveyed within the FOMC materials. The **Banking77** [15] dataset comprises 13,083 samples covering 77 intents related to banking customer service queries, such as "card loss" or "linking to an existing card". This dataset is designed for intent detection tasks and for the development of conversation systems, such as chatbots.

2.6.3 Named Entity Recognition (NER)

The Named Entity Recognition (NER) task is the extraction of information from unstructured text and categorizing it into predefined named entities such as locations (LOC), organizations (ORG), and persons (PER). In the financial domain, detecting key entities can be used to build financial knowledge graphs, identifying the interrelationships among diverse entities. Beyond the fundamental entities, the named entities in the financial text include events, products, time, quantities, monetary values, and percentages. For the financial NER task, the **FIN** dataset [2] is included in FLUE benchmarks. The FIN dataset comprises eight financial loan agreements (totalling 54,256 words) from the US Security and Exchange Commission (SEC) for credit risk assessment. The dataset is manually annotated with entity types, including PER, LOC, ORG, and MISC. Notably, MISC tags are excluded during most LLM evaluations due to their ambiguous definitions.

Table 2.4 shows Entity F1-scores obtained by representative models on FIN dataset. **GPT-4** (over 1T parameters) with a 5-shot approach outperforms (83% on Entity F1) other models, indicating the superior in-context learning capabilities of extremely large language models.

Category	Model	Size	FIN-CreditRisk (Entity F1)
PLM	BERT-base	0.11B	0.79
FinPLM	FinBERT20	0.11B	0.80
	FLANG-ELECTRA	0.11B	0.82
FinLLM	BloombergGPT	50B	0.61
IFT_FinLLM	FinMA 7B-Full	7B	0.69
	FinMA 30B	30B	0.62
LLM	ChatGPT	175B	0.77
	GPT4	1T	**0.83**

TABLE 2.4: Named Entity Recognition evaluation results on FIN dataset. The table reports the 20-shot performance of BloombergGPT and the 5-shot performance of other LLM baseline methods.

FLANG-ELECTRA demonstrates notable performance, while other FinLLMs exhibit sub-optimal results. This indicates that scaling models alone may not be adequate for optimal performance in finance.

As NER task results can vary across datasets, two publicly released financial NER datasets are included for further research: FiNER-139 and FNXL. The **FiNER-139** [105] dataset consists of 1.1M sentences annotated with 139 eXtensive Business Reporting Language (XBRL) word-level tags sourced from the SEC. XBRL tags are XML-based, contain mostly numeric tokens, and aim to facilitate the processing of financial information. This dataset is designed for Entity Extraction and Numerical Reasoning tasks, predicting the XBRL tags (e.g., cash and cash equivalents) based on numeric input data within sentences (e.g., "24.8" million). Similarly, the **FNXL** [151] dataset is sourced from SEC financial reports and comprises 79,888 sentences. It includes 142,922 annotated numerical tokens associated with 2,794 XBRL tags for Entity Extraction and Numerical Reasoning tasks. In contrast to FiNER-139, which focuses on the most frequent 139 tags, FNXL remains unfiltered. The repository contains 3,000 publicly accessible samples.

2.6.4 Question Answering (QA)

Question Answering (QA) is a task to retrieve or generate answers to questions from an unstructured collection of documents. Financial QA is more challenging than general QA as it

requires numerical reasoning across multiple formats, including tables and unstructured texts. **FiQA-QA** [108] represents an early Financial QA dataset, consisting of 17,110 QA pairs for opinion-based QA collected from microblogs and forums. Over time, the Financial QA dataset has evolved to include complex numerical reasoning in multi-turn conversations. This evolution involves the introduction of **hybrid QA**, which is to create paths to connect hybrid contexts, including both tabular and textual content. **FinQA** [24] is a single-turn hybrid QA dataset having 8,281 QA pairs annotated by experts from the annual reports of S&P 500 companies. The FinQA data originates from FinTabNet [207], which contains 70k pages with full table bounding boxes, structural annotations, and more than 110k tables with cell bounding boxes. **ConvFinQA** [23], an extension of FinQA, is a multi-turn conversational hybrid QA dataset consisting of 3,892 conversations with 14,115 questions. This dataset requires answering conversational questions designed for extensive numerical calculations over the input text and at least one table with financial data.

Category	Model	Size	FiQA-QA (nDCG)	FinQA (EM Acc)	ConvFinQA (EM Acc)
PLM	BERT-base	0.11B	0.46	-	-
FinPLM	FinBERT20	0.11B	0.42	-	-
	FLANG-ELECTRA	0.11B	0.55	-	-
FinLLM	BloombergGPT	50B	-	-	0.43
IFT_FinLLM	FinMA 7B-Full	7B	-	0.04	0.20
	FinMA 30B	30B	-	0.11	0.40
LLM	ChatGPT	175B	-	0.49	0.60
	GPT4	1T	-	**0.69**	**0.76**
Human	Human Expert			0.91	0.89
	General Crowd			0.51	0.47

TABLE 2.5: Question Answering evaluation results on FiQA-QA, FinQA, and COnvFinQA datasets. As an evaluation metric, Normalized Discounted Cumulative Gain (nDCG) was used for FiQA-QA, and Exact Match Accuracy (EM Acc) was used for FinQA and ConvFinQA. The table reports the zero-shot performance of all LLM baseline methods.

Table 2.5 presents the evaluation results for the Financial QA task, where all FinLLMs conducted experiments on the FinQA and/or ConvFinQA datasets to assess their numerical reasoning capabilities, with Exact Match Accuracy used as the evaluation metric. **GPT-4** with a zero-shot prompting outperforms all other models (69%-76% on EM Accuracy), approaching

the performance of human experts (Avg. 90% on EM Accuracy). This indicates that extremely large LM would have enhanced capabilities in complex numerical reasoning across diverse data modalities. Notably, **BloombergGPT**'s results (43% on EM Accuracy) were slightly below the general crowd (47% on EM Accuracy), indicating areas for improvement in FinLLMs.

Given the complexity of the Financial QA task and for future research, two additional publicly released financial QA datasets are included: TAT-QA and PACIFIC. **TAT-QA** [210] (similar to FinQA) is a single-turn hybrid QA dataset having 16,552 QA pairs sourced from 182 financial reports. Designed for the Numerical Reasoning (NR) task over both tables and text, TAT-QA requires answers based on operations such as addition, subtraction, multiplication, division, counting, comparison/sorting, and their compositions. **PACIFIC** [35] (similar to ConvFinQA), an extension of TAT-QA, is a multi-turn conversational hybrid QA dataset comprising 19,008 QA pairs within 2,757 dialogues. PACIFIC focuses on proactively assisting users in clarifying ambiguity or uncertainty in their queries by asking clarifying questions.

2.6.5 Stock Movement Prediction (SMP)

The Stock Movement Prediction (SMP) task has received significant attention in both research and the financial industry for a number of years. Typically, it is framed as a classification problem, predicting the next day's price movement (e.g., up or down) based on historical stock prices and associated text data. This task is particularly challenging due to the need to integrate time series problems with temporal dependencies extracted from text information, where text data can act both as noise and signal. In FinLLMs' evaluation, FinMA includes the SMP tasks for the first time, conducting experiments on three publicly released SMP datasets; StockNet, CIKM18, and BigData22.

StockNet [192] collected historical price data and Twitter data between January 2014 and January 2016 for 88 stocks listed in the S&P. The task is framed as a binary classification with a threshold: a price movement higher than 0.55% is labelled as a rise (denoted as 1), while a movement less than -0.5% is labelled as a fall (denoted as 0). This dataset is widely

used for SMP tasks, and the current task-specific SOTA model for this dataset has achieved approximately 61% accuracy. Similarly, **CIKM18** [185] utilises historical price and Twitter data ranges from January 2017 to November 2017 for 47 stocks in the S&P 500. The task is also formulated as a binary classification, but without using a threshold. The model proposed by the authors achieved an accuracy of approximately 59%. **BigData22** [159] compiled data spanning from July 2019 to June 2020 for 50 high-trade-volume stocks in the US stock markets. Like StockNet, it adopts a binary classification formulation with a threshold. The proposed model by the authors focuses on leveraging sparse noisy tweets to extract multi-level patterns from historical prices, resulting in an achieved accuracy of approximately 56% on their dataset.

Category	Model	Size	StockNet (Acc)	CIKM18 (Acc)	BigData22 (Acc)
IFT_FinLLM	FinMA 7B-Full	7B	<u>0.56</u>	0.53	0.49
	FinMA 30B	30B	0.49	0.43	0.47
LLM	ChatGPT	175B	0.50	0.55	0.53
	GPT4	1T	0.52	<u>0.57</u>	<u>0.54</u>
Task-specific	SOTA		**0.61**	**0.59**	**0.55**

TABLE 2.6: Stock Movement Prediction evaluation results on StockNet, CIKM18 and BigData22 datasets. The table reports the zero-shot performance of other LLM baseline methods.

Table 2.6 presents the comparison results of FinMA, ChatGPT, GPT-4, and task-specific SOTA models across three datasets. Considering the challenges of this task, which involves designing models to address removing noises while detecting signals from text data within optimal window sizes, the task-specific SOTA models outperform all LLMs. On average, across these three datasets, **GPT-4** with a zero-shot prompting achieves higher performance (54% on Accuracy) than FinMA (52% on Accuracy) and slightly lower results than the SOTA model (58% on Accuracy). Although NLP metrics such as accuracy are commonly used, they are insufficient for the SMP evaluation. It is important to consider financial evaluation metrics, such as the Sharpe ratio, as well as backtesting simulation results.

For future research, a CryptoBubbles dataset for predicting the cryptocurrency market is included. **CryptoBubbles** [145] consists of historical prices and over 2M Reddit comments

spanning five years for 456 cryptocurrencies. It is designed for the Crypto Market Prediction task and formulates a binary classification problem using the logarithmic price change. The best-performing model by the author on this dataset achieved an F1-score of 53%.

2.6.6 Text Summarization (Summ)

Text Summarization (Summ) is the generation of a concise summary from documents while conveying its key information via either an extractive or an abstractive approach. In finance, it has been relatively underexplored due to the lack of benchmark datasets, challenges with domain experts' evaluations, and the need for disclaimers when presenting financial advice. InvestLM includes summarization tasks for the first time, conducting experiments on the ECTSum dataset. **ECTSum** [119] consists of a total of 2,425 document-summary pairs, containing Earnings Call Transcripts (ECTs) and bullet-point summarisations from Reuters between January 2019 and April 2022.

Category	Model	Size	ECTSum (Rouge-1)
IFT_FinLLM	FinMA 7B-Full	7B	0.08
	InvestLM	65B	0.26
LLM	ChatGPT	175B	0.21
	GPT4	1T	0.30
Task-specific	SOTA		**0.47**

TABLE 2.7: Text Summarization evaluation results on ECTSum dataset. The table reports the zero-shot performance of all LLM baseline methods.

Table 2.7 presents the results of summarization on ECTSum. Similar to other complex financial tasks, the **task-specific SOTA** model (47% on ROUGE-1) outperforms all LLMs. According to the authors of InvestLM, while GPT-4 shows superior performance compared to InvestLM, the commercial models generate decisive answers. The financial summarization task offers significant development opportunities, exploring whether FinLLMs can outperform task-specific SOTA models.

For ongoing research, an additional financial summarization dataset, MultiLing 2019, is included. **MultiLing 2019** [57] contains 3,863 document-summary pairs extracted from

UK annual reports listed on the London Stock Exchange (LSE). It provides at least two gold-standard summaries for each annual report.

Task	Dataset	Modality	Data size	Data Type	Data Source	LLMs	Link
[SA]	FPB [109]	T	4,845 sentences	news	LexisNexis database	Y	HuggingFace
	FiQA-SA [108]	T	1,173 sentences	news headlines, tweets	Not disclosed	Y	Github
	SemEval17 [31] (Task 5)	T	4,157 sentences	news headlines, tweets	Twitter, StockTwits, various news providers	N	BitBucket
	FinEntity [164]	T	979 paragraphs, 2,131 entities	news	Reuters	N	Github
	StockEmotions [86]	T + TS	10,000 sentences, 39 stocks	tweets, historical prices	StockTwits, Yahoo finance	N	Github
[TC]	Headline [156]	T	11,412 sentences	news headlines	various news providers	Y	Kaggle
	M&A [195]	T	4,098 paragraphs	news, tweets	M&A deal from Zephyr	N	Github
	FedNLP [84]	T	1,422 instances	FOMC docs, news	FOMC, various providers	N	Github
	FOMC [149]	T	496 instances	FOMC docs	FOMC	N	Github
	Banking77 [15]	T	13,083 samples, 77 intents	banking queries	Not disclosed	N	HugginFace
[NER]	FIN [2]	T	54,256 words, 8 docs	financial agreement	SEC (Loan Agreement)	Y	Github
	FiNER-139 [105]	T	1.1M sentences, 139 labels	financial reports	SEC (10-K)	N	HuggingFace
	FNXL [151]	T	79,888 sentences, 2,794 labels	financial reports	SEC (10-K)	N	Github
[QA]	FiQA-QA [108]	T	17,110 QA pairs	web posts	Stackexchange (2009-2017)	Y	HuggingFace
	FinQA [24]	T + Tb	8,281 QA pairs	financial reports	FinTabNet	Y	Github
	ConvFinQA [23]	T + Tb	14,115 QA pairs	financial reports	FinTabNet, FinQA 3.8k dialogues	Y	Github
	TAT-QA [210]	T + Tb	16,552 QA pairs	financial reports	AnnualReports	N	Github
	PACIFIC [35]	T + Tb	19,008 QA pairs	financial reports	AnnualReports, TAT-QA 2.7k dialogues	N	Github
[SMP]	StockNet [192]	T + TS	29,250 pred. targets, 87 stocks	tweets, historical prices	Twitter, Yahoo finance	Y	Github
	CIKM18 [185]	T + TS	15,015 pred. targets, 38 stocks	tweets, historical prices	Twitter, Yahoo finance	Y	Github
	BigData22 [159]	T + TS	14,041 pred. targets, 50 stocks	tweets, historical prices	Twitter, Yahoo finance	Y	Github
	CryptoBubbles [145]	T + TS	2.4M tweets, 456 cryptos	tweets, historical prices	Twitter, CryptoCompare Reddit	N	Github
[Summ]	ECTSum [119]	T	2,425 docs-summ pairs	earning call transcripts	The Motley Fool Reuters	Y	Github
	MultiLing19 [57]	T	3,863 docs-summ pairs	UK annual reports	London Stock Exchange	N	Website
[RE]	FinRED [152]	T	6,767 sentences, 29 relations	news earning call transcripts	Webhose, Wikidata SeekingAlpha	N	Github
[ED]	EDT [209]	T	9,721 news (for ED), 30,3893 news (pred.)	news	various news providers, Plygon, Investopedia	N	Github
[CD]	FinCausal20 [110]	T	29,444 sentences	news	Qwam	N	Github
[DU]	FinTabNet [207]	T + Tb	89,646 pages, 11,2887 tables	financial reports	S&P 500 companies	N	Website
	Form-NLU [38]	T + Tb + I	857 form image, 4278 tables	financial reports	Australian Stock Exchange (ASX)	N	Github
[MM]	PEAD [135]	T + A + TS	576 instances, 88,829 sentences	earning call transcripts	SeekingAlpha, EarningsCast (280 corp.)	N	Github
	MAEC [91]	T + A + TS	3,443 instances, 394,277 sentences	earning call transcripts	SeekingAlpha, EarningsCast (1213 corp.)	N	Github
	MONOPOLY [112]	T + A + V + TS	24,180 samples, 340 videos	MPC transcripts, historical prices	Six Central Banks (FRB, BoC, BoE, BNZ, ECB, SARB)	N	Github
[MT]	MINDS14 [51]	T + A (14 langs)	8,168 samples, 14 intents	banking voice assistant	Crowdsourcing	N	HugginFace
	MultiFin [71]	T (15 langs)	10,048 samples, (8,0678 tokens)	public articles	accounting firm (Not disclosed)	N	Github

TABLE 2.8: A Summary of datasets for both conventional and advanced Financial NLP tasks. [11]

[11]The Task column highlights the primary task, although datasets may be available for other downstream tasks. (T = Text, TS = Times Series, Tb = Table, I = Image, A = Audio, V = Video)

2.7 Advanced Financial NLP Tasks and Datasets

Properly designed benchmark tasks and datasets are crucial resources for assessing the capability of LLMs. While the current six benchmark tasks provide a foundation, they tend to be less complex and overlook many existing advanced financial NLP tasks. This section presents 8 advanced financial NLP tasks and their corresponding datasets. Data was collected from papers that are easily accessible, published between 2017 and 2023, and focused on various complex financial NLP tasks. Table 2.8 [12] presents the collected papers for both existing and advanced benchmark tasks and datasets within the financial domain.

2.7.1 Relation Extraction (RE)

The **Relation Extraction** (RE) task aims to identify and classify relationships between entities implied in the text. Similar to NER, this task is part of Information Extraction. The **FinRED** [152] dataset is released for RE and is curated from financial news and earning call transcripts, containing 29 relation tags (e.g. owned by) specific to the finance domain.

2.7.2 Event Detection (ED)

Event Detection (ED) in finance involves identifying the impact of how investors perceive and assess related companies. The **Event-Driven Trading** (EDT) [209] dataset includes 11 types of corporate event detection. EDT comprises 9,721 news articles with token-level event labels and an additional 303,893 news articles with minute-level timestamps and stock price labels.

[12] Due to space constraints, I include the table acronym here. The LLMs column indicates Y if the dataset has been utilized in any existing FinLLMs to date. The abbreviations correspond to [SA] Sentiment Analysis, [TC] Text Classification, [NER] Named Entity Recognition, [QA] Question Answering, [SMP] Stock Movement Prediction, [Summ] Text Summarization, [RE] Relation Extraction, [ED] Event Detection, [CD] Causality Detection, [DU] Document Understanding, [MM] Multi-modal Understanding, [MT] Machine Translation); Datasets are included if it is publicly accessible as of Dec 2023.

2.7.3 Causality Detection (CD)

Causality Detection (CD) in finance aims to identify cause-and-effect relationships within the factual text to generate meaningful financial narrative summaries. The **FinCausal20** [110] dataset from Financial Narrative Processing (FNP) includes tasks for detecting causal schemes in a given text and identifying cause-and-effect sentences.

2.7.4 Numerical Reasoning (NR)

Numerical Reasoning (NR) in finance aims to identify numbers and mathematical operators in either digit or word form, to perform calculations and comprehend financial context (e.g. cash and cash equivalent). Some datasets introduced for NER and QA tasks are also designed for numerical reasoning, including: **FiNER-139** [105], **FNXL** [151], **FinQA** [24], **ConvFinQA** [23], **TAT-QA** [210], **PACIFIC** [35].

2.7.5 Document Understanding (DU)

Document Understanding (DU) is a broad task focused on analysing the layout within a document (e.g. text, tables, or figures), recognising logical relationships, and extracting key information. IBM Research has released the **FinTabNet** [207] dataset, which was collected from the earnings reports of S&P 500 companies. This dataset comprises unstructured PDF documents with detailed annotations of table structures. The FinQA and ConvFinQA datasets, included in QA tasks, have been further developed from FinTabNet. **Form-NLU** [38] consists of 857 form images and 4,278 tables collected from the Australian Stock Exchange (ASX), specifically designed for visually-rich document understanding, which requires comprehending form structure and extracting key-value information.

2.7.6 Multimodal (MM)

Multimodal (MM) understanding is a challenging task across many domains. Recently, several multimodal financial datasets have been introduced, combining text, time series, audio,

and videos. **PEAD** [135] gathers 576 instances with 88,829 sentences extracted from US companies' earnings call transcripts, providing text, time series, and audio data. Similarly, **MAEC** [91] compiles the multimodal data from earnings call transcripts on a larger scale, with 3,443 instances and 394,277 sentences. Additionally, **MONOPOLY** [112] introduces video data from monetary policy call transcripts across six central banks, sharing 24,180 samples from 340 videos with text scripts and time series.

2.7.7 Machine Translation (MT)

Machine Translation (MT) in finance aims to translate sentences from a source language to a target language and comprehend the financial contextual meaning in different languages. **MINDS-14** [51] consists of 8,168 samples of banking voice assistant data in text and audio formats across 14 different languages. **MultiFin** [71] includes 10,048 samples covering financial topics with 6 high-level labels (e.g., Finance) and 23 low-level labels (e.g., M&A & Valuations), sourced from public financial articles in 15 different languages.

2.7.8 Market Forecasting (MF)

Market Forecasting (MF) is an essential task in financial markets, involving the prediction of market price, volatility, and risk. In finance, stock price volatility, often measured as the standard deviation of a stock's returns over a specific period, is a commonly used indicator of financial risk. Various works have studied the problem of financial risk prediction using firm financial reports. This task extends beyond Stock Movement Prediction (SMP), which formulates problems as a classification task. The commonly used metrics for MF include regression evaluation metrics such as R-squared, Mean Squared Error (MSE), Root Mean Squared Error (RMSE), Mean Absolute Error (MAE), and Mean Absolute Percentage Error (MAPE). The datasets introduced in Sentiment Analysis (SA), Event Detection (ED), and Multi-Modal (MM) tasks are also designed for Market Forecasting. This section includes a list of datasets relevant to Market Forecasting: **StockEmotions**(SA) [86], **EDT**(ED) [209], **PEAD**(MM-audio) [135], **MAEC**(MM-audio) [91], and **MONOPOLY**(MM-video) [112].

2.8 Conclusion

This literature review offers a concise yet comprehensive exploration of Large Language Models in the financial domain (FinLLMs). Furthermore, by examining a diverse range of financial NLP tasks and datasets, this chapter presents an extensive overview of the relevant financial NLP literature. Firstly, this chapter presents an overview of the evolutionary model trend, from general-purpose LLMs to financial domain-specific LLMs. Secondly, this chapter conducts a technical comparison of eight different models, from FinPLMs to FinLLMs. This comparison includes techniques such as continual pre-training, domain-specific pre-training, mixed-domain pre-training, mixed-domain LLM with prompt engineering, and instruction-fine-tuning with prompt engineering. Thirdly, this chapter summarizes the evaluation results for six conventional benchmark tasks and includes additional datasets per task to facilitate future research. Moreover, this chapter presents eight advanced benchmark financial NLP tasks and compiles a collection of accessible datasets for each task. The unique opportunities and challenges associated with FinLLMs are presented in the 6 Discussion chapter. In summary, this chapter delivers a comprehensive investigation beyond FinLLMs, and this in-depth summary aims to benefit both the Computer Science and Finance research communities.

CHAPTER 3

Interpretable NLP System in Finance: FedNLP Application

This chapter presents the first financial NLP segment of the thesis, the FedNLP application, a novel financial NLP system designed for end-users to explore a variety of Financial NLP tasks. It is an extension of the work ***FedNLP: an Interpretable NLP System to Decode Federal Reserve Communications*** [84] accepted to the ACM SIGIR 2021. I formulated the research aim, collected the data, designed the methodology, analyzed the data, conducted the experiments, and wrote the whole paper.

The Federal Reserve System (the Fed) plays a significant role in affecting monetary policy and financial conditions worldwide. Although it is important to analyze the Fed's communications to extract useful information, it is generally long-form and complex due to the ambiguous and esoteric nature of the content. This chapter presents FedNLP, an interpretable multi-component NLP system to decode Federal Reserve communications with no coding. This system is designed for end-user to support their holistic understanding of the Fed's communications through multiple NLP techniques such as sentiment analysis, topic modelling, prediction, explanation, and summarization. By employing pre-trained language model methods, the system uses fine-tuned Google's T5 model for summarization and fine-tuned the FinBERT model for prediction. This work focuses primarily on prediction using machine learning, neural networks, and pre-trained language models to forecast the changes to the Federal Funds Rate. To evaluate the system, this work conducted focus group interviews, surveys, and post-experiment interviews with end-users. The results show that summarization using a financial context and an interactive demo are the most useful components. In order to build the system and models, this work collected the Fed's communications from over 30 websites.

3.1 Introduction

Over the years, the role of the U.S. Federal Reserve System (the Fed) has expanded due to changes in the monetary and financial conditions globally. The Fed's decisions have a chain effect on a broader range of economic factors like inflation, employment, the value of currency, growth, and loans [70]. Therefore, it is important to analyze the Fed's communications that anchor and guide market expectations. However, it is generally long-form and complex due to the ambiguous and esoteric nature of content [9].

Additionally, the Fed has increased its interest in research exploring the importance of Natural Language Processing (NLP) for macroeconomics. It is aligned with the remarkable progress in NLP that has seen the emergence of a massive number of model architectures (e.g. Transformers [168]), pre-trained models (e.g. BERT [36], T5 [138]), and high-level of library wrappers (e.g. HuggingFace [143]). Because the Fed supervision carries vast amounts of unstructured data, a significant improvement in NLP research could assist their needs. However, there are no pilot studies to identify how NLP components could help end-users analyze Federal Reserve communications.

This chapter presents **FedNLP**, an interpretable multi-component **NLP** system that aims to decode **Fed**eral Reserve communications with no code. The system is designed for end-users who are unfamiliar with programming by assisting their holistic and intuitive understanding of the Fed's communications through the use of multiple NLP components. Inspired by recent research that combines language tools [165], this system focuses on presenting multiple NLP components such as sentiment analysis [104], topic modelling [139], prediction [36, 198], explanation [141], and summarization [138] in one web application (Fig. 3.1).

This work's objectives are; to define the system and component requirements, to collect real-world data, to build the end-to-end system containing multiple financial NLP components, to perform in-depth experiments on a prediction task, and to evaluate the system.

Initially, the work defines end-users, system design criteria, system components, and functional system flows. Additionally, this research collects text data associated with the Fed's

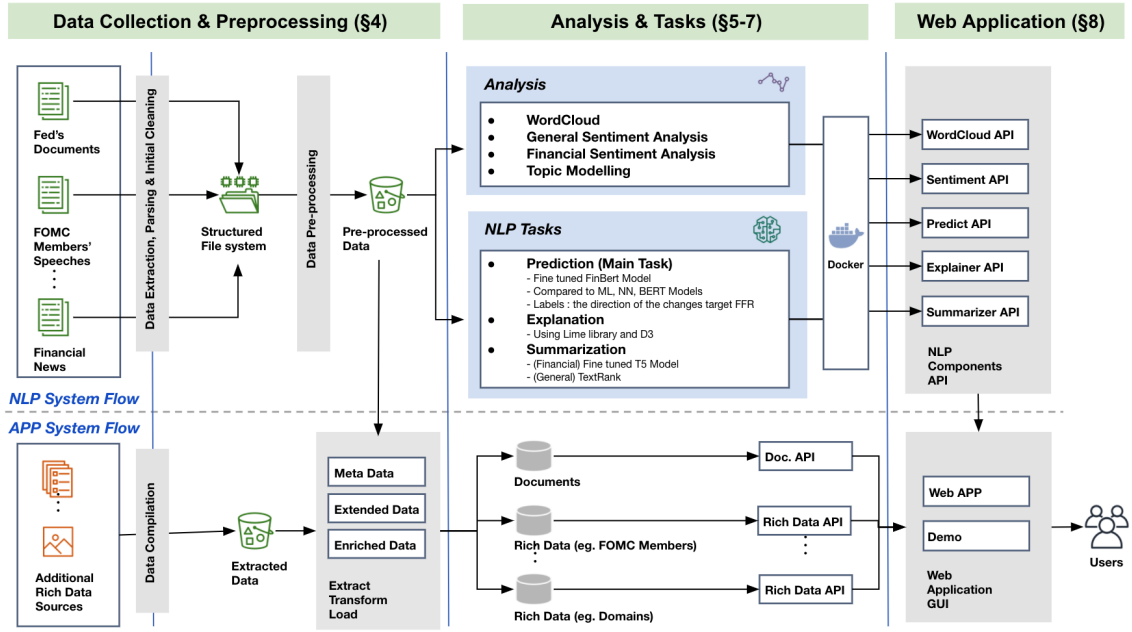


FIGURE 3.1: Overview of FedNLP System Flow. It has two main process flows of NLP (for modeling, python) and Application (for user interface, AngularJS). [1]

communications from over 30 websites related to changes in the target Federal Funds Rate (FFR). The FFR is the target interest rate that banks pay to borrow and lend from each other on an overnight basis. The FFR is decided by the Federal Open Market Committee (FOMC), a committee within the Fed, where the body members set the monetary policy and often speak to the media. This research collects the Fed's documents (e.g. post-meeting statements, minutes, and press conferences), FOMC members' speeches, and the financial news related to the Fed.

With the collected real-world data, this work implements eight widely used NLP components and algorithms (Table 3.1). The sentiment analysis, prediction, explanation, and summarization tasks provide a side-by-side comparison of generic and finance-specific algorithms and pose the question to end-users whether financial-specific algorithms are more capable of capturing the Fed's communication than generic algorithms. Furthermore, this chapter focuses on the prediction task that forecasts the direction of changes to the FFR (e.g. *lower,*

[1]The overall system flow consists of four activities – Data collection and preprocessing, Text Analysis (WordCloud, Sentiment Analysis, and Topic Modelling), NLP Tasks (Prediction, Explanation, and Summarization), and Web Application.

maintain, or raise) by analyzing the FOMC members' speeches and conducting extensive experiments. Ultimately, the proposed system is deployed in a web application to provide a simple and easy-to-access environment for end-users. The proposed system is evaluated through focus group interviews, user surveys, and post-experiment interviews.

The main contributions of this chapter are as follows:

- This research proposes the first interpretable, multi-component NLP system, **FedNLP**, designed to assist end-users in decoding Federal Reserve communications.
- This research presents an extensive study to predict the changes in the target Federal Funds Rate (FFR) using various machine learning, neural networks, and pre-trained language models.
- This research also includes a pilot study of human evaluation to assess which system components are most effective in helping end-users understand Fed documents.
- The proposed system is accompanied by a public demo to facilitate further development, and the dataset is publicly released for additional research.

3.2 Background and Related Work

3.2.1 The Fed and FOMC

The Federal Reserve (The Fed) controls the interest rate, specifically the Federal Funds Rate (FFR), to maximize the employment rate and achieve stability in the prices of goods and services in the US. Since the FFR indirectly impacts a very broad range of the economy, it is important to interpret the underlying factors that contribute to it. For example, when the FFR is low, banks generally lower their interest rates. Lower interest rates make it cheaper for people and businesses to borrow money. Mortgages, auto loans, student loans, and borrowing costs on credit cards are all indirectly impacted, all of which can affect an individual's financial position. In addition, lower interest rates can reduce the profitability of keeping money in bank accounts, potentially encouraging people to spend or invest rather than save.

Historically, when the economy shows signs of weakness, like during the Great Recession or the Covid-19 pandemic, the Fed typically lowers rates, and this type of decision is made by FOMC. As the FOMC has become more transparent with its communications [148] and has expanded its research interests to include the use of NLP, it is important to identify how NLP components could help end-users make better-informed decisions. This research focuses on FOMC members' communications, such as reports, press releases, and speeches, as they also hold certain importance and insights for the market [72, 58].

3.2.2 Leveraging Financial Documents for Prediction Tasks

In the financial industry, most market participants aspire to accurately forecast asset value and market behavior to make the smartest investment decisions. This is a challenging problem as financial systems are usually volatile and influenced by many factors [208]. To a large extent, past research efforts have focused on the use of time-series modelling and prediction techniques using historical pricing data [80]. With recent advances in NLP, it has become possible to harness novel sources of data [196] including unstructured textual data in the form of financial news [37, 205], financial reports [78, 140], earnings call transcripts [176, 135], and social media [10, 125, 192]. More recently, deep learning methods for NLP have become more common, and many papers have reported the use of neural networks for text-driven stock classification (or prediction) tasks [40]. Although stock market prediction has attracted a considerable amount of research, the Federal Funds Rate prediction remains an area of limited research. This chapter focuses on the FFR prediction, which has a significant influence on the stock market prices [8, 142, 106] by leveraging FOMC documents and FOMC members' speeches.

3.2.3 Pre-trained Language Models

Pre-trained Language Models (PLMs) such as BERT [36], ELMo [132], and T5 [138] are an approach to extracting knowledge from large-scale unlabelled data that has significantly improved performance on many NLP tasks. Unlike traditional word embedding [114, 131]

where words are represented as single vector representations, PLMs return contextualized embeddings for each word token which can be fed into downstream tasks [198]. This approach has been expanded to financial domain tasks, with several researchers releasing BERT variant models [198, 100] specifically designed for the unique language of finance. These models aim to bridge the large differences in vocabulary and expression gaps between the financial corpus and the general domain corpus. The prediction task involves fine-tuning the FinBERT model [198], a publicly available pre-trained model trained on 4.9 billion financial tokens.

3.2.4 Visualization Tools and Systems

Interactive analysis has been explored to understand Machine Learning performance. Several systems have adopted a black-box approach, focusing on user examination of inputs and outputs without relying on internal model workings. Many of these general-purpose systems prioritize visual inspection of model behavior on sample data, including ModelTracker [4], Prospector [79], Manifold [203], or What-If Tool [182]. For example, the What-If Tool provides rich support for intersectional analysis within a dataset, tests hypothetical outcomes, and focuses on ML fairness.

In linguistic tasks, visualization has shown to be useful tool for understanding deep neural networks such as LSTMVis [160], Seq2Seq-Vis [161], BertViz [170], ExBERT [62], or LIT [165]. Typical solutions include visualizing the internal structure or intermediate states of the model to enhance understanding and interpretation, evaluating and analyzing model or algorithm performance, and interactively improving models at various development stages, such as feature engineering or hyperparameter tuning through the integration of domain knowledge. However, these tools have primarily focused on developers and lacked the ability to handle long or complex industry-specific documents. This research proposes a visualization system designed for end-users to offer a holistic view of how NLP techniques analyze and interpret the Fed's communications.

3.3 Proposed Framework

The design concept of the proposed system is **usability**. Particularly how functional components can support **end-users**. As the system is to reduce a gap between the advancement of NLP technology and the needs for the use of NLP, this work defines an end-user as a person who works within a broad range of business sectors such as finance and accounting, often reads economic and financial news, and has low to no programming skills. By building a "practical use" NLP system, this work provides an environment where end-users can improve their understanding of Fed communications. The usability of the system is defined as follows:

- **System/Demo usability**: With no code and no technical support, an end-user can explore the dataset and get experience with the use of NLP in analyzing Federal Reserve communications.
- **Component usability**: Through multiple NLP components and visualizations, an end-user can get a holistic understanding of the Fed's documents.
- **Financial-focused component usability**: By using a financial-focused algorithm or by fine-tuned pre-trained models with financial textual data, an end-user can gain more legitimate insight than by using general algorithms.

Table 3.1 shows the multiple NLP components and the associated algorithms that have been implemented. For the financial component, Loughran and McDonald (LM) sentiment analysis, a lexicon-based method for economic and financial documents, is employed. Additionally, FinBERT and T5 models, fine-tuned with financial text data, are used for prediction and summarization tasks, respectively. The component requirements are gathered through the preliminary focus group interviews, which are explained in §2.12.

Fig. 3.2 shows the functional system flow that consists of NLP and Application modules required to deliver a no-code system to an end-user. In each NLP task, multiple models were built, and the final models were selected for web applications. In external data, FOMC documents and FOMC members' speeches consist of the main dataset. Guardian news is used

Component	Algorithm	Description	General	Financial
Sentiment Analysis	TextBlob [103]	Returns polarity and subjectivity using TextBlob for general settings.	v	
	LM Sentiment [104]	Returns polarity and subjectivity using LM sentiment for financial settings.		v
Topic Modelling	LDA [139]	Visualizes term clusters and topics in HTML using LDA model.	v	
Prediction	XGBoost [22]	Displays ML model predictions with explanation component.	v	
	FinBERT [198]	Displays model predictions using a fine-tuned FinBERT.		v
Explanation	Lime [141]	Visualizes top 10 highly-contributing features and highlights sentences.	v	v
Summarization	TextRank [113]	Displays an extractive summarization using a graph-based ranking model.	v	
	T5 [138]	Displays an abstractive summarization using a fine-tuned T5.		v
Demonstration	Decoupled APIs	Shows multi-components in one webpage that works with new input data.	v	v

TABLE 3.1: Multiple language processing components and algorithms in the proposed FedNLP. "General" denotes general algorithms and "Financial" denotes the financial domain-specific algorithms.

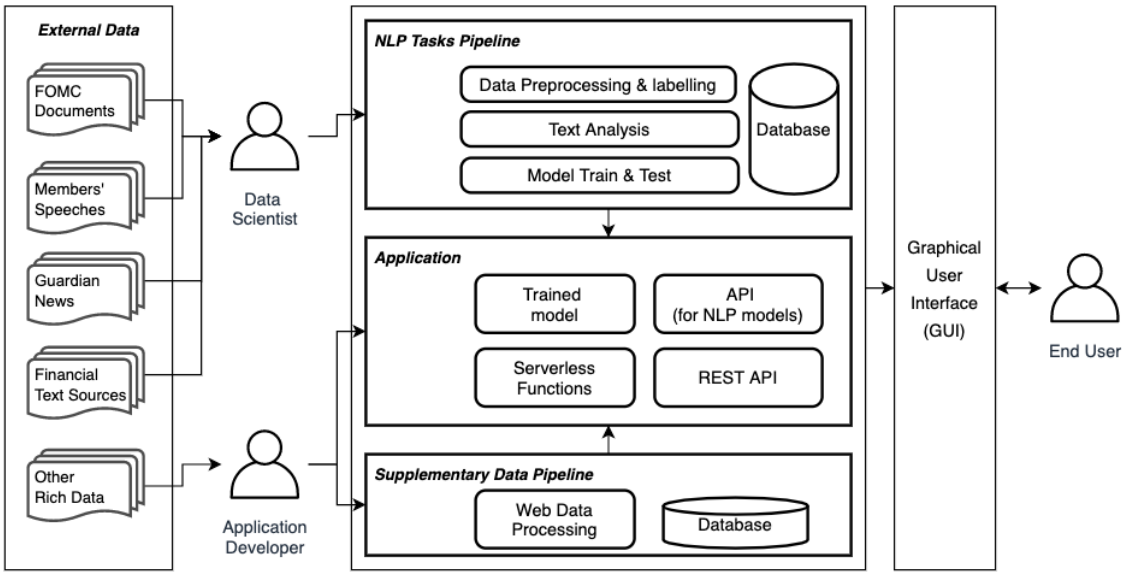


FIGURE 3.2: Functional System Flow of the FedNLP.

for the summarization task, and financial text sources are used for the prediction task. other rich data contains static content in web apps, including images.

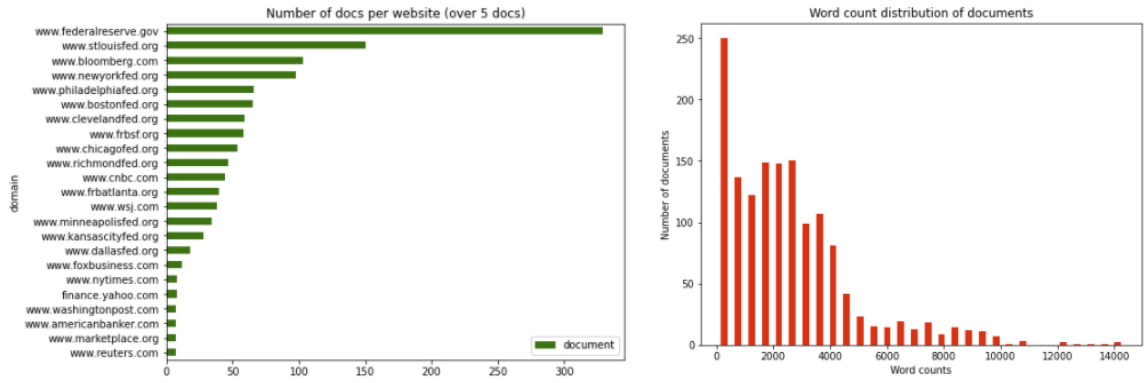


FIGURE 3.3: (L) Number of documents per source domain. (R) Word count distribution in total documents.

3.4 Dataset

3.4.1 Data Collection and Preprocessing

Generally, the FOMC meeting takes place eight times per year. The statement and press conference materials are released immediately and the minutes are published in three weeks. The FOMC consists of the seven members of the Board of Governors of the Federal Reserve System, including the president of the Federal Reserve Bank of New York, and four rotating presidents of the remaining eleven Reserve Banks, serving one-year terms each.

In this research, three post-meeting documents are collected from the Fed website. Moreover, the FOMC members' speeches are collected from over 30 different websites (Fig. 3.3). Each website and PDF has its structure, making standardized retrieval difficult. As a result, this work built a highly flexible, modular, and extensible data collection system that facilitates the retrieval of standardized structured data. Dataset generation contains 4 steps - 1) sourcing new documents, 2) parsing documents into the raw text format, 3) pre-processing the raw text, and 4) saving the data into the standardized format.

Step1. Document Sourcing: The St Louis Fed website maintains a canonical list of all FOMC members' communications. The list contains basic data such as date, the member's name, speech titles, and source URL. This work uses *Postman* to retrieve this data and save it in a structured JSON file. This research retrieved all data from January 2015 to July 2020.

.

Step2. Parsing: For each domain in the structured JSON file, this work analyzed the structure of the content and built a domain-specific parser (e.g. Bloomberg HTML parser). Using CURL, the research retrieved the content of each document and parsed it with the appropriate parser. PDF documents are all parsed with the PDF parser, and HTML documents are parsed with one of many domain-specific HTML parsers. To facilitate the simple and accurate raw data extraction, this research used *tika* to parse PDF content and *BeautifulSoup* to parse HTML. Each of the parsers was extended to ensure consistency and accuracy of the extracted raw content.

Step3. Data Preprocessing: The collected raw data files have a range of content structures and include unnecessary text and formatting. This work created a pre-processor that cleans and isolates the actual document content for a canonical source. To detect paragraphs, this research created a function to read the number of characters in one line and check the newline marker as well as the stopwords. In addition, all white space between the line and pagination text is removed. In the modelling stage, minimal data cleaning is performed to keep the documents' originality.

Step4. Saving: Once parsed and pre-processed, the raw data is saved with the Text Data Saver module into a single raw text file per document. The file is saved with a name and folder structure denoting the document title, domain, and date. Following this, the content is consolidated and bundled into two *pickle (.pkl)* files for use by the NLP models.

This research further collected news articles from The Guardian related to the Federal Reserve in the same date range as the main dataset. This data is used for the summarization task to train a T5 pre-trained model with financial text data from news headlines and content.

3.4.2 Data Labelling

Data labelling provides discrete values for the prediction task to forecast the direction of the change in the Federal Funds Rate (FFR). The prediction task can be formulated as a

multi-class classification. Let r_t be the FFR at FOMC meeting date t and let $r_{t-\Delta 1}$ be the previous FFR at the previous meeting date $t - \Delta 1$. Let L_t be the label of direction of the changes in the FFR at t and L_t is classified in the following way:

$$L_t = \begin{cases} 0, (lower) & r_t < r_{t-\Delta 1} \\ 1, (maintain) & r_t = r_{t-\Delta 1} \\ 2, (raise) & r_t > r_{t-\Delta 1} \end{cases} \tag{3.1}$$

All input documents $D_{\Delta 1}$ released between t and $t-\Delta 1$ gets L_t label as this research considers these documents assist the decision of r_t at FOMC meeting date t.

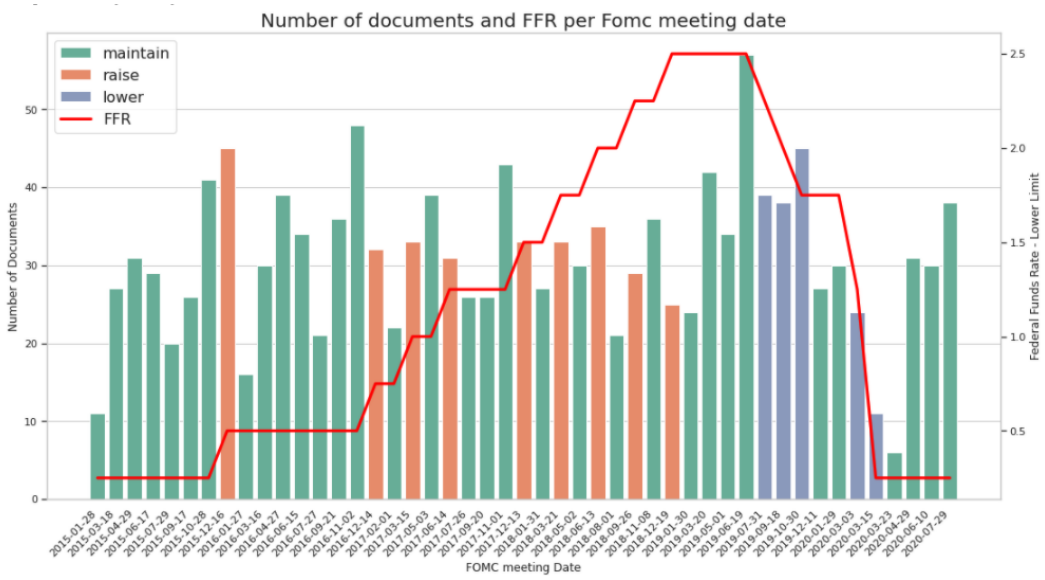


FIGURE 3.4: Number of documents (bar) and Federal Funds Rate (line graph) per FOMC meeting date.Each color in the bar graph represents annotated labels, showing the direction of changes in the Federal Funds Rate per FOMC meeting date.

Dataset	Train	Test
Number of vocabulary	36,813	10,616
Number of documents	1052	399
Avg. sentences in a document	149.62	22.64
Max sentences in a document	677	125
Ave. words in a sentence	21.97	22.56
Max words in a sentence	4,307	1,097

TABLE 3.2: Data Statistics of FedNLP. It includes FOMC documents and FOMC members' speech documents.

3.5 Analysis

3.5.1 Data Analysis

The collected data consists of 1,451 documents, made from a combination of the FOMC documents (122 documents) and FOMC members' speeches (1,329 documents) from January 2015 to July 2020. As the FOMC documents (statements, minutes, and press conferences) are released as a post-event to the Fed's decision, this research includes these documents only in the training dataset. The maximum sentence length is 677, and the maximum number of words in one document is 14,364. In order to minimize the computation loss and improve the model's training, this work set the number of words threshold per speech document to be under 10,000 and over 50 words while keeping the post-meeting minutes over 10,000 words in FOMC documents.

The collected documents are annotated as *lower, maintain* and *raise*, which represents the direction of change in the FFR from a single decision made on a particular FOMC meeting date. (Fig. 3.4) In total, the label distribution of the dataset was "lower" at 11% (157 documents), "maintain" at 69% (998 documents), and "raise" at 20% (296 documents), and this ratio was applied to training and test sets. The training dataset consists of 122 FOMC documents and 930 Speeches documents, while the test dataset only consists of 399 Speeches documents. The final dataset and its details are shown in Table 3.2.

3.5.2 WordCloud

In order to visualize topics within the data, this research created a WordCloud per label across the collected dataset. As seen in Fig. 3.5, words with the same frequency are used for all labels, indicating the analysis at the word level does not capture much information. Thus, this research used unigram, bigram and trigram visualizations for topic modelling that may capture further contextual information. As WordCloud gives a quick and intuitive understanding of the document, this research presents it in the web application for analyzing documents on the demonstration page.

FIGURE 3.5: WordCloud per label: (from left to right) lower, maintain, raise

3.5.3 Sentiment Analysis

An area of exploration in this research is the comparison of the effectiveness of general and financial domain-specific sentiment analysis methods. The system includes sentiment analysis which shows the attitude or the emotion of the writer (e.g. positive, negative, or neutral). For the generic representation, this work applies a TextBlob algorithm [103] that trains the data using *NLTK corpus* with a Naive Bayes classifier. For the financial representation, this work adopts the lexicon-based method for economic and financial documents, which was constructed by Loughran and McDonald (LM sentiment [104]). LM sentiment consists of financial word dictionaries appearing in corporate 10K/Q documents and earning calls. The polarity and subjectivity are calculated by the scores of the words present in the document as follows.

$$Polarity = \frac{Pos. - Neg.}{Pos. + Neg.} \tag{3.2}$$

$$Subjectivity = \frac{Pos. - Neg.}{N} \tag{3.3}$$

Where:

- $Pos.$ = the total number of positive words present in a document.
- $Neg.$ = the total number of negative words present in a document.
- N = the total number of words present in a document.

Polarity returns a score in a range of [-1, 1] where 1 means a positive attitude and -1 means a negative one, so this polarity score is called as a tone of voice in the system. Subjectivity returns a score in a range of [0, 1] where 0 refers to factual information (objective) and 1 refers to personal opinion (subjective).

With TextBlob, the polarity scores of most of the Fed documents were neutral or slightly positive. This is understandable as the Fed's tone of voice is often ambiguous because the FOMC members are mindful of the impact of their decisions on the market. Interestingly, LM sentiment was able to capture a more diverse tone of voice in the dataset. This lexicon-based method extracts positive or negative components, which indicate the speaker's position on the topic. The system uses the *pysentiment2* library to implement LM sentiment for analyzing the dataset. In the web application, each document page and a demo page display both TextBlob and LM sentiment analysis for comparison by the end-user.

3.5.4 Topic Modelling

Topic Modelling is used to discover various topics in unlabelled documents by grouping or clustering documents based on the words they contain. As documents on similar topics tend to use a similar sub-vocabulary, the resulting clusters of documents can be interpreted as discussing different topics. To discover the factors influencing the decision of the FOMC speakers, this research generated three subsets – a speaker-based aggregated dataset, a label-based (e.g. raise, maintain, or lower) aggregated dataset, and an FOMC members' speeches dataset.

The implementation of topic modelling involves four steps – (1) data processing, (2) optimizing the number of topics, (3) generating models for three subsets, and (4) displaying the results per subset and selecting the HTML output for the web application.

(1) Data processing for topic modelling requires more steps than the data collection stage. The *NLTK* library was used to further clean the data, including expanding contracted words, removing unwanted text, lowercasing, tokenization, additional stopword, and lemmatization in combination with POS tagging.
(2) The cleaned and processed dataset was used to create the id2word dictionary and term-document matrix by leveraging corpora from the *gensim* library. To achieve an optimal number of topics, the LDA models per subset with high coherence scores were taken to be the final model.
(3) The distribution of topics for each document was obtained by applying Softmax to document weights. The results of three subsets were visualized using the inter-topic distance map, top-30 most relevant terms per topic, WordCloud per topic, and t-SNE graph, thereby deriving some useful insights.
(4) Among the results, the inter-topic distance map from the Speeches dataset was selected to show the spread of topic clusters and displayed this enriched data using *pyLDAvis* [139] library and *iframe* in the web application.

Figure 3.6 displays the inter-topic distance map and top-30 most relevant terms per topic from the speaker-based subset. Each circle represents the different topic clusters. A similar position in the map means that topic clusters have similar keywords. It is observed that the speakers often said similar topics, such as banking, employment, and education.

Additionally, the experiments with the label-based subset show similar keywords in each topic cluster. Although the different keywords could be interpreted as the most influential topics in making a decision, it was difficult to distinguish differences among topics 0, 1, and 2, as seen in Fig 3.7 WordCloud results. The FOMC members' speeches dataset shows widely spread topic clusters compared to both a speaker-based subset and a decision-based subset. In Fig 3.7 t-SNE graph, the documents were dominated by topics 0 (blue), 1 (orange), and 2 (green)

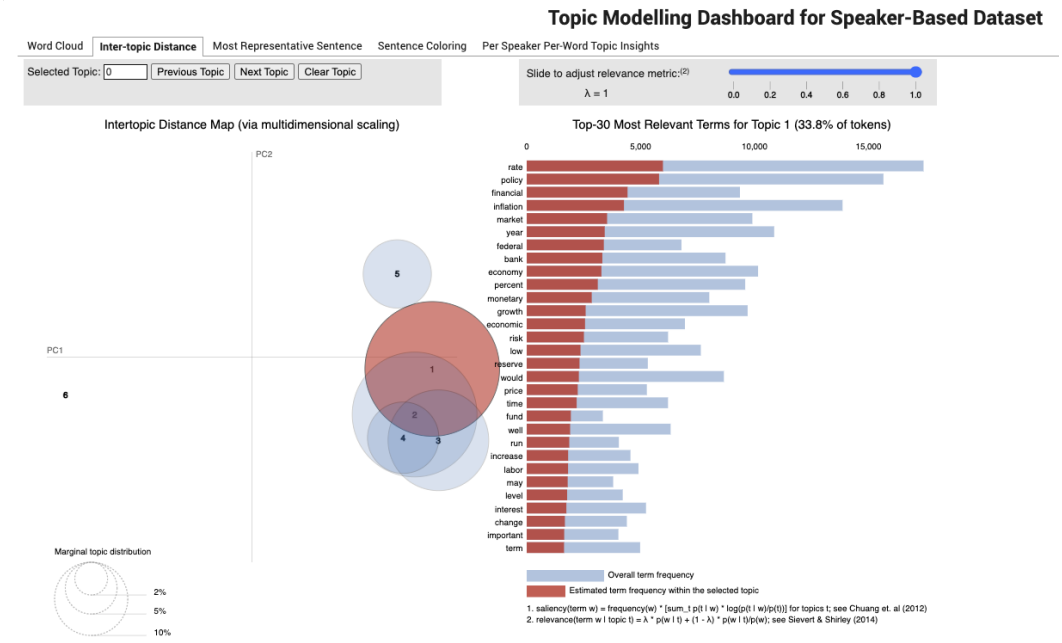


FIGURE 3.6: LDA graph from the speaker-based subset

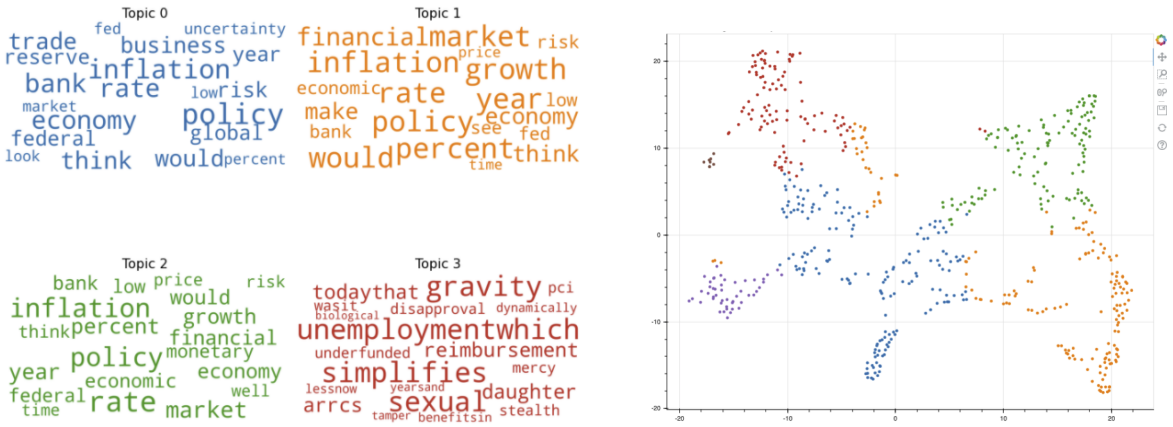


FIGURE 3.7: (L) WordCloud per topic from the label-based subset, (R) t-SNE graph from the FOMC members' speeches subset

and the keywords are similar over the topics such as "inflation", "policy", and "economy". This system implements the HTML outputs from the FOMC members' speeches dataset in the web application to provide an intuitive understanding of the collected documents.

3.6 Prediction

To facilitate the research objectives, this section predicts the direction of changes to the FFR (e.g. *lower, maintain, or raise*) by analyzing Federal Reserve communications. The FOMC documents are included in the training set as they are released immediately after the FOMC meeting. This enables the model to be trained on the textual representations of the FOMC's decision. The FOMC member's speeches, which form the pre-decision data, are divided into two groups consisting of 70% training and 30% testing data.

The research conducts comprehensive experiments, including traditional Machine Learning (ML), Neural Networks (NN), and pre-trained models, such as **SVM, Linear SVC, Logistic Regression, Random Forest, XGBoost, CNN, BERT, and FinBERT**.

3.6.1 Traditional Machine Learning Models

On SVM, Linear SVC, Logistic Regression, Random Forest, and XGBoost, both GloVe [131] and Doc2Vec [82] embeddings have been applied over the dataset. To obtain document features from word-level features, the word embeddings have been averaged. The overall F1 score was unsatisfactory since the obtained embeddings are no better than bag-of-words as contextual information is lost. Doc2Vec performed slightly better on the same models, with an increase in the F1 scores of the less represented classes. Apart from these embeddings, TF-IDF was also used as features in a separate experiment to understand the impact of rarely occurring words. In addition, as most ML models were detecting only *maintain* class, further experiments on up and down samplings were conducted to handle class imbalance.

3.6.2 Neural and Pre-trained Models

CNN

A Convolutional Neural Network (CNN) [77] for text classification with various NLP features is used (e.g. LM sentiment, TextBlob sentiment, TF-IDF, Part of Speech (POS) tag, Dependency Parsing). By changing hyperparameter conditions, the experiment set the model

with a convolution size of 128, kernel size 10, and max pooling layer of size 10. The models were run for 30 epochs with ReLU activation. As inputs, Glove and Doc2Vec embeddings were used. These inputs were 100 dimensions each and trained on 10 epochs. Among NLP features, the combination of LM sentiment and TextBlob achieved high accuracy (0.65) and high F1 score (0.55) in the CNN experiments.

BERT

This research follows a fine-tuned BERT [36] base architecture on the classification downstream task for the FFR prediction. The dataset collected from the FOMC and its members' speeches are converted into the input format for PyTorch implementation. The model has 12 Transformer blocks, 12 attention heads, and 768 hidden states for each position output. Each word is embedded into a vector size of 512, which means the length of the longest sentence. The embedding happens in the bottom-most encoder, and a list of vectors processes into a self-attention layer and then into a feed-forward neural network. The output of the bottom encoder flows upwards to the next encoders. Self-attention is the method the Transformer model uses. It pays attention to the relevant words in the input sequence for clues that can help enable better encoding for the currently processed word. The prediction task only takes the first position of the output vector that passes the special [CLS] token. The output from the Transformer model is used as the input of a simple linear layer with a softmax activation function.

FinBERT

Additionally, this research fine-tuned the FinBERT model [198], which is based on the BERT base model architecture and was pre-trained on a financial corpus of 4.9 billion tokens. As seen in Figure 3.8, the financial corpora are generated from the most representative resources in the finance and business domain – Corporate Reports 10-K and 10-Q from the Securities Exchange Commission (SEC), Earning Call Transcripts from the website Seeking Alpha, and Analyst Reports from the Investext database. The authors of the FinBERT model produce FinVocab, a new WordPiece vocabulary on the financial corpora using the SentencePiece library. This research uses an uncased version of FinVocab (30,873 tokens) for input word tokenization and pre-trained FinBERT for embeddings. This research keep the same setting

of BERT architecture to compare the impact of financial vocab and pre-trained FinBERT embeddings. The experiment shows that pre-training on financial-domain unlabelled data can improve performance on prediction tasks.

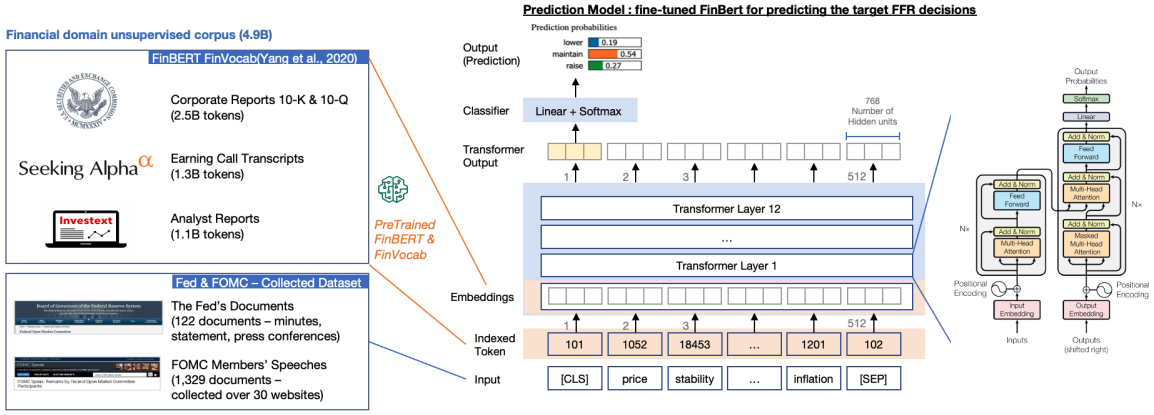


FIGURE 3.8: Prediction Model Architecture

3.6.3 Experimental Results and Analysis

Models	Features	Acc.	F1 *wa*	F1 *l*	F1 *m*	F1 *r*
SVM	TF-IDF	0.70	0.58	0.00	0.82	0.00
Linear SVC	TF-IDF	**0.73**	0.67	0.39	0.83	0.25
XGBoost	TF-IDF	**0.73**	**0.66**	**0.41**	0.83	0.17
Linear SVC	D2V	0.64	0.60	0.31	0.78	0.14
Logistic Regression	D2V	0.70	0.58	0.05	0.82	0.00
Random Forest	D2V	0.70	0.58	0.00	0.82	0.00
CNN	Glove + D2V	0.62	0.59	0.22	0.76	0.22
	Glove + D2V + Senti.	0.65	0.55	0.00	0.79	**0.55**
BERT	fine-tuning BERT	0.66	0.54	0.00	0.79	0.00
FinBERT	fine-tuning FinBERT	0.72	0.65	0.00	**0.84**	0.00

TABLE 3.3: Prediction performance comparisons in the imbalanced dataset, where D2V, acc, F1 *wa*, F1 *l*, F1 *m*, F1 *r* denotes Doc2Vec embedding, accuracy, weighted average F1 score, F1 for lower, F1 for maintain, F1 for raise class, respectively.

Table 3.3 presents the selected results with an imbalanced dataset for comparison. The two ML baseline models (Linear SVC and XGBoost) can detect *lower* and *raise*. Among all the experiments on ML models, Linear SVC with TF-IDF and XGBoost with TF-IDF features

achieved the highest test accuracy of 0.73 and weighted average F1 score of 0.66 while detecting all three classes. In Neural Network baseline models, the best setting on CNN and BERT base overfit and performs relatively poorly. FinBERT achieves a test accuracy of 0.72 but detects only maintain class which shows overfitting issues.

Sampling	**Models**	**Acc.**	**F1 *wa***	**F1 *l***	**F1 *m***	**F1 *r***
Up-sample	SVM + TF-IDF	0.32	0.16	0.00	0.49	0.00
	Linear SVC + TF-IDF	0.89	0.88	0.96	0.81	0.88
	XGBoost + TF-IDF	0.87	0.86	**0.97**	0.78	0.84
	BERT	0.84	0.84	0.93	0.77	0.84
	FinBERT	**0.92**	**0.92**	0.96	**0.89**	**0.94**
Down-sample	SVM + TF-IDF	0.30	0.14	0.00	0.46	0.00
	Linear SVC + TF-IDF	0.59	0.57	0.67	0.62	0.44
	XGBoost + TF-IDF	**0.70**	**0.69**	**0.75**	**0.70**	**0.64**
	BERT	0.50	0.54	0.57	0.63	0.26
	FinBERT	0.57	0.57	0.64	0.42	0.62

TABLE 3.4: Prediction performance comparisons in the up/down sampling

Overall, the document similarity per class increases the complexity of predicting the decision, as seen in Table 3.3. Moreover, most models have overfitting issues because of the imbalanced dataset. The results of both BERT and FinBERT also show that the BERT-style pre-trained model may not capture enough contextual information from long-form documents due to the limitation of 512 max sequence length. The prediction experiments provide the baseline results for further study and indicate what needs to be considered to deal with the Fed document, which is long-form and complex due to the ambiguous and esoteric nature of communication.

3.7 Other NLP Tasks

To ensure the legitimacy of insights and maximize the potential of NLP in the Fed's communications, this research further explores explanation and summarization tasks. While these tasks serve as supplementary modules within the system, future plans include developing advanced deep-learning NLP models based on comprehensive user feedback for each component.

3.7.1 Explanation Task

End-users frequently inquired about the trustworthiness of prediction models and their results. Unlike traditional time series forecasting methods that rely solely on historical numeric data to predict future trends, text-based prediction models offer a more comprehensible explanation that can be easily understood by individuals. For classification tasks, there are several tools to interpret the model's results, for example, LIME [141] or SHAP [107]. This research implements Local Interpretable Model-agnostic Explanations (LIME) that provide a visualization by using the classifier's output to generate a linear surrogate model. I selected two prediction models based on the evaluation results – XGBoost and FinBERT – and combined them with the LIME library.

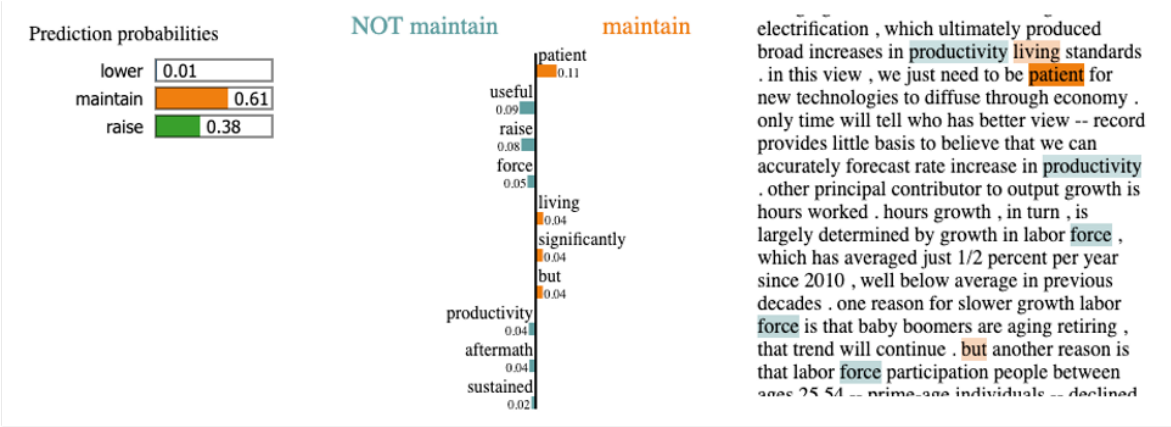


FIGURE 3.9: A prediction sample showing top 10 contributed words (with XGBoost and lime)

For example, Fig. 3.9 and Fig. 3.10 show the prediction results from the same sample data and its top contributed words for the prediction. The sample data is from a speech of Chair Powell on April 06, 2018, and the label of ground truth is *maintain*. Although both models correctly predict the same label, the weighted word visualization shows very different results. In Fig. 3.9, the probability of *maintain* label from XGBoost is 0.61, but the explanation result is not so clear in showing the relationship between the highlighted words (e.g. *significantly, but*) and the *maintain* label. As seen in Fig. 3.10, the FinBERT model combined with the LIME module in the Elie5 library shows a unigram and bigram visualization per label. In

Jerome Powell | Richard Clarida | Esther George | Neel Kashkari | Randal Quarles | James Bullard | Michelle Bowman | Patrick Harker

y=lower (probability **0.362**, score **0.613**) top features			**y=maintain** (probability **0.495**, score **2.063**) top features			**y=raise** (probability **0.143**, score **-1.065**) top features		
Contribution?	Feature	Value	Contribution?	Feature	Value	Contribution?	Feature	Value
+0.736	monetary	17.000	+0.431	inflation	31.000	+0.209	pace	5.000
+0.566	economy	14.000	+0.405	percent	22.000	+0.203	strong economy	4.000
+0.477	years	16.000	+0.383	participation	12.000	+0.182	people labor	2.000
+0.239	might	5.000	+0.354	unemployment	19.000	+0.171	data	6.000
+0.216	current	7.000	+0.295	measures	9.000	+0.170	balance sheet	3.000
+0.213	united	7.000	+0.272	labor force	12.000	+0.165	discussion	5.000
+0.205	participation rate	3.000	+0.257	factors	8.000	+0.164	support	7.000
+0.203	<BIAS>	1.000	+0.247	nearly	4.000	+0.161	people ages	1.000
+0.175	research	2.000	+0.194	period	4.000	+0.157	expected	2.000
+0.175	participants expressed	1.000	+0.176	united states	7.000	+0.144	strong	11.000
+0.173	longer run	5.000	+0.167	move	5.000		*... 594 more positive ...*	
+0.168	utilization labor	2.000	+0.161	states	7.000		*... 578 more negative ...*	
	... 527 more positive ...		+0.159	prime age	8.000	-0.145	percentage adults	1.000
	... 479 more negative ...		+0.157	inflation continued	2.000	-0.148	downward	2.000
-0.171	job market	5.000	+0.157	right	3.000	-0.149	may	4.000
-0.176	percent	22.000	+0.146	strong global	1.000	-0.163	business investment	4.000

FIGURE 3.10: A prediction sample showing top 30 contributed words (with FinBERT and Elie5)

this sample, the probability of *maintain* label from FinBERT is 0.49, and it captures more understandable results such as *inflation, unemployment, and labor force*.

The system implements XGBoost with LIME as the explanation module because of computation costs. For the deployment, further optimization of the explanation is required by isolating the result from the visualization library (D3). The choice of XGBoost and delivery optimization reduced the time to deliver a result from the explanation module from 10 minutes to 30 seconds.

3.7.2 Summarization Task

The summarization module provides key information about the document, giving users a simple way to quickly decide whether or not to read the full content. Automatic summarization gives a direct benefit because the Fed's documents are often long and complex. Automatic summarization has been studied for decades and there are two types of categories - extractive and abstractive. Extractive summarization aims to capture most information with the least redundancy, whereas abstractive summarization aims to generate new sentences to encapsulate the maximum gist of the input document. This research uses TextRank [113] and T5 [138] for extractive and abstractive summarization, respectively.

TextRank is a graph-based ranking model that identifies connections between text representations as graph vertices and implements ranking based on similarity measures. This research uses TextRank as a tool for general purposes because it does not require training corpora making it adaptable to any domain. Moreover, this research utilizes a fine-tuned T5 model for domain-specific financial summarization. T5, the "Text-to-Text Transfer Transformer," is designed for various NLP downstream tasks and was introduced alongside the Colossal Clean Crawled Corpus, a 745-gigabyte dataset of clean English text scraped from the web. To adapt T5 for financial text data, additional news data was collected and used for fine-tuning. Given the absence of ground truth for summarization in real-world data, this work assumes that a news headline serves as a summary of the information contained within the news body. In the case of the Fed data, due to the lack of readily available summarized information, the research conducted human evaluations on representative samples.

The Fed-related financial news was collected from the Guardian in the same range of period of the dataset. This study formulates a text-to-text framework and retains the common settings of a maximum sequence length of 512 and a batch size of 128 sequences. Due to the limitation of the maximum sequence length of 512, the model iterates the summarization function across the document.

3.8 Web Application

In order to provide a simple no-code experience, all components are delivered to end-users through a familiar interface - a web application. The web application consists of 3 separate systems: the end-user interface (GUI) is a simple Angular application, a REST API that delivers static content such as document summaries, interest rates, and date ranges, and an NLP API that serves as an endpoint for invoking ML predictions and NLP analysis functions.

Graphic User Interface (GUI)

The GUI is a lightweight web-based Angular application that provides a simple, intuitive interface for end-users to explore documents and related predictions. Following Angular design principles, the application architecture consists of components, such as the document

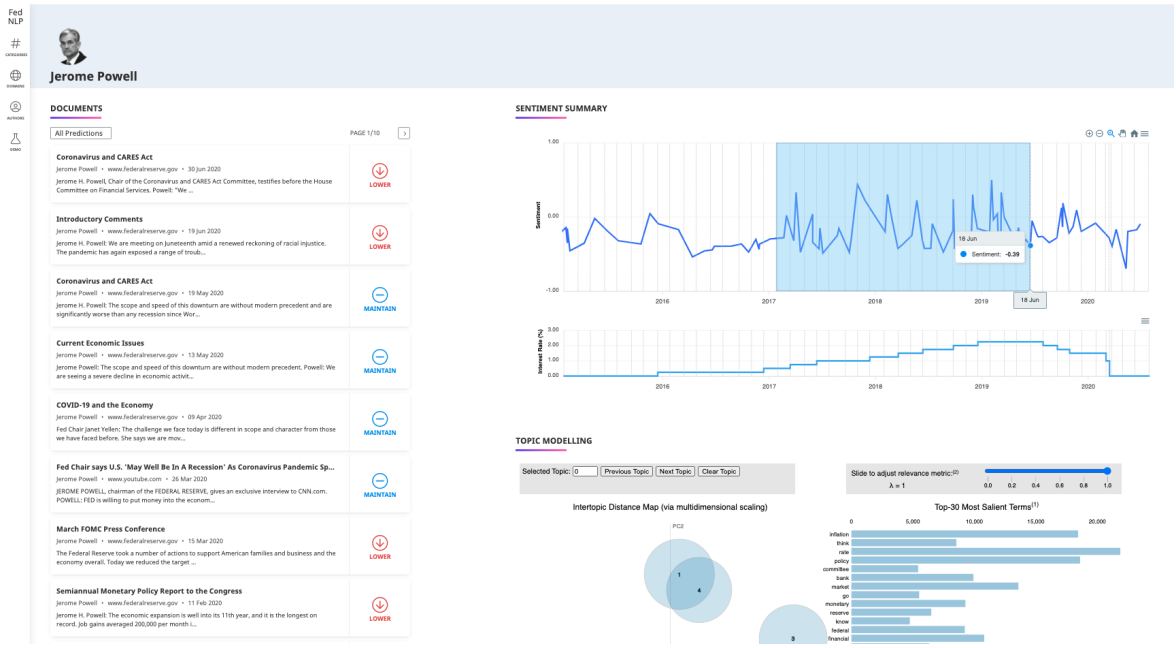


FIGURE 3.11: FedNLP Interface.

listing, document view and graphs, and page components for areas such as sections, demos, and home pages. Additionally, six data services modularise access to a single REST API or NLP API. These services make HTTP requests to an API endpoint and parse the response into JavaScript objects for use within components. The GUI is hosted on AWS, published to an S3 bucket, and deployed as a static website using AWS CloudFront for distribution and AWS Route 53 for domain resolution.

REST API

The REST API delivers static content in JSON format to the GUI. The API consists of five Node.js microservices (categories, domains, authors, documents, and document-extensions) deployed to AWS Lambda in a serverless configuration. Each microservice is a simple data-retrieval script that 1) Determines which object/s to retrieve based on path parameters, 2) Makes the call to the AWS DynamoDB database containing the static content, and 3) Formats and emits the response into JSON data. To improve the application performance and data load time, the document data is split into two endpoints; *documents* contains lightweight data used in lists and section pages, and the *document-extensions* endpoint returns full document content used on the document view only. A simple Node.js script extracts and splits document and

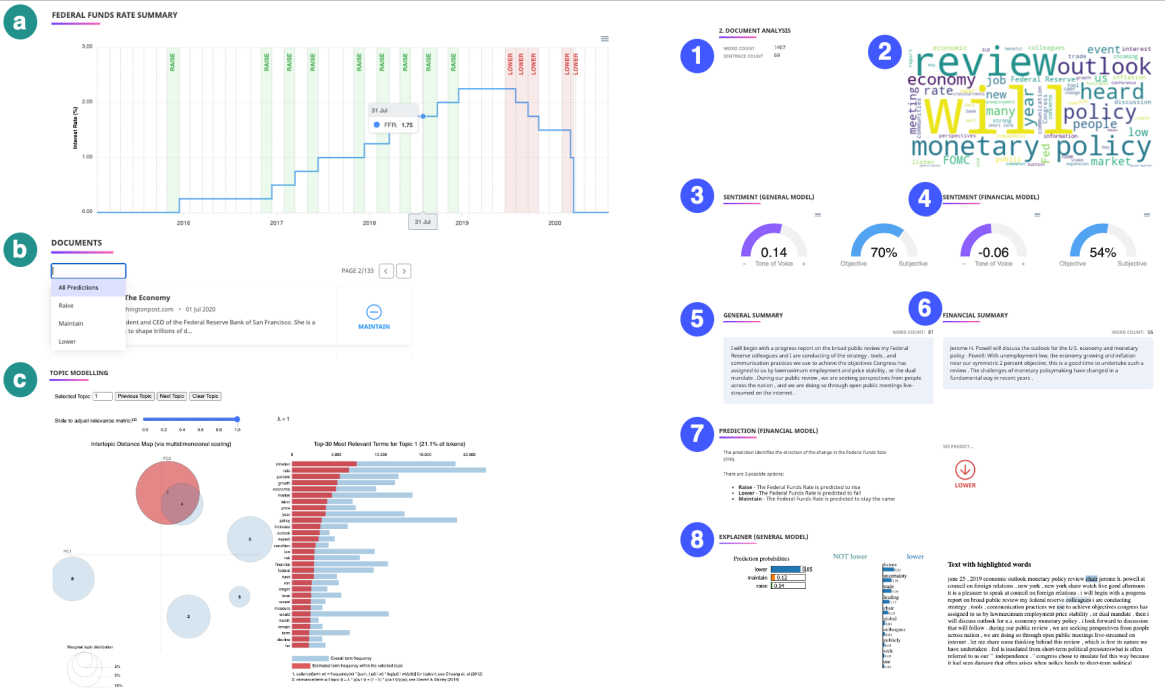


FIGURE 3.12: FedNLP System Components. [2]

document-extension content from the NLP pipeline document data. A dedicated DynamoDB load script uploads this data into the corresponding table. While category, domain, and author content were manually sourced, it was transformed and uploaded with the same DynamoDB load script.

NLP API

The NLP API is an endpoint for text analysis (WordCloud, sentiment analysis, and topic modeling) and NLP tasks (prediction, explanation, and summarization). The API itself consists of a simple Flask web server running inside a Docker container on an AWS EC2 server. The Docker environment initializes with all required libraries, including PyTorch and TensorFlow, as well as the configuration for Flask and network. On initialization, the trained and packaged model from the NLP pipeline is downloaded from an AWS S3 bucket onto the server. The Flask API exposes one route for text analysis or each NLP task, handles routing, data input parsing, execution of the underlying code, formatting, and emitting the response as JSON data.

[2] **(Left) Landing page components:** (a) Federal Funds Target Rate (Lower Bound) Graph and the Fed's Decision (b) a List and Filtering Function for the Fed's Documents (c) Topic Modelling Graph.

3.9 System Evaluation

3.9.1 User Study Design

This study conducted preliminary focus group interviews to identify system requirements and then performed surveys and post-experiment interviews to validate the usability of the proposed system.

Preliminary focus group interviews were organized by recruiting end-users who work in a broad range of business fields and have low to no programming skills. Focus group interviews started with a brief introduction to Federal Reserve communications and a semi-structured interview to understand the end-user's familiarity with NLP. This was followed by an in-depth discussion of the type of functionality that would be beneficial to improve the participant's understanding of the Fed's communications. From this discussion, this study identified multiple NLP components to implement in the system.

I recruited 20 participants from two different groups. The end-user groups consisted of 10 participants who worked in the finance, accounting, and banking sectors. The NLP research group consisted of 10 participants with a computer science background who were familiar with NLP models.

In order to compare and assess the usability of each component and the system as a whole, NLP researchers also completed surveys and post-experiment interviews. The survey itself started with a quick system walkthrough and an explanation of each component as well as providing the system evaluation criteria. To minimize random noise and make the process more streamlined, this research employed a within-subjects study [18] that enables each participant to measure all the system components. This survey question contains five sample documents corresponding to the FedNLP document detail components, multiple system components, and a demo page. To measure the reading time of original documents and summaries, participants were asked to take as much time as they needed.

(Right) Demo page components: (1) Word, Sentence Count (2) WordCloud (3) General Sentiment (TextBlob) (4) Financial Sentiment (LM) (5) Summarization (General, TextRank) (6) Summarization (Financial, fine-tuning T5) (7) Prediction (Financial, fine-tuning FinBERT) (8) Explanation (General, XGBoost).

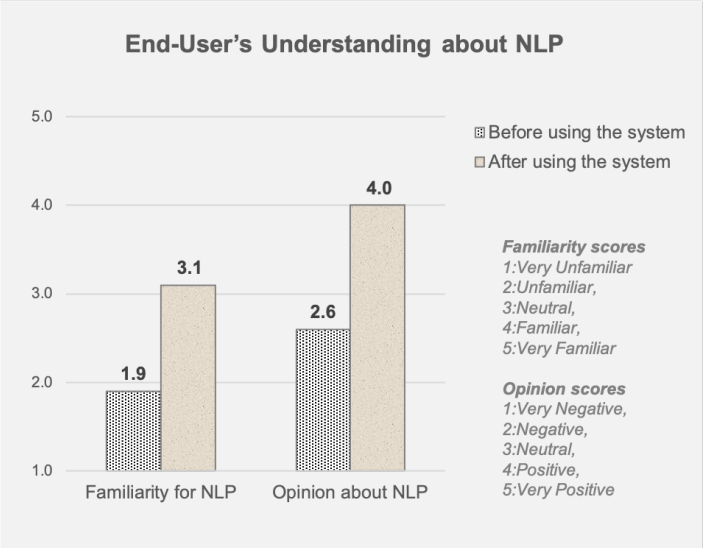


FIGURE 3.13: An average rating on before and after comparison of end-users understanding of NLP

Lastly, this research conducted individual post-experiment interviews and asked participants about their thoughts on the difference between the initial expectation and the experience while using the system. Additionally, a holistic evaluation of the system is asked for, as well as ease of understanding the terminology, familiarity with browsing the system elements, and intention to use the system in the future.

3.9.2 Focus Group Interviews

Gathered Requirements	**Components**
Inputting a new document with no code.	Demonstration
A past trend and visual analysis	Sentiment Analysis Topic Modelling
Comparison of different models.	Financial vs. General Algorithms
Getting an expectation about future trends from document	Prediction
Behind the reason for the model's prediction.	Explanation
A quick way of understanding document.	WordCloud Summarization

TABLE 3.5: Components gathering in focus group interviews

As outlined in §3.3, the research has focused on functional usability in this research. Initially, the end-user's familiarity with NLP is defined as it impacts the overall evaluation results. Fig 3.13 shows a comparison of the end-user's familiarity and opinions before and after using the system. Familiarity indicates how well the end-user knows the concept and application of NLP. Opinion indicates the end-user's attitude about the use of NLP in real-world applications. Most participants were unfamiliar with NLP, saying, *"We have heard the NLP term but do not know what it is"*. In addition, the majority had neutral to negative opinions about the use of NLP, saying that *"At some point, AI may outperform humans; however, not convinced that it is currently a reality."* After using the system, all of the participants responded that they are more familiar with NLP and have a much more positive opinion about the use of NLP. Additionally, in-depth interviews allow us to determine which functionality would be more useful in the proposed system. The results of the gathered requirements and system components in the focus group interview are shown in Table 3.5.

3.9.3 Surveys

This survey contained 3 main activities – 1) read and evaluate components for 5 sample scenarios, 2) browse the system, and 3) use the demo. For each scenario, participants were requested to read an original document, review the NLP components, and then rate the usability statements about summarization, sentiment, prediction, and explanation (Table 3.6).

System/Demo usability

Overall, the average ratings on the statements for each component are positive (ratings over 3) from both user groups. Regarding the **system as a whole**, both user groups expressed that browsing the system was easy to use and gave a positive rating on no technical support. Additionally, both user groups selected an FFR graph as the most useful among the system components.

Regarding the **demonstration**, all participants gave very positive feedback on ease of use and indicated a desire to use it in the future. As shown in Figure 3.12, a demo page contains eight different components, and all components were selected as useful components (ratings over

Components	Statements	NLP Researchers	End Users
System Workflow	This system is easy to understand.	3.88	3.50
	I need no additional technical support to be able to use this system.	3.88	3.60
Demonstration	This demo is easy to use.	**4.75**	**4.50**
	I would like to use this demo in the future.	**4.00**	**4.00**
Sentiment Analysis	I gain some insights from the general sentiment.	3.68	3.35
	I gain some insights from the financial sentiment.	3.68	3.65
	Having two different sentiments gives a better understanding than having only one.	3.38	3.75
Prediction	The prediction is an effective decision-support tool for the FFR.	**3.88**	**3.95**
Explanation	The explainer is an effective decision-support tool for the FFR.	3.25	3.55
Summarisation	The summary is readable.	**4.53**	**4.50**
	The summary contains the key information from the original document.	3.70	3.45
1:Strongly disagree, 2:Slightly Disagree, 3:Neutral, 4:Slightly agree, 5:Strongly agree			

TABLE 3.6: User Group's Average Rating on Evaluation Statements for FedNLP

or equal to 4) from the end-user group. In the NLP research group, WordCloud, financial summary, and financial model's prediction were rated over 4 indicated as useful components.

Component usability

For **prediction** (a fine-tuned FinBERT model), both user groups gave positive feedback (ratings over 3). In one of the scenarios, an incorrect prediction leads to lower ratings. For **explanation** (an XGBoost with TF-IDF model), both user groups also gave positive ratings; however, this was the lowest rated of all components. Participants in both groups identified that some unrelated highlighted words in the output led to low trust in the model. Regarding the **summarization** (a fine-tuned T5 model), all participants gave very positive feedback, highlighting that the summary is readable (ratings over or equal to 4) with the highest overall rating for usefulness. Differences in ratings for the summarization component exist due to the NLP research group placing more weight on key information extraction.

Finance-focused component usability

The system displays both generic and financial models for summarization, prediction, and sentiment analysis tasks. The adaptation of financial context representations results in a higher average rating than general settings, with the exception of the sentiment analysis.

3.9.4 Post-experiment Interviews

Overall, both user groups expressed that the **system** is *"impressive and interesting to use"*, which aligns with the research's purpose to generate a pilot study for analyzing Federal Reserve communications. The end-user group commented on the use of technical terminology, saying, *"Further explanation on topic modelling will assist users to better understand the model. Explaining the figures and the search tools would help the usability."* Additionally, both user groups selected an FFR graph as the most useful among the system components.

Regarding **component usability**, the fundamental understanding of NLP and domain knowledge may have affected the results. For example, one participant in the end-user group said, *"Some of the terms, such as ground truth, weren't clear what they represent."*. Another participant suggested the possibility of a prediction function combined with numeric market data, saying, *"As markets are complex adaptive systems, making any prediction in regards to the FFR direction would make sense in the context of existing market prediction."* Similarly, some participants in the end-user group said, *"The explainer is difficult to understand."* For them, I took a sample document from the survey and described how the top 10 keywords work with the model's prediction during the post-experiment interview. For example, in the speech of Chair Jerome Powell in May 2020, "pandemic" has value in maintaining decision whereas "coronavirus" does not maintain decision. It is because the Fed's documents used the "coronavirus" term often when lowering its target FFR, and the "pandemic" term was often used since the target FFR stayed in Zero Lower Bound (ZLB), which was in line with the "maintain" decision. Following this explanation, the participants commented that an explainer would be an effective decision-support tool for the FFR and suggested adding descriptions for end-users. In addition, the end-user group stated that summarization would be more useful for a long PDF document rather than a short document to help users decide whether to read it or not as a supplementary tool.

Regarding **financial-focused component usability**, the end-user group suggested having sufficient descriptions (e.g. *"Not sure what Tone of Voice means.", "Knowing the methodology behind each model would improve its trustworthiness."*) and changing user interface (e.g. *"Use*

another type of graph rather than a semi-circular gauge") would deliver better visualization and cause less confusion.

3.9.5 Lessons Learned

In summary, user evaluations demonstrate that the FedNLP system delivers legitimate insights through its multi-component structure, aiding end-users in comprehending the potential of NLP for Federal Reserve communications. The pilot study of the FedNLP highlights which components are effective and useful and provides a rationale for their benefits. One notable advantage of this study is its articulation of a potential application of financial NLP by offering a testing environment to identify how each component could help end users. Furthermore, the study suggests avenues for further system improvement, such as incorporating more detailed descriptions, modifying certain aspects of the user interface, and implementing curated search functionality to enhance user understanding of financial NLP applications.

Given the limited number of participants in this pilot study, I acknowledge potential limitations. Although post-experiment interviews helped to mitigate this, I intend to validate the findings through additional empirical studies. While a more extensive evaluation of the system is necessary to solidify the preliminary results, the outcomes of the pilot study are encouraging.

3.10 Conclusion

This chapter proposes the **FedNLP** system, which is designed to let end-users explore various NLP analyses and tasks to assist with decoding Federal Reserve communications. To the best of my knowledge, this system is the first of its kind to present the use of NLP in analyzing many forms of Fed documents, including post-meeting minutes, members' speeches, and transcripts. FedNLP enables end-users to get holistic insights on Federal Reserve communications through sentiment analysis and topic modelling. The system also provides capabilities for predicting the target federal funds rate decisions using pre-trained language model techniques and displays the interpretable results by highlighting the most important words from the original document. The visualizations allow end-users to see

an overview of the documents published on over 30 websites between January 2015 and July 2020 and the results from NLP tasks. Furthermore, the system also enables end-users to experiment with custom input through an interactive demo that presents multiple NLP analysis results automatically through decoupled, productized APIs. The future direction of this research is to enhance the system to give end-users full comprehension of NLP models. In practical use, FedNLP will be emphasized as a supplementary system to provide text analysis indicators from Federal Reserve communications and conduct further empirical studies in financial NLP research.

CHAPTER 4

Beyond Financial Sentiment Analysis: StockEmotions Application

---

This chapter presents the second financial NLP segment of the thesis, the StockEmotions application, a financial domain-focused dataset designed to investigate the potential of combining textual and emotional features to improve market prediction. This chapter is an extension of the work ***StockEmotions: Discover Investor Emotions for Financial Sentiment Analysis and Multivariate Time Series*** [84] accepted to the AAAI-23 Bridge. I formulated the research aim, collected the data, designed the methodology, analyzed the data, conducted the experiments, and wrote the whole paper.

While the application of NLP techniques in the financial domain and Information Retrieval (IR) research has gained increasing attention, limited resources remain a significant challenge. This chapter presents StockEmotions, a new dataset to retrieve emotions from social media content that includes useful information for financial sentiment and stock prediction. To address the broader impact of the dataset, this work collects raw data from StockTwits, curates the data by leveraging lexical search, extracts topic representation by using semantic search and hierarchical document embedding, and applies a multi-step annotation pipeline inspired by human and AI collaboration. The final dataset consists of 10,000 English comments and provides granular features for several IR research. To demonstrate the usability of the dataset, an in-depth dataset analysis and experimental downstream tasks were conducted. For financial sentiment/emotion classification tasks, DistilBERT outperforms other baselines. For multivariate time series forecasting, a Temporal Attention LSTM model combining price index, text, and emotion features achieves the best performance than using a single feature.

4.1 Introduction

Financial NLP is a rapidly expanding research field, garnering significant interest from both academic and industry professionals. Researchers are seeking to analyze financial sentiment collected from social media [31, 20, 189] or news [40, 84], and combine financial text mining with historical price data for stock market prediction [146]. In particular, public mood and financial sentiment play a significant role in investment decisions as is explored in behavioral finance [53, 200]; furthermore, social media messages have been studied as useful resources for detecting sentiment in NLP research [50].

However, existing datasets are (1) very limited in availability, mostly small, and even containing empty labels in the training set [108]. In addition, there is (2) no text data available that includes investors' emotions for stock market time series prediction. For example, existing studies use a proxy of public mood to predict the stock market [10] instead of extracting information from text data or applying an emotion taxonomy. The StockNet dataset [192] incorporates Twitter data and historical stock price, but their Twitter data is dumped text data without articulating emotions.

Inspired by behavioral finance [59], this research introduces StockEmotions, a dataset for emotion classification in the stock market, composed of 10,000 sentences collected from StockTwits. This dataset contains 2 financial sentiment classes annotated by users who share the comments as well as 12 emotion classes by leveraging a pre-trained language model (PLM) and finance experts. This work designs an emotion taxonomy associated with existing psychology studies [200, 90] to maximize the impact of the dataset in the financial domain. Figure 4.1 shows samples from the dataset. For example, the comment "$TSLA rocket man, does it again! Triple home run!🚀🚀🚀" has the *bullish* financial sentiment class, the *excitement* emotion class, and time series data and emoji data. Unlike the existing datasets [31, 108, 192], StockEmotions provides diversified emotions in the financial context as well as positive and negative sentiments.

In order to demonstrate the usability of the dataset, this research conducts a dataset analysis and presents baseline models for downstream tasks. For financial sentiment analysis, seven

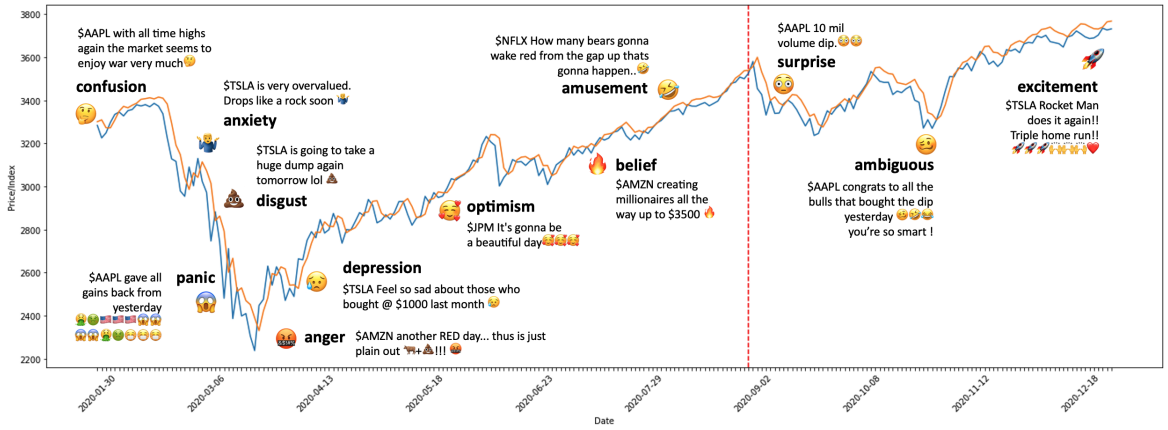


FIGURE 4.1: Example from StockEmotions dataset showing investor psychology on the stock market. A combination of input data (stock price index, text and emoji, and emotion label) is used on a Temporal Attention LSTM for multivariate time series forecasting.[1]

baseline models are implemented, including GRU [25], DistilBERT [144], BERT [36], and RoBERTa [99]. DistilBERT outperforms other baselines, achieving an average F1-score of 0.81 for financial sentiment classification and 0.42 for emotion classification, respectively. For time series forecasting, a modified Temporal Attention LSTM [183] is implemented in order to identify the importance of text and emotion for the prediction. When the model jointly learns from stock price/index, text, and emotion features, this model achieves the best performance on S&P 500 compared to having numeric input data only. The major contributions of this study are as follows:

- This research creates StockEmotions, a financial-domain-focused dataset for financial sentiment/emotion classification and stock market time series prediction;
- This research applies a multi-step annotation pipeline that brings the collaboration of human and pre-trained language model;
- This research demonstrates the dataset usability through downstream tasks and the impact of investor emotions for time series forecasting in particular.

[1] blue line = the actual S&P index, orange line = the prediction S&P index with a rolling window size 5, red vertical line = the point of data splitting; Further details are in the experiments section.

4.2 Related Work

4.2.1 Textual Datasets on Emotion Classification

Emotion classification is a subfield of sentiment analysis that involves identifying a set of discrete emotion categories based on a block of text or speech. It has been widely studied in NLP [1, 130] and IR research [14], with data collected from sources such as social networks[1, 115, 34] or dialog [63, 19]. Most existing studies contain annotations from two main emotion categories proposed by Ekman's 6 emotions [42] and Plutchik's 8 emotions [133]. Also, several existing datasets from social media focus on informal languages such as slang, emoji, and hashtags [44, 154]. In recent years, a multimodal dataset has received increasing attention with advanced neural network models that combine textual data with vision and/or audio.

A previous study proposes the largest human-annotated emotion dataset, GoEmotions [34], labelled for 27 emotion categories or neutral. In this work, researchers apply the new methodological approaches in psychology by building a granular taxonomy for language-based emotion space. The benchmark results show that BERT outperforms other baselines, and further research shows similar performance [3]. However, most of the dataset is limited in availability. Some only provide the trained model without original data, or their dataset link is now closed. In addition, existing datasets are collected from the general domain, which is not well suited for capturing emotions in the financial context.

4.2.2 Textual Datasets in Finance

The financial textual datasets have been used for information extraction in order to analyse valuable insights or support decision-making. This is achieved through several tasks such as financial sentiment analysis [31, 20] or event extraction [37, 209]. Furthermore, the extracted textual features are often employed to predict the stock market, which can be formulated as a classification task to predict upward or downward movements [192, 66, 146], a time series regression task to predict stock price changes [10, 50], a stock ranking task [48], or stock

Dataset Name	Domain	Data Source	Data Period	Data Size	Tasks	Avail.
EmoNet [1]	General	Twitter	Jul 2009 - Jan 2017	1.6 Million tweets	Emotion Classification (24 classes / Plutchik's8 classes)	N
SemEval18 Task1 [115]	General	Twitter	2016 - 2017	22k tweets	Emotion Classification and other 4 tasks (e.g. Intensity)	N
GoEmotions [34]	General	Reddit	Jan 2005 - Jan 2019	58k comments	Emotion Classification (28 classes, Ekman's 6 classes)	Y
SemEval17 Task5 [31]	Finance	StockTwits Twitter Financial News	Oct 2011 - Jun 2015 Mar 2016 Aug 2015 - Nov 2015	1,847 comments 1,591 tweets 1,780 news	Financial Sentiment Analysis (Score Prediction)	Y
FiQA18-SA [108]	Finance	Microblog News	Not specified	436 comments	Aspect-based Financial Sentiment (Classification, Score Prediction)	Y
StockNet [192]	Finance	Twitter StockPrice	Jan 2014 - Jan 2016	Not specified (26,614 target date)	Stock Prediction (Movement: up/down classes)	Y
MAEC [91]	Finance	Earning Calls (Text, Audio)	2015 -2018	3,443 instances	Stock Prediction (Return Correlation)	Y
EDT [209]	Finance	Financial News	Mar 2020 - May 2021	9,721 news (event) 303,893 news (pred.)	Event Extraction (11 event classes) Stock Prediction (Movement)	Y
StockEmotions [86]	**Finance**	StockTwits	Jan 2020 - Dec 2020	10,000 comments	**Emotion Classification (12 classes)** **Stock Prediction (Time Series)**	Y

TABLE 4.1: Comparison of StockEmotions with other Textual Datasets in the general (emotion classification) and the financial domain (broad tasks). (Avail. = Dataset availability as of Jan 2023).

returns correlation task [91]. These prediction tasks are drawn from various data sources, such as financial news [209], social media [192], and financial reports [91].

In particular, existing datasets collected from StockTwits [124, 92, 31] or Twitter [192] have relatively small size of data, and only focus on financial sentiment instead of considering in-depth emotion features in the semantic space. Recently, analyzing the investor's emotions has been further studied [41]; however, an online available dataset is rarely found. Also, most studies to predict the stock market movement using Twitter apply an investor mood index [10, 50] instead of extracting investor emotions from the collected text or disregarding emotions [192]. The comparison of existing datasets is summarised in Table 4.1.

4.3 Constructing Dataset

4.3.1 Data Retrieval

StockTwits is a social media platform similar to Twitter but focused on the stock market. The users share short comments about companies with special features such as a cashtag

(e.g. $TSLA) and financial sentiment (e.g. bullish and bearish) annotated by the users. This research uses the StockTwits APIs to retrieve content and metadata (e.g. company, datetime, userID, sentiment, and link) as allowed by the platforms' terms and conditions. This research creates a docker environment and uses `python-dotenv` to parse key-value pairs provided by StockTwits APIs. The collected raw data contains over 3 million comments that cover over 80% of the S&P 500 by market-capitalization-weighted. This research selects the S&P 500 listed companies as it represents more than 75% of the total U.S. stock market capitalization. The date range is between 01 January 2020 and 31 December 2020, which covers a roller coaster of investor emotion in the era of COVID-19 [120].

As shown in Figure 4.2, the platform is commonly used for promoting certain stocks or commercial websites, so the content has significant noise from commercial users. The quantity of content is skewed towards speculative or "meme" stocks, and the quality of content often contains toxic, offensive language and sarcasm. In addition, the emotion of a message can often be confusing as investors express the opposite sentiment based on their investment position. For example, investors who hold Tesla stocks express their anger when the stock has fallen; however, their sentiment is annotated as bullish by implying their emotions of hope or belief. Many existing financial sentiment datasets using StockTwits have not addressed the content quality issue described above.

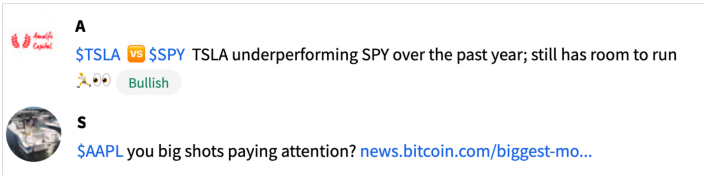


FIGURE 4.2: Samples of StockTwits. User may express their sentiment toward certain stocks or leave it blank.

4.3.2 Data Processing

Sentiment Filtering and Search Commercial Text This work filters the comments that include sentiment (e.g. bullish or bearish) annotated by users, and this reduces the raw data by 60%. Although it contains user biases, it is an effective starting point for retrieving high-quality data. To remove advertising or compromised data, I reviewed commercial users'

posts where the same comment appeared in several companies in a short period of time. Their comments generally include the terms of *join, visit, find, or chart* for increasing leads to their website. This work identifies usernames, retrieves content patterns, and generates the commercial pattern dictionary. Then, commercial users and comments were removed through the exact match method using lexical search queries.

Tokenization and Length Filtering The sequence length is limited to a maximum of 512 tokens to use BERT [36]. Also, this work selects comments over 3 tokens long using NLTK's word tokenizer and normalizes the token repetitions by having over 4 unique tokens long. For example, the comment *"$AAPL, short short short!"* is removed because it repeats the same token although the length is over 3 tokens. This length filtering is applied because short comments do not convey enough contextual information.

Masking The special characters for masking are used, such as a cashtag with a [CTAG] token, a hashtag with a [HTAG], or a website URL with a [URL]. When a comment contains only masked tokens, it is also removed. Through length filtering and masking, the raw data is reduced to 1.4 million comments.

Handling Emoji Users often express their emotions using emojis in social media, and therefore, this research narrows down the comments consisting of at least one emoji. For instance, in this comment *"$AAPL, 🎵 this is how we do it 🤑 🍏"*, it is difficult to detect investor's emotion with text only. Thus, this work converts emoji to its textual meaning using `emoji.demojize`. Also, a duplicated emoji in the same comment remains one short name. For example, multiple emoji of "🤣 🤣 🤣" is converted into one short name as "rolling on the floor laughing". It is applied to avoid skewed textual information due to repeated same emoji in the comments.

Other Filtering In order to have a fair representation of companies, this work randomly selects the comments to curate a maximum of 100 comments per company. This filtering process is intended to mitigate the influence of popular stocks that have a large volume of comments. In addition, comments including only one CTAG token are retained to identify

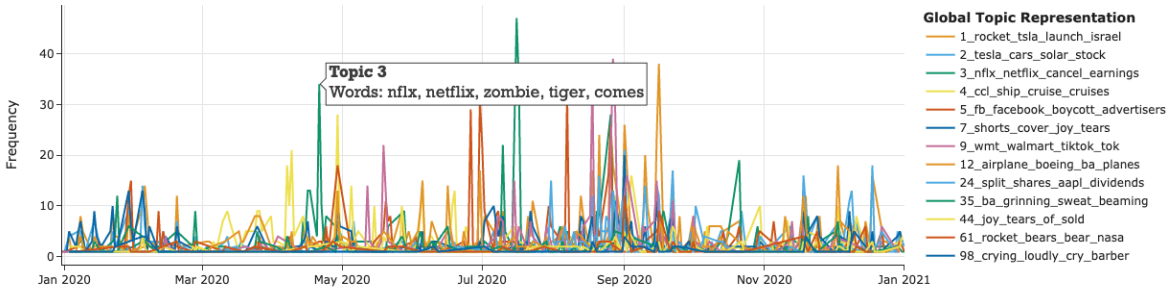


FIGURE 4.3: Topic modeling results for topics over time.

the intended target company of the comment. Following the aforementioned data filtering process, a total of 50,281 comments were obtained.

4.3.3 Topic Modelling

The objective of constructing the StockEmotions dataset is to classify fine-grained emotions in the financial domain that can be used for stock market prediction tasks. In order to understand whether the curated data represents emotions and to establish the sampling strategy for annotation, this work conducts topic representations using BERTopic [54], which extracts coherent topics with the class-based TF-IDF procedure.

Data Pre-processing For corpus analysis, numbers and punctuation were removed, while stopwords were retained in the comments. The pre-processing step is essential to ensure that the extracted topics are meaningful and relevant. Without this step, the topics show a large mixed representation, including numbers and symbols such as *"pg; your; 11256"*, which do not convey the intended contextual meaning.

Topics Overtime To analyze the changes in topics across different times, dynamic topic modelling techniques were used by fitting the BERTopic model. It generates a global representation of the entire corpus and then creates a local representation of each topic and timestep by calculating the class-based TF-IDF (c-TF-IDF), where the class c refers to all documents for each cluster as a single document. The c-TF-IDF replaces the inverse document frequency

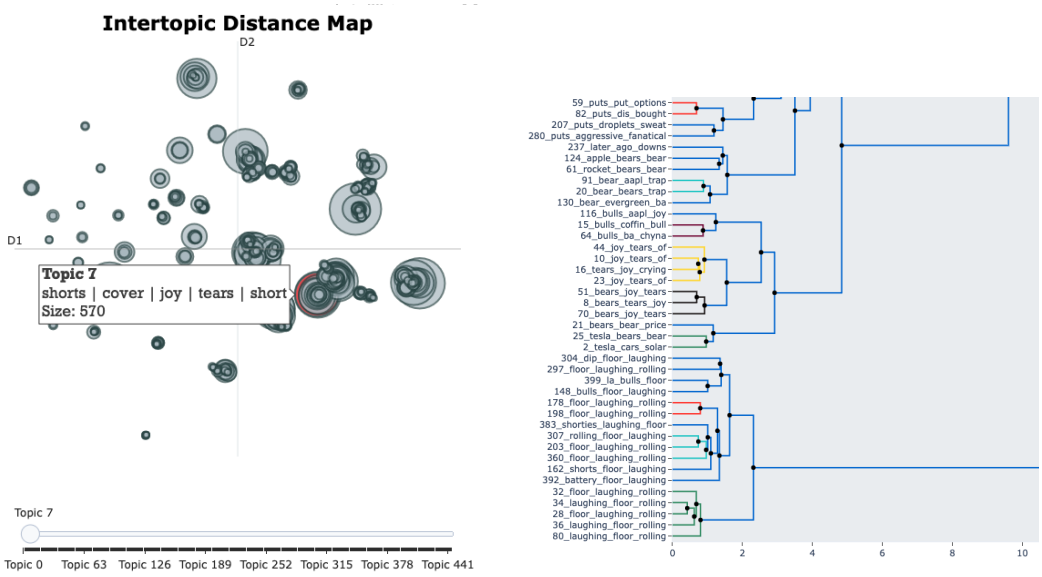


FIGURE 4.4: (L) Intertopic Distance Map and (R) Hierarchical Clustering

from the TF-IDF to the inverse class frequency acquired by the clustering. The c-TF-IDF is represented by the term frequency and the inverse class frequency:

$$W_{t,c,i} = tf_{t,c,i} \cdot log\left(1 + \frac{A}{tf_t}\right)$$

where t is the frequency of term, c is the collection of documents in a cluster at timestep i, and A denotes the average number of words per class. More details can be referred to the BERTopic [54].

As shown in Figure 4.3, certain topics present a sudden surge in frequency over a short period, indicating the occurrence of financial events such as earning release (e.g. topic 3), news of launching new factories, new products, boycotts (e.g. topic 1, 2, 5), or stock split (e.g. topic 24). Although topic modelling can offer insights into market trends and signals over time, it struggles to detect shifts in investor sentiment or emotions.

Other Topic Modelling Results This work acquires 447 topics from the 50,281 curated data. The model calculates the c-TF-IDF representation of the topics and reduces them to 2-dimensional space using UMAP. As shown in Figure 4.4, the (L) intertopic distance map presents topic clusters along with the corresponding size of each topic and its constituent

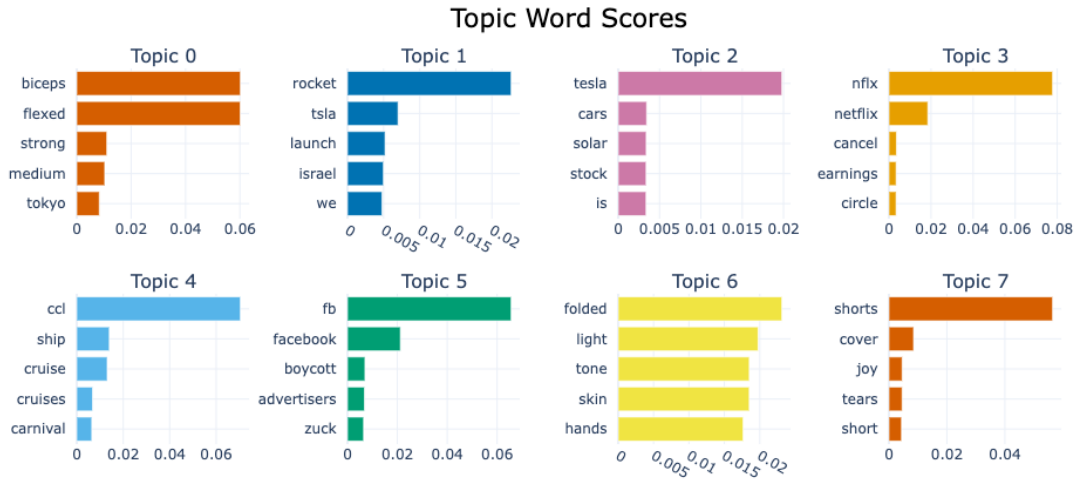


FIGURE 4.5: Word Scores per Topic

words. When hovering over the topic, the duplicated circle represents similar financial sentiment words such as short, joy, tears, or bear (e.g. around topic 7 circles). In addition, the (R) hierarchical clustering shows the structured topic representation of what would happen when combining topics. The converted emojis to their contextual representations were found to have a significant impact on generating topics and merging topics at higher levels of the topic hierarchy. Analysis of the (c) topic word scores displays which tokens are more likely to contribute to the corresponding topics, and company names are particularly prominent.

The results of the topic modelling analysis indicate that relying solely on semantic search to detect emotions in textual data may not be sufficient. This underscores the importance of implementing an annotation process. The distance map and hierarchical clustering results suggest that a reasonable number of emotion categories for the annotation process would be approximately 12-14 classes.

4.3.4 Data Annotation

Taxonomy of Emotions Inspired by the psychology of a market cycle [59], this work designs the emotion taxonomy. In contrast to existing emotion classes that consolidate positive

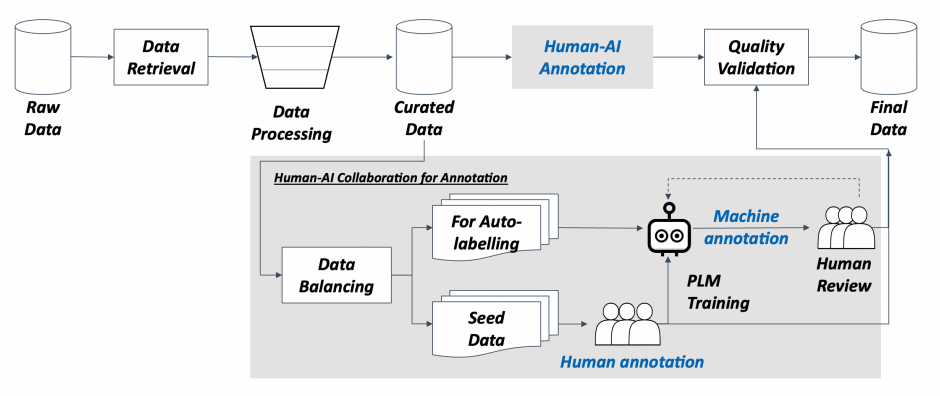


FIGURE 4.6: An overview of dataset creation pipeline.

feelings into "*joy*", it attempts to classify diversified mood proxies, including hope, optimism, belief, and thrill. In order to maximize the impact of the dataset, this reseatch links the taxonomy with existing studies in psychology that serve Ekman's 6-emotions [42] (e.g. joy, surprise, disgust, fear, anger, and sadness) and Plutchik's 8-emotions [133]. Examining an emotion taxonomy in behavioral finance and psychology returns discrete emotions, but some are too similar. To address this, a subset of the data was constructed, and two financial experts were asked to annotate it using a predefined set of emotions. I noticed that too similar labels such as *hope, wish, desire*, and *confidence* make the annotation task more difficult thus, similar items are combined into a single representative emotion label (e.g. *optimism*).

Annotation Guideline As shown in Table 4.2, emotion taxonomy, definition, synonyms, and related emoji are given to annotators. In addition, metadata, including user sentiment, DateTime, and target company, is provided to annotators in order to understand investors' intentions. The meaning of taxonomy is clarified through the seed round and final round of annotation. For instance, *ambiguous* is also used for strong self-satisfaction or bragging about their success in a stock market downturn to tease opposing investors (e.g. with sarcasm). The initial taxonomy contains more labels, such as *capitulation* and *denial*, but it is removed in the final round due to the small size of the data. If it is off-topic or unable to detect emotions, a *neutral* label is asked to annotate, and this research discards it for the final dataset.

Annotation Process To annotate the data, this research applies a multi-step pipeline inspired by human and machine collaboration [121], which brings the combined capabilities of a

Emotion	Definition	Synonyms	Emoji
amusement	the pleasure that you get from being entertained or from doing something interesting.	enjoyment, delight, laughter, pleasure, fun	😂
anger	a strong feeling of being upset or annoyed because of something wrong, unfair, cruel, or unacceptable.	rage, outrage, fury, wrath, irritation	😡
anxiety	a feeling of nervousness or worry about what might happen	nervousness, alarm, worry, tension, uneasiness	😨
belief	a feeling of certainty that something exists, is true, or is good, associated with the company's operation.	trust, faith, confidence, conviction, reliance	🔥
confusion	a refusal or reluctance to believe	scepticism, doubt, disbelief, distrust, uncertainty	🤔
depression	a state of feeling sad, extreme gloom, inadequacy, and inability to concentrate	sadness, despair, giving up, hopelessness, gloom	😭
disgust	a feeling of very strong dislike or disapproval.	loathing, dislike, hatred, sicken, abomination	💩
excitement	a feeling of having great enthusiasm, strong belief, intense enjoyment, or great eagerness.	enthusiasm, passion, cheerfulness, heat	🚀
optimism	a feeling of being hopeful about the future or about the success of something in particular.	hope, wish, desire, want, positiveness	💰
panic	a very strong feeling of anxiety or fear, which makes you act without thinking carefully.	horror, terror, fear, dismay, terrify	😱
surprise	a feeling caused by something that is unexpected or unusual. (e.g. earning surprise)	amazement, astonishment, shock, revelation	😳
ambiguous	unclassified emotions in the list or when the target of emotion is confused.	(subject to annotator's understanding of the text)	😕

TABLE 4.2: Emotion Definition provided to the annotators.

pre-trained language model (PLM) and human annotators. As shown in Figure 4.6, this research samples the seed dataset, which has balanced sentiment classes, and recruits three financial experts to annotate it. The taxonomy of emotions is updated based on feedback from annotators and empirical studies [59]. In the seed dataset (30% of the final dataset), this work achieves an average Cohen Kappa score of 0.79 [30]. This research constructs a multi-classification language model by fine-tuning BERT on the automatic labelling data that is proportionally selected from each month. For the final dataset, annotators are asked to choose either revise or agree on the automatically labelled emotion. This research updates the fine-tuning BERT for the revised label and iterates the process.

Data Quality Validation As this research applies semi-automatic labelling, the data quality validation process is performed by comparing the existing emotion classification method, GoEmotions [34], and evaluating some samples. The GoEmotions is a human-annotated

Number of Utterance	10,000
Number of Sentiment	2 - bullish (55%), bearish (45%)
Number of Emotion	12 - ambiguous(9%), amusement(8%), anger(4%), anxiety(14%), belief(9%), confusion(6%), depression(2%), disgust(13%), excitement(14%), optimism (16%), panic(3%), surprise(3%)
Avg. Length	19.2 tokens per utterance
Unique Emoji	761
Time Period	01 Jan 2020 - 31 Dec 2020

TABLE 4.3: Key statistics of StockEmotions. Each label shows the proportion in the total dataset.

dataset of 58k Reddit comments and classifies 27 emotion categories or neutral. When applying the GoEmotions APIs to the StockEmotions dataset, the results show that over 50% of content is categorized as neutral, and the following labels representing the most frequent emotions: amusement (8.36%), admiration (5.26%), and joy (4.40%). Additionally, this study sampled neutral-labeled data from GoEmotions and solicited evaluations from three human evaluators who had not participated in the initial annotation process. All evaluators agreed that the comments contained investors' emotions, suggesting that existing methods are insufficient for detecting emotions in the financial context, while these methods do.

4.4 Data Analysis

Table 4.3 shows key statistics for the dataset. The dataset provides 2 financial sentiment classes, 12 emotions, and time series data for the 10,000 comments. For the sentiment classes, this research makes a balanced dataset that consists of 55% of bullish and 45% of bearish. For the emotion classes, the distribution is imbalanced: *optimism, excitement, anxiety, and disgust* appear most frequently, whereas *panic, surprise, and depression* appear rarely. I also find that the users often share their wishes, hopes, or achievements even in the market downturn.

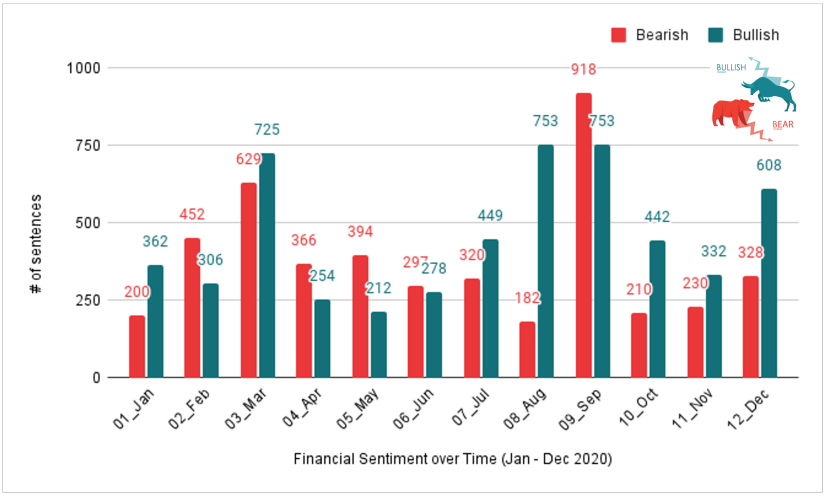


FIGURE 4.7: Financial Sentiment over time including bearish (negative, red) and bullish (positive, green)

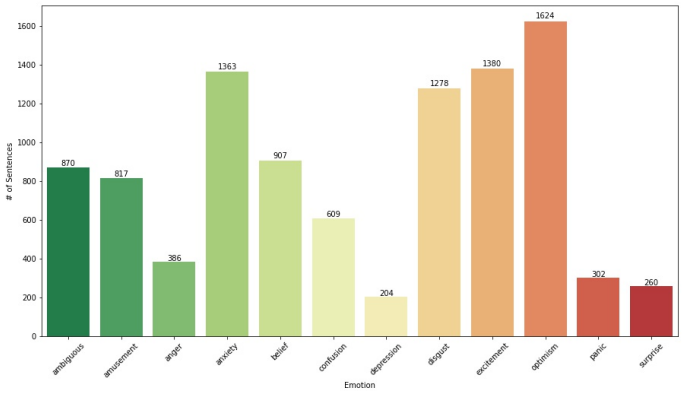


FIGURE 4.8: Emotions Distribution in the dataset from bullish to bearish market cycle.

4.4.1 Sentiment/Emotion Distribution

Figure 4.7 shows the distribution of financial sentiment over time. The bearish sentiment indicates a downward price movement, and the bullish indicates an upward price movement. Due to COVID-19, the stock market significantly crashed in March and September 2020, indicating a large volume of user comments. In addition, figure 4.8 shows the distribution of 12 emotions, showing imbalanced data. I find that the users often share their wishes, hopes, or achievements even in the market downturn.

Bear.	Count	Emotion	Bull.	Count	Emotion
😂	994	disgust	🚀	904	excitement
🤣	474	disgust	😂	683	excitement
🐻	162	anxiety	💰	304	excitement
🤔	147	confusion	🤑	294	optimism
📉	128	anxiety	🔥	276	excitement
🤑	121	anxiety	🤣	207	amusement
🤡	117	disgust	🐻	176	optimism
🔥	99	disgust	😎	164	amusement
💩	92	disgust	📈	159	optimism
💰	89	anxiety	🤔	157	confusion

TABLE 4.4: Top 10 Emoji Distributions associated with sentiment and the most frequent emotion.

4.4.2 Emoji Analysis

This research further investigates how emojis are distributed along with sentiments and emotions. Interestingly, six emojis appear in the top 10 emoji rankings in both positive and negative sentiment. For example, the emoji 😂 and 🤣 are used to not only express *excitement* but also convey laughing in *disgust*. In addition, investors share 🐻 to express their *anxiety* in a bearish market or *optimism* in a bullish market. For example, the sentence *"$TSLA may be part of S&P 500 driving pt 2000, beware* 🐻*"* contains *bullish* (positive) sentiment and *optimism* emotion for Tesla and delivers a warning to bear position investors.

Emoji Associations with Sentiment and Emotion Table 4.4 shows the top 10 emoji rankings in each sentiment and the most frequent emotion in each emoji. Investors often share 🐻 to express their *anxiety* in a bearish market or *optimism* in a bullish market. For example, the sentence *"$TSLA may be part of S&P 500 driving pt 2000, beware* 🐻*"* contains user annotation as bullish (positive) and optimism emotion for Tesla and delivers a warning to bear position investors.

4.5 Classification Experiments

4.5.1 Experimental Setup

For the classification task, this research conducts five independent experimental runs with different random seeds and reports the average performance results across all runs. All experiments were running on 16GB T4 Nvidia GPUs.

For classification baseline experiments, this research randomly splits the data into train/ validation/ test sets in the proportions of 80%/10%/10%. Also, this research compares machine learning methods, standard neural networks, and pre-trained language models as follows:

- **Logistic Regression**: a probabilistic-based algorithm to predict the class with the highest probability.
- **Naïve Bayes SVM** [175]: a variant of Support Vector Machines (SVM) using Naïve Bayes log-count ratios as feature values.
- **GRU** [25]: a gating mechanism in RNN, showing the empirical studies of better performance on less frequent datasets.
- **Bi-GRU**: a forward and a backward directional model consisting of two GRUs.
- **DistilBERT** [144]: a light version of BERT for faster training.
- **BERT**$_{base}$ [36]: a deeply pre-trained language model using bidirectional encoder representations from Transformer.
- **RoBERTa**$_{base}$ [99]: a robustly optimized method for BERT by removing the next sentence pre-training objective.

4.5.2 Grouping Emotions

For mapping Ekman's 6 emotions, this research uses a similar approach to GoEmotions [34]' mapping. Also, this research uses Plutchik's 8 emotions to group StockEmotions taxonomy as follows:

Model	**Sentiment**			**Emotion**												
(F1-score)	bear.	bull.	**avg.**	ambg.	amus.	angr.	anxt.	belf.	cnfs.	dprs.	disg.	exct.	optm.	panc.	surp.	**avg.**
LogitReg.	0.71	0.77	**0.74**	0.12	0.29	0.44	0.37	0.29	0.48	0.24	0.29	0.39	0.31	0.24	0.15	**0.32**
NBSVM.	0.71	0.78	**0.75**	0.10	0.27	0.45	0.30	0.36	0.46	0.21	0.34	0.42	0.29	0.29	0.22	**0.33**
GRU.	0.72	0.79	**0.76**	0.20	0.31	0.21	0.41	0.15	0.46	0.19	0.33	0.39	0.38	0.43	0.06	**0.34**
Bi-GRU.	0.73	0.78	**0.76**	0.22	0.33	0.49	0.39	0.30	**0.54**	**0.29**	0.39	0.43	0.32	0.41	0.06	**0.36**
DistilBERT.	**0.79**	**0.83**	**0.81**	0.12	**0.37**	0.56	0.42	**0.42**	0.51	**0.29**	**0.43**	**0.51**	**0.42**	**0.48**	0.21	**0.42**
BERT.	**0.79**	**0.83**	**0.81**	**0.27**	0.30	**0.59**	**0.46**	0.37	0.50	0.22	0.37	0.48	0.40	0.41	0.41	**0.40**
RoBERTa.	0.78	0.82	**0.80**	0.09	0.25	0.13	0.44	0.29	0.50	0.21	**0.43**	0.44	0.39	0.11	0.21	**0.39**

TABLE 4.5: F1-score$_{micro}$ results on the test set for classification task. Emotion labels are in alphabetical order; *ambiguous(ambg.), amusement(amus.), anger(angr.), anxiety(anxt.), belief(belf.), confusion(cnfs.), depression(dprs.), disgust(disg.), excitement(exct.), optimism(optm.), panic(panc.), and surprise(surp.)*.

- **Grouping Ekman's 6 emotions**: anger (maps to: anger), disgust (maps to: disgust), fear (maps to: anxiety, panic), joy (amusement, belief, excitement, optimism), sadness (maps to: depression) and surprise (all ambiguous emotions).
- **Grouping Plutchik's 8 emotions**: anticipation (map to: optimism, confusion), anger (maps to: anger), disgust (maps to: disgust), fear (maps to: anxiety, panic), joy (amusement, excitement), sadness (maps to: depression), surprise (surprise, ambiguous), and trust (maps to: belief).

4.5.3 Results of Financial Sentiment/ Emotion Classification

Table 4.5 shows classification results in terms of the F1-score obtained from baseline models. For **financial sentiment (binary) classification**, the results provide a robust baseline, achieving an average F1 score from 0.74 to 0.81 across the models. For **emotion (multi-class) classification**, DistilBERT achieved the best performance results (F1-score = 0.42) across the full taxonomy, similar to the BERT performance on GoEmotions (F1-score = 0.46) [34].

Table 4.6 and Table 4.7 show multi-class classification results, and BERT achieved the best performance results of F1-score = 0.48 for Ekman's 6 emotions grouping and F1-score = 0.42 for Plutchik's 8 emotion grouping models. These results indicate that emotion classification in this context is a challenging task with significant potential for future advancements. Due to the imbalanced data, I found that less frequent emotions, such as *ambiguous* and *surprise*

Ekman Emotion	Precision	Recall	F1
anger	0.54	0.64	0.59
disgust	0.40	0.38	0.39
fear	0.54	0.53	0.54
joy	0.74	0.78	0.76
sadness	0.17	0.21	0.19
surprise	0.46	0.40	0.43
macro-avg.	**0.48**	**0.49**	**0.48**

TABLE 4.6: Ekman's Mapping Results using BERT.

Plutchik Emotion	Precision	Recall	F1
anticipation	0.53	0.54	0.53
anger	0.45	0.44	0.44
disgust	0.36	0.58	0.45
fear	0.47	0.55	0.51
joy	0.53	0.54	0.53
sadness	0.24	0.32	0.27
surprise	0.44	0.03	0.06
trust	0.41	0.41	0.41
macro-avg.	**0.43**	**0.43**	**0.40**

TABLE 4.7: Plutchik's Mapping Results using BERT.

are likely to be confused. Furthermore, financial jargon and slang (e.g. ATH - All Time High; BTD - Buy The Deep; LMFAO - Laughing My Freaking Ass Off) may also generate confusion. For example, a positive market movement is represented by long, call, green, and bull, while a negative market movement is indicative by put, short, red, and bear. This type of jargon is frequently used across emotions, and it tends to be confused by the model.

4.6 Time Series Experiments

4.6.1 Methodology

Inspired by deep learning models in time series [66, 65, 183], this work implements a Temporal Attention LSTM with bidirectional encoder representations from transformers (BERT). As shown in Figure 4.9, the model learns temporally relevant information from

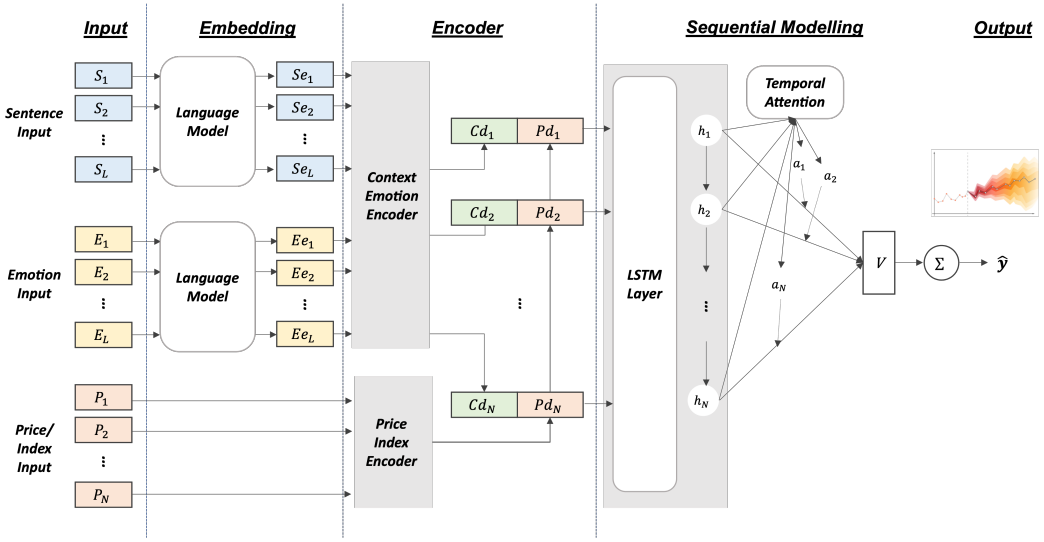


FIGURE 4.9: An overview of Temporal Attention LSTM.

sentences, emotions, and numerical data. To predict the price index of the S&P 500, the model incorporates the numerical modality from the price index and the contextual modality from sentence embedding using BERT as well as emotion embedding using GloVe [131] in a unified sequence. All embedding is encoded based on the intra-day sequence, returned as the context vector, concatenated with processed price index data, and fed into the Temporal Attention LSTM.

4.6.2 Experimental Setup

This Multivariate Time Series task is for investigating the relationship between stock prices, user-generated financial text, and emotion. To facilitate this analysis, a separate experimental setup was established, dividing the dataset by timestamp and integrating it with stock market data obtained through Yahoo Finance APIs. This work splits the data in the period of 01 Jan. 2020 - 03 Sep. 2020 for the training set (the former 67% of the total) and the data in the period of 04 Sep. 2020 - 31 Dec. 2020 for the test set (the remaining 33% of the total).

Evaluation Metrics For the performance evaluation, Mean Squared Error (MSE) is used, which measures the average squared difference between the estimated price and the actual

value.

$$MSE = \sum_{i=1}^{D} (y_i - \hat{y}_i)^2$$

Hyperparameter Search The hyper-parameter settings are as follows: a rolling window size [2, 3, **5**, 10, 15, 20, 25, 30]; hidden size [**25**, 50, 100]; and epochs [100, 150, 200, **250**]. A window size of 3 days and 5 days returns averaged the best performance, indicating that the financial contextual information in the real-world market would impact on the price/index in 3-5 days. The experiment results can be different based on the hyperparameter setting, as large text feature size causes noise for predictions.

Model	**window size = 3**			**window size = 5**		
hidden size	25	50	100	25	50	100
only index	1.13	1.53	1.83	1.15	2.07	1.99
+ text	2.18	2.30	1.49	0.89	1.32	1.53
+ text + emo.	**1.06**	**1.00**	**1.39**	**0.83**	**1.08**	**1.48**

TABLE 4.8: Mean Squared Error (MSE $*10^{-3}$) results on S&P 500 and StockEmotions using a Temporal Attention LSTM model for time series *(epochs = 250, emo. = emotion features).*

4.6.3 Results of Multivariate Time Series

Table 4.8 shows the S&P 500 index time series prediction in terms of MSE results across the different inputs. I found that the best-performing results (MSE = 0.83*10^{-3}) are on S&P 500 when combining price index, text, and emotion features with a rolling window size of 5. The experiment demonstrates that incorporating text features into the index enhances prediction accuracy compared to relying solely on numeric features. Furthermore, combining emotion features with text and numeric features achieves the best performance, highlighting the effectiveness of emotions in stock market forecasting.

4.7 Conclusion

This chapter presents StockEmotions, a financial-domain-focused dataset for financial sentiment classification and time series forecasting. StockEmotions dataset analysis and experiments on several downstream tasks show the usability of the dataset. In order to present the impact of text and emotion combinations, a Temporal Attention LSTM architecture is presented.

This research highlighted the benefits of utilizing multimodal data including text, emotion, emojis, and time series data. Building upon the initial research demonstrating the advantages of leveraging different modalities, future investigations will delve into specific details such as implementing joint learning at different parts of the network, determining optimal embedding sizes, assessing the impact of concatenation before or after the network, experimenting with window sizes, and conducting back-testing experiments. Future work will explore integrating various deep-learning models with Knowledge Graph-based models for stock market prediction tasks. Additionally, more baseline models will be researched to demonstrate performance variations when using text-only, text-price combinations, and text-emotion-price together.

Furthermore, the research showcased the utilization of annotation leveraging PLM capabilities. Given the impressive results demonstrated by LLMs, efforts have been directed toward LLM-based data annotation. This research will further extend to review LLM-based annotation techniques and collect recent data, scaling it to a larger scope.

Data Disclaimer The StockEmotions dataset is anonymized by regenerating sentence ID and follows terms and conditions from the StockTwits platform for collecting the data. The collected data belongs to the era of COVID-19, so it would represent different results considering a much longer period (e.g. 10 years of stock market prediction).

CHAPTER 5

Financial Document Understanding: FinDoc Application

This chapter presents the third financial NLP segment of the thesis, the FinDoc application, a visually rich document understanding task that leverages information retrieval with LLMs. This chapter is being prepared for submission to the AAAI 2025 and is under review. I formulated the research aim, reviewed the previous works, processed data annotation, checked the data quality, analyzed the data and the experiment results, and wrote the whole paper.

Document Understanding (DU) is a complex task that demands multimodal capabilities to detect layouts from documents and extract key information. In finance, the application for processing documents has received strong interest for its potential in reducing repetitive tasks and correcting errors. However, implementing Document AI (DocAI) models in real-world applications faces several limitations, including privacy concerns, a performance gap in handling unseen, unlabelled, and unstructured documents, and the usage of financial jargon. This chapter introduces FinDoc, a multimodal financial Document Understanding (DU) and Document QA system. The system addresses challenges related to privacy concerns and lower performance when relying solely on Large Language Models (LLMs). It presents combined capabilities by integrating LLMs with Retrieval-Augmented Generation (RAG). Furthermore, the system architecture is built on seamless modules to handle a variety of tasks, including Intent Classification and Slot Filling, Layout Analysis, and Key Information Extraction, utilizing pre-trained document AI models. In addition, this research outlines the challenges encountered in handling real-world documents in each module and provides a demonstration of an end-to-end QA system in a web environment, showcasing a replicable financial application.

5.1 Introduction

Recently, the development of pre-trained Document AI (DocAI) models achieved remarkable progress for Document Understanding (DU) tasks. This involves managing diverse unstructured/semi-structured PDF documents, detecting and parsing layout information across single or multiple pages, and extracting key information in the different layout entities. While Large Language Models (LLMs) or foundation models demonstrate multimodal capabilities in various tasks, their performance on DU tasks is inferior to pre-trained DocAI models. This indicates that LLMs struggle to effectively capture key information within documents despite demonstrating multimodal capabilities.

Various technology companies, such as AWS, MS, and Google, offer DU tasks for real-world applications, known as Intelligence Document Processing (IDP) services. IDP refers to the system that can automatically extract data from business documents and process the relevant data into actionable information. In financial services, a tremendous number of documents are generated in various formats, including unstructured, semi-structured, and structured. Accordingly, the efficacy of IDP has a significant impact on business processes by reducing repetitive tasks and correcting errors. For example, automated data entry in bookkeeping and content extraction from scanning receipts have become increasingly common.

While these services assist in accelerating the document-handling process, there are numerous barriers to implementing IDP in finance. Firstly, financial documents often contain sensitive internal and private data that cannot be used for model training and require exceptionally careful handling. Secondly, each company contains diverse document layouts and form structures, causing performance variations of pre-trained models. This creates a performance gap, where a pre-trained model that performs well on benchmark datasets exhibits poor performance when applied to real-world financial documents. Thirdly, financial jargon causes a barrier since most pre-trained datasets predominantly consist of general text and lack specific financial terms.

To address the above problems, this research introduces FinDoc, a multimodal Financial Document Understanding and Document Question Answering (DocQA) system. FinDoc

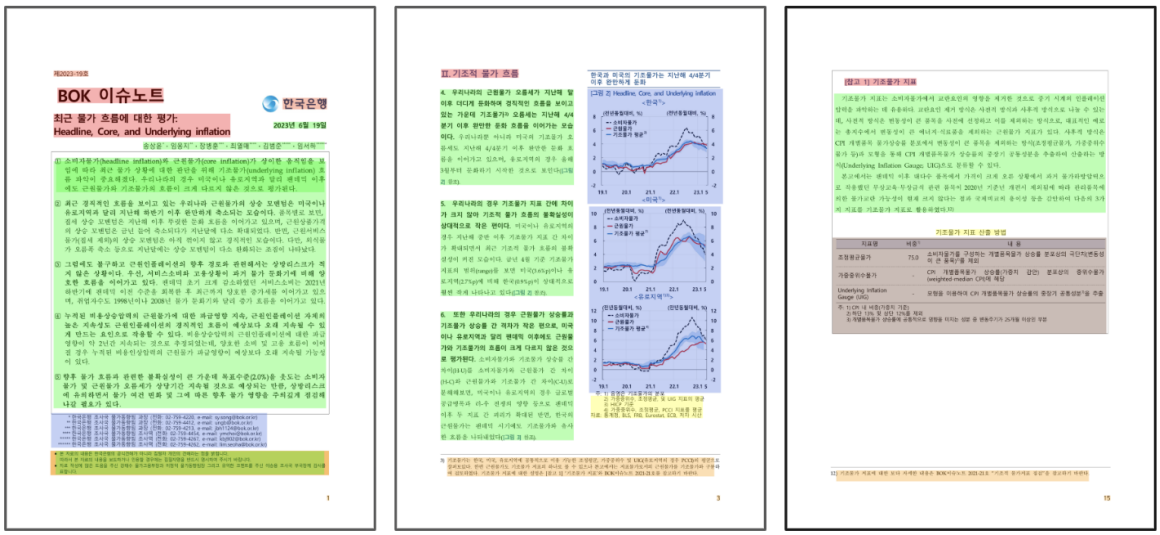


FIGURE 5.1: Examples of financial reports used in the FinDoc system. The color shows the results from the layout analysis.

combines the capabilities of LLMs with Retrieval-Augmented Generation (RAG), retrieving internal financial documents to address privacy concerns and enhance accuracy. The system architecture comprises four modules: document searching, document parsing, information retrieval, and summarization. The **document searching** module extracts user input queries using slot filling and intent detection and then utilizes a knowledge graph to match Top-k documents. The **document parsing** module employs pre-trained DocAI models to identify document layouts and parse document component entities. This enables the system to effectively handle noisy documents. The **information retrieval** module utilizes the parsed document entities as feature embeddings. The cross-modality encoder processes these embeddings and sends them to both the image search engine and the text search engine. The **summarization** module generates answers by leveraging LLMs with referenced document entities obtained from the image search and text search engines.

For the demonstration, this study uses financial reports from the Bank of Korea, which offers open access. These reports contain diverse layout structures across various types of documents, predominantly in Korean content. In the human evaluation, the FinDoc system achieved around 83% accuracy in identifying relevant documents to the question and generated highly coherent answers. The main contributions of this chapter are outlined as follows:

- This research introduces the first Financial Document Understanding and DocQA system that integrates the capabilities of LLM and Retrieval-Augmented Generation (RAG).
- This research demonstrates advanced techniques implemented throughout the entire system architecture, including Intent Detection and Slot Filling, Top-k search algorithms, Document Layout Analysis, and Key Information Extraction.
- This research conducts a multimodal model that enables handling text, layout, and visual features using a cross-modality encoder, managing unstructured documents in PDF format.

5.2 Related work

5.2.1 Document Understanding Tasks

Document Understanding (DU) includes document classification, document layout analysis, key information extraction, and visual question answering. These Document Understanding subtasks require a comprehensive understanding of the document structure, components, and content by integrating textual, layout, and visual modalities.

Document Classification (DC) [5] refers to categorizing documents into specific classes, such as receipts, invoices, or financial statements. **Document Layout Analysis (DLA)** [191, 190, 67] involves identifying document layout elements, such as text, tables, figures, and captions, through various combinations of different modalities. For example, bounding box information, which is rectangular outlines around objects, can be combined with contextual information. Alternatively, a combination of positional embedding, text embedding at the token level, and corresponding image embeddings can be employed. **Key Information Extraction (KIE)** [163, 172] involves extracting key content from document layout elements and generating logical relations, often in the form of key-value pairs. **Visual Question Answering (VQA)** [39, 111] refers to a task involving the extraction of queries from questions, searching for visually rich documents that correspond to the queries, and generating natural language

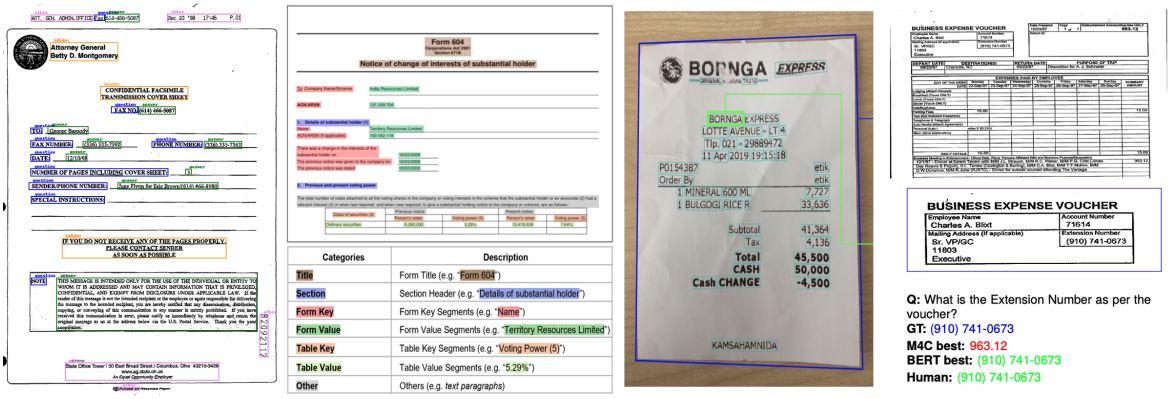
answers. This work focuses on DLA and KIE tasks. Additionally, Document QA is employed to generate answers instead of VQA, which requires searching for answers from the visual component.

5.2.2 Document Understanding Datasets

Various DU datasets have been collected from diverse industries and document types, including form, receipt, and report. For the DLA task, **FUNSD** [69] includes 199 noisy scanned form documents sourced from different domains, such as marketing, advertising, and science reports, related to US tobacco firms. Each document is presented in either printed or handwritten format, with 4 predefined labels, including question, answer, header, and other. For the KIE task, **CORD** [129] comprises more than 11,000 Indonesian receipt image and JSON pairs. It includes box-level text and parsing class annotations in two hierarchical levels. It has 8 superclasses, including store, payment, menu, and total, and 54 subclasses, including store name, store address, and store telephone. For the VQA task, **DocVQA** [111] comprises 50,000 questions and answers framed on 12,767 document images collected from five different industries, including tobacco and food. The question types include asking about date, title, total, amount, or name, such as *"What is the phone number in the voucher?"* The answer is the extracted content from the text, table, or figure.

The popular datasets collected from the financial services are FormNLU and FinTabNet. **FormNLU** [38] consists of 867 financial form documents collected from Australian Stock Exchange (ASX) filings. It includes three form types: digital, printed, and handwritten, and designed for both DLA and KIE tasks. For the DLA task, it includes 7 predefined labels (i.e., title, section, form key, form value, table key, table value, and others) to identify the document entity. For the KIE task, it involves identifying keys (e.g., Holder Name) with 12 predefined labels and extracting corresponding values (e.g., a string value of the substantial shareholder) from the form or table.

FinTabNet [207] comprises 89,646 unstructured PDF documents with 112,887 tables with detailed table structure annotations. The dataset is collected from earnings reports of S&P



(a) **FUNSD** (b) **FormNLU** (c) **CORD** (d) **DocVQA**

FIGURE 5.2: Examples of Document Understanding Datasets: (a) FUNSD [69], (b) FormNLU[38], (c) CORD[129], (d) DocVQA[111].

500 companies. Subsequently, FinQA [24] and ConvFinQA [23] datasets for financial QA have been further developed from FinTabNet. Unlike questions in DocVQA, Financial QA involves identifying numerical reasoning, as exemplified by asking like, *"Considering the weighted average fair value of options, what was the change of shares vested from 2005 to 2006?"* The answers to such questions require calculations based on relevant numbers extracted from both tables and textual content.

This research focuses on utilizing non-trained internal data through information retrieval. This research constrains the QA task to a simplified version, leaving VQA and Financial QA for future exploration. The benchmark datasets are reviewed to compare document layout structures, and pre-trained DocAI models were selected for the baseline.

5.2.3 Document Understanding Models

The DU tasks require the integration of advanced techniques from NLP, Computer Vision, and Information Retrieval. Transformer-based pre-trained DocAI models have become widely used, and each model employs various techniques to capture the cross-modality nature of text, layout, and visual features. The variations arise based on the model's emphasis on 1) the combination of features utilized (e.g., text and layout only, or all together), 2) the approach to combining these features in the network (e.g., during pre-training or fine-tuning),

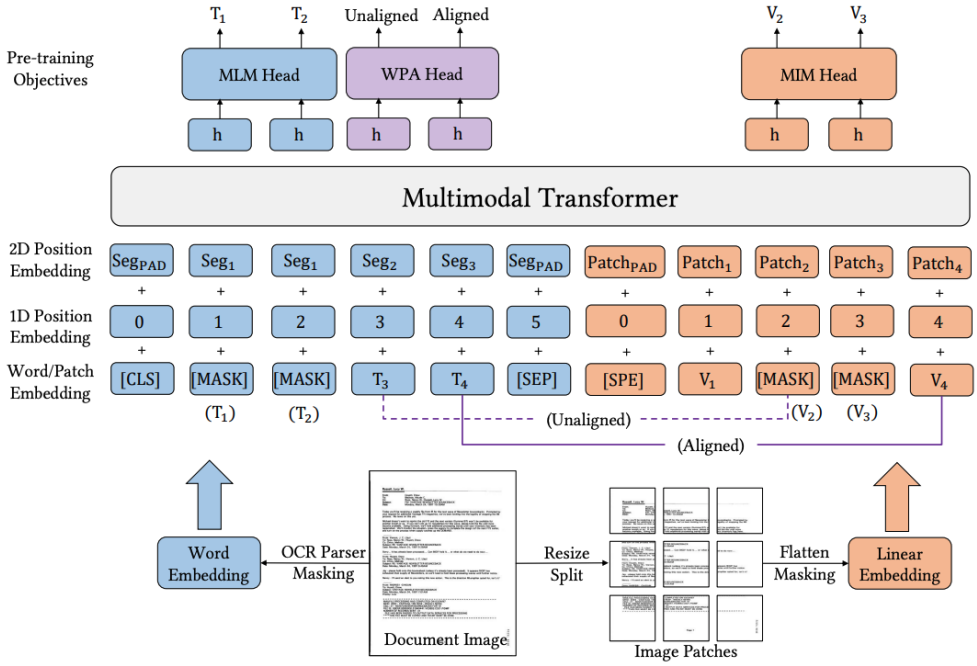


FIGURE 5.3: The architecture and pre-training objectives of LayoutLMv3

3) the selection of training objects, and 4) whether the model can be executed with reduced computational cost.

The **BROS** [61] model, an acronym for BERT Relying On Spatiality, uses a combination of text and layout (spatial information) features, excluding visual features. It encodes the sequence of token embeddings in 1D space with the corresponding relative positions between text blocks by normalizing the bounding boxes of each text block. The model adopts the area-masking pre-training objective in 2D space, where texts in a specific area of a document are masked and supervised. Without using visual features, the model reduces computational costs while achieving performance comparable to other DocAI models (e.g. LayoutLM [191]) that utilize all features.

The LayoutLM-series combines text, layout, and visual features, proposing various techniques to enhance joint learning and capture the cross-modality nature within a unified framework. The vanilla **LayoutLM** [191] model expands a 1D positional encoding of BERT to a 2D positional layout encoding (e.g., x0, y0, y1, y2) and integrates it with visual embeddings

during the fine-tuning stage. To improve the cross-modality interaction, **LayoutLMv2** [190] incorporates visual features during the pre-training stage through two objectives: text-image alignment and text-image matching. Text-image alignment is to align text lines with corresponding image regions, while text-image matching refers to learning within a specific matching area whether the document image and textual content are correlated. To reduce computational costs while retaining visual features, **LayoutLMv3** [67] proposes the unified text and image masking approach with three pre-training objectives: Masked Language Model, Masked Image Model, and Word-Patch Alignment. The Word-Patch Alignment objective refers to learning cross-modal alignment by reconstructing masked word tokens from text features with corresponding masked patch tokens from image features.

Furthermore, research in DU has extended to leverage LLMs, however, conventional LLMs are predominantly pre-trained on text-only features, and multimodal LLMs tend to be memory-intensive. To improve the performance of LLMs while avoiding the resource-intensive image encoders, **DocLLM** [171] is introduced. It follows the structure of the LLaMA2 model and uses both text and layout features. The spatial layout features are acquired through optical character recognition (OCR) and integrated using bounding box coordinates corresponding to the text tokens. Afterward, the model employs instruction fine-tuning methods utilizing 16 benchmark datasets across diverse DU tasks.

Category	**Model**	**Size**	**Modality**	**Dataset**		
				FUNSD (F1)	**CORD (F1)**	**DocVQA (ANLS)**
Pre-trained models	**BERT**$_{base}$	110M	T	60.3	89.7	63.7
	BROS$_{base}$	110M	T + L	83.1	95.7	-
	LayoutLMv2$_{base}$	200M	T + L + I	82.8	95.0	78.1
	LayoutLMv3$_{base}$	133M	T + L + I	<u>90.3</u>	<u>96.6</u>	78.8
	LayoutLMv3$_{large}$	368M	T + L + I	**92.1**	**97.5**	**83.4**
General LLMs	**GPT-4+OCR**	1T	T	37.0	58.3	<u>82.8</u>
	Llama2+OCR	7B	T	17.8	13.8	47.4
Fine-tuned LLMs	**DocLLM-1B**	1B	T + L	48.2	66.9	61.4
	DocLLM-7B	7B	T + L	51.8	67.4	69.5

TABLE 5.1: Comparison of model performance on FUNSD, CORD, and DocVQA datasets. The reported results are from the LayoutLMv3 and DocLLM papers.

Table 5.1 shows consolidated results on FUNSD for DLA, CORD for KIE, and DocVQA for VQA across various DocAI models. The results are sourced from the LayoutLMv3 and DocLLM research. Overall, the performance of LLMs and fine-tuned LLMs is inferior to Transformer-based pre-trained DocAI models. For example, GPT-4 (1T) with OCR and DocLLM (7B) achieved F1-scores of 37% and 51.8%, respectively, on the FUNSD dataset. In contrast, LayoutLMv3-large (368M) achieved a significantly higher F1-score of 92.1% on the same FUNSD dataset. In comparison to the DLA and KIE tasks, GPT-4 exhibits robust performance in the VQA task, achieving an Average Normalized Levenshtein Similarity (ANLS) score of 82.8%, close to the performance of LayoutLMv3-large (83.4%). This suggests that despite demonstrating multimodal capabilities, LLMs struggle to effectively capture key information within documents.

Furthermore, real-world financial PDF documents in English and Korean were tested on GPT-4. Recently, a common misconception has emerged that LLMs can provide solutions to any query. However, as seen in Fig 5.4 for DU tasks, GPT-4 generates incorrect answers for both tasks. In the left-hand side example, GPT-4 generates some random values but presents them as extracted information from tabular data. In the right-hand side example, the GPT-4 response includes Korea, even though it does not exist in the chart. These examples demonstrate how LLMs often produce generic or factually incorrect outputs, a phenomenon commonly referred to as "hallucinations".

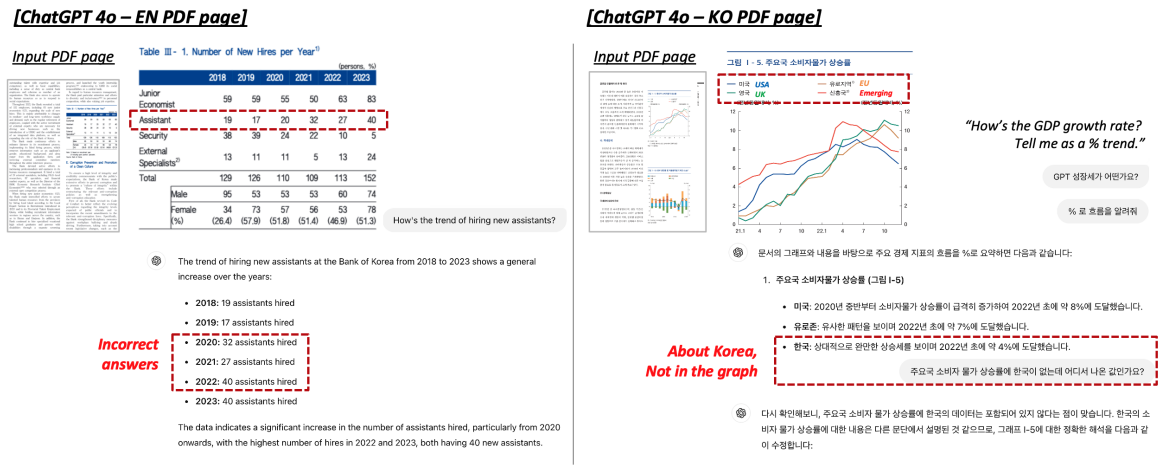


FIGURE 5.4: Examples of real-world financial PDF documents in English and Korean Tested on GPT-4.

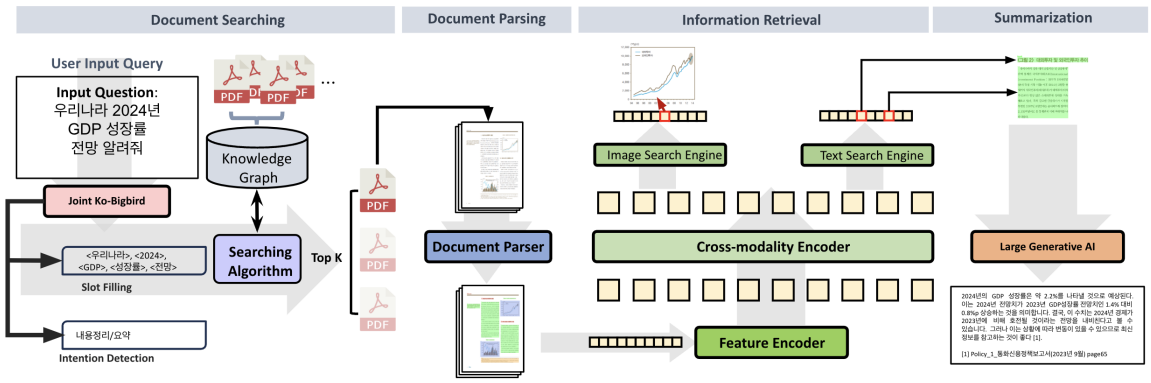


FIGURE 5.5: FinDoc system architecture. The system pipeline has four modules: Document Search, Document Parsing, Information Retrieval, and summarization

Given the complexity of DU tasks, a well-designed implementation is essential to achieve high performance while minimizing computational costs. This research employs Transformer-based pre-trained DocAI models for the primary tasks of DLA and KIE and integrates LLMs for the answer generation module.

5.3 Methodology

The research's methodology aims to demonstrate the capability of document question-answering tasks for internal financial documents. This research employs a variety of techniques, such as Intent Classification and Slot Filling, Layout Analysis, Key Information Extraction, and Retrieval Augmented Generation (RAG) using LLMs. The system architecture consists of four modules designed to extract key information from multiple internal documents: (1) **Document Searching**, which involves transitioning from a database containing multiple documents to a single page; (2) **Document Parsing**, progressing from a page to document component entities; (3) **Information Retrieval**, extracting key information from a document component entity using key-value pairs; and (4) **Summarization**, the final step involving the generation of answers with reference to the original document.

5.3.1 Document Searching

Document Pre-processing To utilize internal unlabelled documents, it is necessary to transform them from document repositories into a suitable format for document search. PDF documents with multiple pages are converted into a single-page image, and the associated metadata is stored in the knowledge base. Generally, real-world PDF documents lack metadata information. To address this limitation, the PDFs were converted to XML format and metadata, including key-value pairs, were generated. For instance, a key might be *"report type"* with the corresponding value assigned as *"Economic Review"*. Additionally, each stored document was transformed into vector embeddings using language models, incorporating textual features to enhance document search functionality. The textual content of the document is segmented into smaller chunks, typically the size of a paragraph. These embeddings are then stored in the vector database. Either vector embeddings, metadata, or a combination of both is examined, and the most effective option is utilized to identify relevant documents corresponding to the user query.

Query Generation This research employs Intent Classification and Slot Filling techniques to chunk user input questions. The input utterance is analyzed to detect user intent, such as "find the reasons" or "summarize" In addition, a sequence of tokens is utilized for slot filling, identifying meaningful slots through a combination of Named Entity Recognition (NER) and part-of-speech (POS) tagging. For example, in the user question, *"Summarize the main factors in changes of GDP in Korea in 2023"*, the slot-filling process involves tokenizing the query and transforming it into Named Entity Recognition (NER) tag format. For instance, *"Korea"* is assigned a location (LOC) tag, and *"2023"* is given a date (DAT) tag.

When relying solely on general NER tags, important information would not be detected in financial documents. To address this limitation, noun keyword tags were extracted using Part-of-Speech (POS) tagging. By selecting sample documents, Term Frequency-Inverse Document Frequency (TF-IDF) was implemented, which measures the importance of a term within a document related to a collection of documents. The top 50 noun words identified by TF-IDF reasonably represent important information, leading to the formation of noun

content	doc#	category	sub_categor	tags_conten	NER	
P0000516_200412_중국의 경제개혁과 북한(	0	file	name			
[' 중국', ' 대외경제정책연구원', ' 한국은행', ' 인	0	filtered	ner	tags	ORG	
[' 1998', ' 1982', ' 1999년', ' 1978', ' 1988', '	0	filtered	ner	tags	DAT	
['금강산', '상해', '개성공단', '개성공업지구', '나진	0	filtered	ner	tags	LOC	
[' 5%', ' 87%', ' (2)', ' (3', '7%', ' 17.4%', ' 79	0	filtered	ner	tags	PNT	
[' 1달러', ' 10', ' 140', ' 44원이', ' 1억 5천만달	0	filtered	ner	tags	MNY	
0.280410755	0	top50	tfidf	noun	국유기업	state-owned enterprise
0.2755756169	0	top50	tfidf	noun	개혁	reform
0.1956174002	0	top50	tfidf	noun	공유제	sharing system
0.1179809746	0	top50	tfidf	noun	경제개혁	economic reformation
0.1127121884	0	top50	tfidf	noun	설립	establishment

FIGURE 5.6: Examples for NER and TF-IDF with POS tag

keyword tags. With the addition of the TF-IDF noun tag, terms such as *"GDP", "factor"*, and *"change"* can be successfully detected in the above example.

Relevancy Search This research generates embeddings for the user query using the same language model used for document vector embeddings. Subsequently, a similarity search using RAG was conducted to identify the top-K most relevant document indexes within the embedding space. As an Information Retrieval framework, RAG retrieves relevant information from a knowledge base. This retrieved information is then used to augment the input of LLMs, thereby enhancing the accuracy and relevance of the text generated by LLMs. After identifying the relevant indexes, documents corresponding to K=2, and K=5 are retrieved and examined to select the most suitable candidates of K.

5.3.2 Document Parsing

This research utilizes pre-trained DocAI models for the DLA task, which involves parsing the searched documents into document layout component entities on each page. As shown in Fig 5.1 examples, the layout components include elements such as title, body, figure, table, caption, footnote, and more.

The initial application using pre-trained DocAI models returns low-quality results, mainly because of the disparities between the benchmark dataset and real-world documents. Moreover, the models fail to detect the logical structure between related layout components, presenting challenges in handling relation-sensitive questions or queries. To address these limitations,

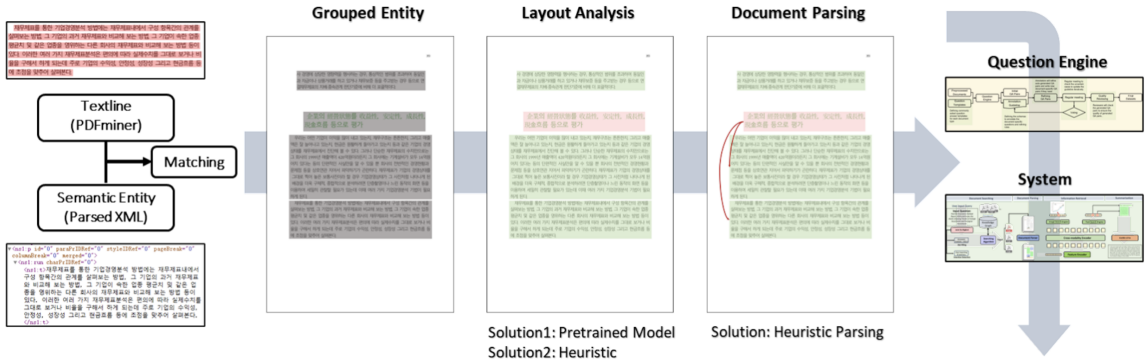


FIGURE 5.7: Document Parsing Process Workflow

few-shot learning techniques were employed by sampling a subset of documents for thorough annotation. This approach assists the model in acquiring prior information in the financial domain. Next, the bounding box coordinates of each layout component were extracted, and geometric information was gathered from the unlabelled PDF documents.

Subsequently, the text content was extracted using EasyOCR, an open-source Python library that employs the CRAFT (Character Region Awareness For Text detection) [7] model for object detection and the CRNN (Convolutional Recurrent Neural Network) [153] model for sequence recognition. Furthermore, to handle lengthy documents, this research investigates a section retrieval model that limits the token length to either 2048 or 4096 tokens.

5.3.3 Information Retrieval

The document layout components derived from the parsed documents are used as input features for the Key Information Extraction model. This research utilizes an additional pre-trained DocAI model that can handle cross-modality, integrating text, layout, and visual features. The input features are processed by the cross-modality encoder, where the document layout components are identified as keys and the corresponding information from the figures or tables is extracted as values. Following this, the key-value pairs are sent to both the image search engine and the text search engine within the system. Pairs with the highest probability are then sent to the LLM-based question engine to generate relevant answers.

FIGURE 5.8: A sample illustrating the knowledge acquired by the IR module

Figure 5.8 illustrates a table extracted from the document parsing module. The blue-colored area represents the bounding box for the table, while the green-colored area indicates the recognized cell content within the table. For example, if a user asks, *"What is the amount for Treasury Bond Futures (3 years) in Foreign investment in domestic securities in November 2020?"*, then, the model can provide the numerical data extracted from the corresponding cell, along with a reference to the page and table.

5.3.4 Summarization

This research leverages the LLM-powered generation model, specifically OpenAI's GPT-3.5-turbo, to generate summarized answers. These answers include the retrieved information, such as the document name, page number, document layout entity, and its corresponding content. In addition, a threshold to enhance relevance is established. If the output probability provided by the text search engine falls below the threshold, this study infers that the available relevant data within the internal repository is insufficient. Consequently, this makes the system send the user query to the LLM's API to generate general answers. This plays a crucial role in exception handling. For instance, if a user asks a question like *"What's the weather like today?"* which is not relevant to the internal financial document repository, the system retrieves the answer from the general LLM's API instead of returning an error message.

5.4 Implementation

5.4.1 Data Pre-processing

This research utilizes open-access financial reports sourced from the Bank of Korea, including documents such as Economic Review, Monetary and Credit Policy Report, Minutes, Annual Report, and Local Information. Each report presents diverse layout structures, including two-column layouts or one-column layouts without specific formatting. In other words, all reports are semi-structured or unstructured documents, posing significant challenges for document layout analysis due to the absence of a comparable benchmark dataset (Fig 5.9). Additionally, these reports predominantly contain Korean content, with occasional inclusion of English and Chinese words. Both Korean and English text is retained, including any alphanumerical content, during the tokenization process.

Moreover, the file format is HWP (Hangul Word Processor), primarily used for saving documents in the Korean language, Hangul. The HWP file format presents another substantial challenge, primarily due to the inherent difficulty in extracting content and the limited availability of open-source libraries for handling HWP file formats. HWP files were converted to PDF format, and a single image file was extracted for each page of the multi-page PDF documents. Using the open-source Python library PDFminer, each text line within the documents were extracted. Furthermore, to extract content with a hierarchical format, HWP files were converted to XML, resulting in a content dictionary with a hierarchical structure. Metadata was then extracted from this XML, and the XML content was parsed. This information was then utilized for semantic entity recognition.

5.4.2 Data Annotation

To identify the document layout in unstructured and unlabelled documents, this research creates a subset of documents for human annotation. This involves a comprehensive review of various document layouts, leading to generating detailed tags that facilitate the handling of logical relations between document entities, instead of using the conventional layout

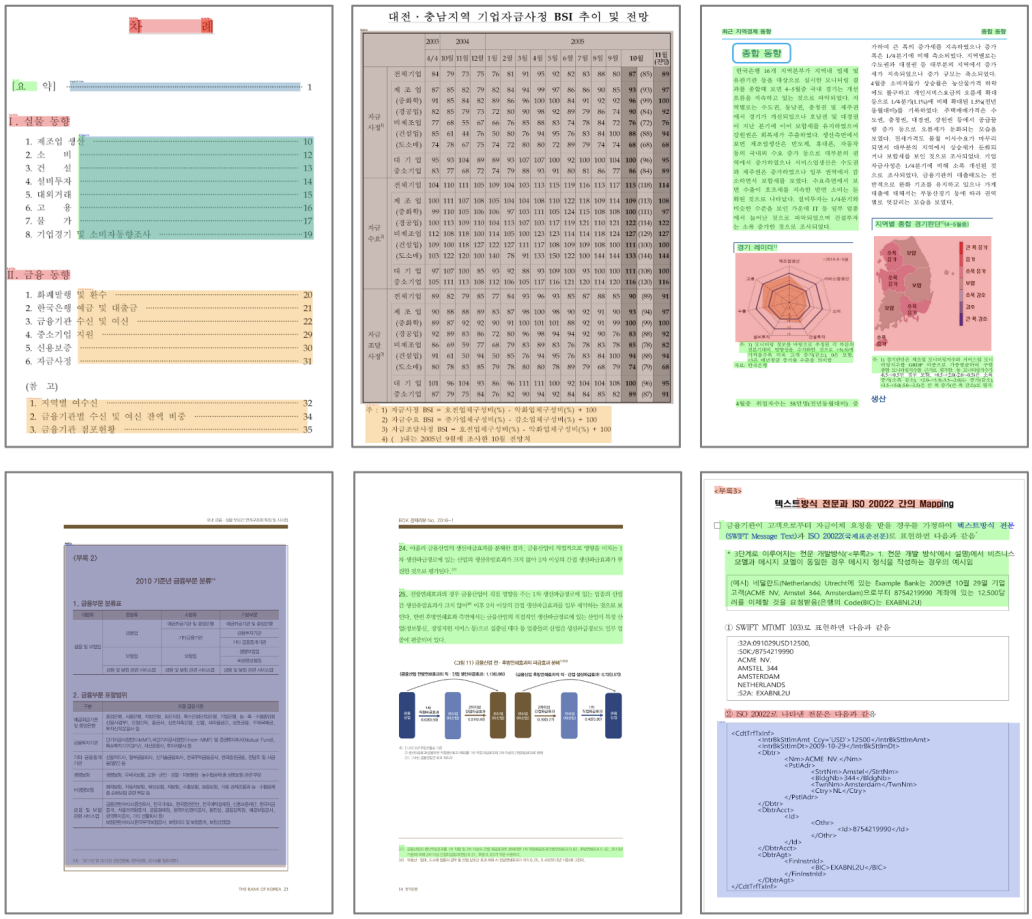


FIGURE 5.9: Samples of unstructured and unlabelled real-world documents. Initial results indicate low document layout analysis performance without additional processing.

components. As shown in figure 5.10, hierarchical text information is labeled including text level 1, text level 2, text level 3, subsection, and subsubsection. Additionally, the Minutes document does not have a specific format, requiring thorough annotation for hierarchically structured data (Fig 5.11)

5.4.3 Query Collection

This research selects question topics that include monetary and credit policy, economic outlook, financial stability, payment and settlement, and overseas economic trends. Using sample paragraphs related to these topics, this research utilizes LLMs to generate query-style

FIGURE 5.10: Human annotation guidelines. The document layout component entities are customized instead of following to conventional components.

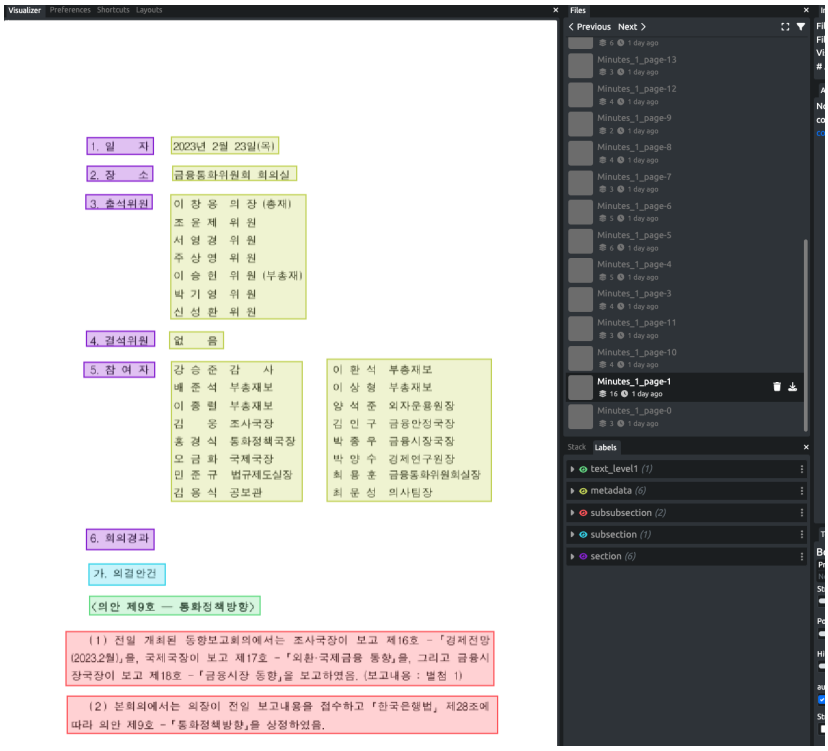


FIGURE 5.11: An example of data annotation process for unstructured document

questions. The prompt employed was *"Generate 3 query-style questions based on the above content; the answer should be directly extracted from the above paragraph. The question might be asked by end-users Please follow the format below Question (Korean version): Question (English): Answer (Korean version): Answer (English):"*. In the given example,

one of the generated questions is *"How did the direction of monetary policy change in South Korea after the foreign exchange crisis?"*. I reviewed the questions and answers generated by LLMs and generated the question template by selecting some high-quality question samples.

In the question template, this research curates the 75 most relevant questions associated with the internal document repository. The questions are utilized for intent classification and slot filling to generate query embeddings, employing the joint Ko-BigBird-BERT model. This embedding is utilized to search for the most relevant documents through the RAG process, described in 5.3.1 Document Searching section.

For quality control, this work selects 12 sample questions and generates ground truth answers corresponding to the document page and document layout components. The ground truth answers are generated by financial experts. The QA sample pairs selected for quality control are then used for evaluation, assessing the model's ability to correctly identify references and generate coherent and accurate answers.

5.4.4 Web Implementation

In the backend, five models are implemented: 1) slot and intent detection model, 2) document search top-k algorithms, 3) document layout analysis model, 4) key information extraction model, and 5) LLM-based generation model with RAG. Each of these models is detailed in the Methodology section. All the models are deployed on an internal server equipped with an NVIDIA A100 80GB GPU.

For the frontend implementation, this research adopts an open-source chatbot UI designed for the prototype desktop version. The backend and frontend systems were integrated by establishing internal APIs. These APIs facilitate the transmission of user queries to the backend models and subsequently convey the generated answers to the frontend web applications. The answer is displayed along with the document name, page number, and a link to the document repository, as shown in Fig 5.12.

FIGURE 5.12: FinDoc QA System Demonstration

5.5 Evaluation

When developing a QA system using real-world data, the documents lack labels and reference answers. Thus, a human evaluation is conducted related to the 12 sample QA pairs for which financial experts generated the ground truth. Note that the sample size for evaluation is small, and further research will require a more extensive quantitative evaluation.

This research follows the qualitative metric from Automated Evaluation of RAG, focusing on three aspects 1) Faithfulness, 2) Answer Relevance, and 3) Context Relevance. **Faithfulness** is a metric assessing whether the generated answer aligns factually with grounded sources and if the retrieved context can serve as justification for the generated answer. **Answer Relevance** is a metric assessing whether the generated answer directly addresses the question without incomplete or redundant information. **Context Relevance** is a metric evaluating whether the

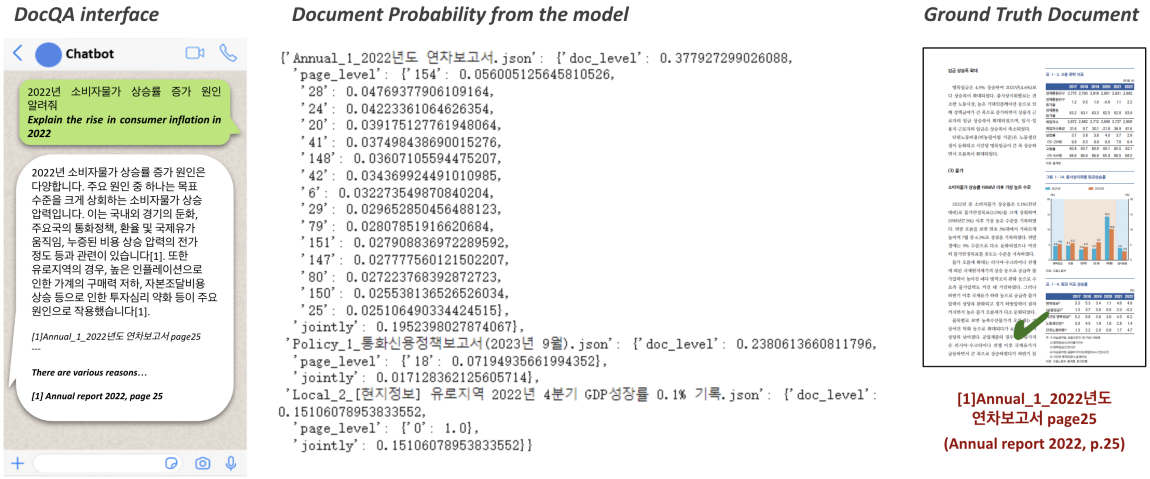


FIGURE 5.13: An Example of Evaluation. It involves reviewing the answer in the web interface and comparing the model's probability with the ground truth document. The demo system generates answers in Korean.

retrieved context contains the necessary information to answer the question while minimizing the inclusion of irrelevant information.

In the evaluation stage, a financial expert who was not involved in generating the sample QA pairs reviewed the generated answers by comparing the ground truth with the model output for the retrieved context. For Faithfulness, 83% of the samples have the ground truth pages ranked within the top 2 probability document pages. Fig 5.13 illustrates that the question is *"Explain the rise in consumer inflation in 2022"* and the model generates answers by retrieving the correct page that aligns with the ground truth. The model output includes the probability at both the document level and page level, providing insights into how the retrieved context is ranked higher than others.

For Answer Relevance, 75% of the samples directly address the question. Some questions, such as *"Explain domestic economic trends"*, ask for broad perspectives, causing the model to fail in extracting the ground truth pages. While the generated answers may sound coherent and make sense when read independently, they often contain irrelevant information and do not directly answer the question. Furthermore, some answers include a warning message, such as *"This rate can change over time, so please check the recent data"*, which is a thoughtful addition, but it may seem redundant.

For Context Relevance, 83% of the samples contain the necessary information from the retrieved context. In examining the failure cases, the question asking *"Explain the impact of the rise in central bank interest rates on the economy in 2023"* retrieved the ground truth page but from a different paragraph. The question contains cause-and-effect relations, yet the generated answer shows that a rise in consumer loans leads to an increase in central bank interest rates, indicating an opposite relationship. Since the answer incorporates all common key topics, it may appear to make sense without a detailed examination.

5.6 Conclusion

This chapter proposes FinDoc, a multimodal financial document understanding and Document QA system designed to handle unstructured and unlabelled internal financial documents. This complex system integrates several distinct deep learning models to address a variety of tasks, such as Intent Classification and Slot Filling, Layout Analysis, Key Information Extraction, and Retrieval Augmented Generation (RAG) using LLMs. The system architecture is built on four seamless modules: Document Searching, Document Parsing, Information Retrieval, and Summarization. To process real-world documents with multiple pages, Document Searching and Document Parsing necessitate various document pre-processing techniques. These include generating metadata through XML conversion from PDF, augmenting user queries containing financial terms through NER and POS tagging, and enhancing document layout analysis using few-shot learning on pre-trained DocAI models. This research finds that the QA algorithms implemented in the combination of the Information Retrieval and Summarization modules work quite effectively. Furthermore, this research demonstrates the system in a web environment, offering a replicable financial application in Intelligent Document Processing. For future research, this research plans to develop a new pre-trained DocAI model capable of handling text, layout, and visual features, catering to languages beyond English.

CHAPTER 6

Discussion and Conclusion

This study pioneered the research direction of employing deep learning-based feature extraction and information fusion for finance NLP applications. To demonstrate the capabilities of NLP in addressing real-world cases, this research presents three financial NLP segments. The detailed research process includes real-world data collection, the implementation of deep learning NLP models across various tasks, extensive experiments covering both quantitative and qualitative analyses, and the demonstration of end-to-end NLP application systems.

In Chapter 3, the FedNLP system is introduced as the first interpretable multi-component NLP system designed for decoding Federal Reserve communications. This pilot system presents the application of NLP in analyzing various forms of financial documents and enables end-users to get holistic insights. In Chapter 4, the StockEmotions dataset is introduced, representing a novel dataset for emotion classification in the stock market that goes beyond conventional financial sentiment classification. The creation of this dataset involves a multi-step annotation pipeline, fostering collaboration between humans and pre-trained language models. In Chapter 5, the FinDoc system is introduced as a multimodal financial Document Understanding and Document QA system, specifically designed to handle unstructured and unlabelled internal financial documents. This complex system combines various deep learning models to address a range of tasks, including Intent Classification and Slot Filling, Layout Analysis, Key Information Extraction, and RAG with LLMs. Overall, this research demonstrates the practical capabilities of financial NLP applications by addressing various aspects, including real-world data, multiple tasks, and showcasing demo systems.

Moreover, this research provides the first comprehensive review of LLMs in finance. It explores the evolution from general-domain LMs to financial-domain LMs, compares five

techniques across financial LMs, and summarizes 14 downstream financial NLP tasks along with corresponding 33 datasets for further research. To support AI research in finance, this study compiles a collection of accessible datasets and evaluation benchmarks. I hope that this research can provide valuable insights into the utilization of NLP applications in finance, expanding perspectives on the potential opportunities presented by LLMs in the financial domain.

6.1 Discussion on GenAI in Financial NLP

As reviewed in Chapter 2, while research on LLMs has advanced rapidly, the exploration of FinLLMs remains in its early stage. This section discusses various aspects guiding the future directions of Financial NLP, primarily from the perspective of Generative AI (GenAI) in industry.

GenAI emerged prominently, showcasing positive outcomes and introducing new potential risks. According to the Google Cloud Generative AI Benchmarking Study [1], 82% of organizations considering or currently utilizing GenAI believe it will either significantly change or transform their respective industries. In terms of economic impact, the McKinsey Global Institute [2] estimates that GenAI could contribute the equivalent of USD 2.6 trillion to USD 4.4 trillion annually in value worldwide. Among industry sectors, banking is anticipated to have one of the most significant opportunities, with an annual potential of USD 200 billion to USD340 billion, primarily driven by increased productivity.

For the financial NLP research, this suggests significant potential for exploration into the impact and applications of GenAI, particularly within the banking and finance sector. Through this research, I examined various NLP applications in finance, encompassing processes from data collection to end-to-end applications. Multiple NLP techniques and downstream tasks contribute to the development of real-world applications. To effectively implement GenAI, it

[1] https://cloud.google.com/blog/topics/financial-services/five-generative-ai-use-cases-financial-services-industry

[2] https://www.mckinsey.com/industries/financial-services/our-insights/capturing-the-full-value-of-generative-ai-in-banking

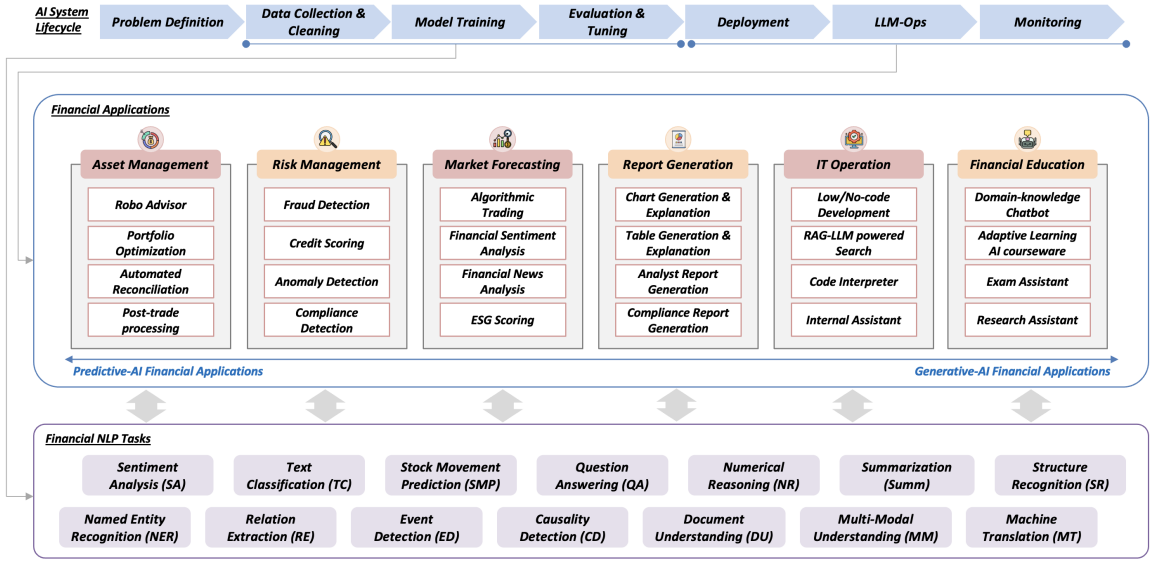


FIGURE 6.1: Financial Applications associated with AI Lifecycle and Financial NLP Tasks.

is crucial to review which financial NLP applications are applicable and how they align with financial NLP tasks.

Fig 6.1 provides an overview of financial applications associated with Financial NLP tasks and the AI system lifecycle. Towards the left, more predictive AI applications are illustrated, including categories, such as asset management, risk management, and market forecasting. Towards the right, more generative AI applications are illustrated, including report generation, IT operation, and financial education. These segments are presented for illustrative purposes, highlighting that traditional applications are also significantly influenced by GenAI. For instance, chatbot services have undergone a substantial transformation from rule-based NLP to GenAI. To explore the potential and recently developed real-world financial applications, this section presents three recent real-world cases and proposes three areas where GenAI can be applied in finance.

ING Bank uses GenAI chatbot to better assist customers: [3] In September 2023, ING Bank launched a GenAI chatbot as a customer-facing pilot in Europe. Collaborating with the McKinsey QuantumBlack team, the service was built and deployed in seven weeks. This

[3] https://www.mckinsey.com/industries/financial-services/how-we-help-clients/banking-on-innovation-how-ing-uses-generative-ai-to-put-people-first

chatbot offered customers immediate, tailored assistance while maintaining clear safeguards to mitigate risk. The pilot study established a blueprint for scaling across 10 markets, with the potential to impact over 37 million customers in 40 countries. This significantly outpaces previous industry-standard chatbots, which can require several years of programming and becoming operational.

Moody's Research Assistant, GenAI-powered research tool: [4] In December 2023, Moody's Corporation launched Moody's Research Assistant, a search and analytical tool powered by LLMs and Moody's extensive proprietary data. This research tool is commercially available for financial market participants and synthesizes vast amounts of information on Moody's credit research, so users can quickly assess lending or investment opportunities, monitor developments, compare entities, and enhance analytical workflows at scale. The research tool significantly reduces the time it takes to conduct in-depth risk analysis, enabling users to generate more holistic insights faster.

Bloomberg's GenAI tool on Mobile Apps: [5] In July 2024, Bloomberg announced the availability of AI-powered Earnings Call Summaries and the Document Search function on its mobile apps. Bloomberg's GenAI tool has been available for its terminal users since January 2024, providing summaries and analyses of company performances, highlighting essential points for companies in the Russell 1000 and Europe's top 1000. It offers summaries enriched with financial language nuances and investor-relevant insights, including factors like company guidance, capital allocation, and macro-environmental impacts. Bloomberg's solutions are expected to streamline the process of discovering the most salient points in earnings call transcripts.

6.1.1 Intelligent Financial Information Retrieval

Banks invest a substantial amount of time in internal activities related to document search and summarization. Consequently, this diminishes the time available for client interactions.

[4] https://www.moodys.com/web/en/us/creditview/blog/introducing-research-assistant.html

[5] https://www.bloomberg.com/company/press/bloomberg-releases-gen-ai-enhanced-solution-on-mobile-apps/

The implementation of GenAI holds the potential to assist bank employees in efficiently locating and comprehending information within various contracts, such as policies, credit memos, underwriting, trading, lending, claims, and regulatory documents. Moreover, GenAI can extend its utility to address unstructured PDF documents, offering capabilities such as summarizing regulatory filings for a specific bank. For example, it can accelerate report generation by summarizing vast economic data and statistics from a global context. Additionally, in the context of corporate banking, GenAI contributes to preparing for customer meetings by generating comprehensive and intuitive pitch books and presentation materials, fostering engaging conversations.

6.1.2 GenAI-Enhanced Chatbot

At times, customers encounter challenges in seeking solutions to specific, unique problems that are not pre-programmed within existing AI chatbots or accessible through the knowledge repositories available to customer support agents. This is where GenAI proves beneficial in facilitating access to pertinent information for customers. Its proficiency lies in navigating extensive datasets, extracting and summarizing relevant details, and aiding customer agents or complementing existing AI chatbots. Furthermore, GenAI-powered chatbots can enhance conversational capabilities, providing a more tailored and satisfying customer experience.

6.1.3 AI-Driven Macroeconomic research

To comprehensively understand global markets and assess risks, investment firms need to analyze a wide range of company filings, transcripts (e.g., earnings calls, monetary policy), consensus estimates, macroeconomic reports, regulatory filings, and complex multimodal data. GenAI tools can act as invaluable research assistants for investment analysts in macroeconomic market research, helping them navigate vast datasets, identify key information, and generate concise summaries quickly and intelligently.

6.2 Opportunities and Challenges

This section explores the potential and limitations of financial NLP research, particularly in the context of leveraging GenAI and LLMs.

6.2.1 Datasets

High-quality and multimodal data are significantly important for developing sophisticated language models in any domain. As most FinLLMs are trained using general-domain LLMs on financial-specific data, the challenge lies in collecting high-quality financial data in diverse formats. Building instruction-finetuned financial datasets by converting existing datasets for specific financial NLP tasks will contribute to building advanced FinLLMs. In addition, aligning with the general trends in AI, financial NLP research is increasingly focused on handling multimodal data, including text, tabular, audio and video. Hence, the research on the capabilities of LLMs on financial multimodal datasets will become increasingly important, enhancing the performance of not only FinLLMs and conducting advanced financial SOTA models on complex tasks.

6.2.2 Techniques

One major challenge in financial NLP research is utilizing internal data without compromising privacy, causing security issues, or undermining trust in the responses generated by LLMs. To address this challenge, promising techniques for LLMs, such as Retrieval Augmented Generation (RAG), are being implemented in the financial domain. The RAG system is similar to an open-book approach, allowing the model to retrieve information without memorizing it. Particularly, implementing RAG in a Financial QA system has several advantages. It provides the model with access to reliable facts, enabling the generation of cross-referenced answers, therefore improving reliability, and mitigating hallucination issues [73]. Moreover, RAG enables the use of internal non-trainable data without retraining the entire model, ensuring privacy concerns are not breached.

6.2.3 Evaluation

Current evaluations of FinLLMs have focused on less complex financial NLP tasks. The challenge in the evaluation of financial NLP is leveraging the expertise of financial professionals to validate model performance on advanced tasks. The current evaluation results were presented using commonly used NLP metrics such as F1-score or Accuracy. However, knowledge-driven tasks require human evaluation by financial experts, appropriate financial evaluation metrics over NLP metrics [122], and expert feedback for model alignment. Furthermore, the advanced financial NLP tasks I presented, would discover the hidden capabilities of FinLLMs. These complex tasks will assess whether FinLLMs can serve as general financial problem-solver models [56], considering both cost and performance for specific tasks.

6.2.4 Implementation

LLMs serve as intermediate products that can be fine-tuned for specific tasks or used through various prompting techniques without weight updates. The challenge in selecting suitable models and techniques lies in the trade-off between cost and performance. Depending on the task complexity and inference cost, selecting general-domain LLMs with prompting or task-specific SOTA models might be a more practical choice than building FinLLMs. This requires LLMOps engineering skills, including soft prompt techniques and monitoring operation systems with a Continuous Integration (CI) and Continuous Delivery (CD) pipeline.

6.2.5 Applications

Unlike simpler applications, building real-world financial systems with LLMs requires the seamless integration of multiple complex NLP tasks. For example, developing a financial document understanding system requires LLMs with multimodal capabilities to handle tasks, including layout understanding, key information extraction, and question answering. Practitioners must focus on advanced NLP tasks and pre-integrated solutions. By addressing these components effectively and combining them seamlessly, we can build more robust end-to-end financial systems. Furthermore, sharing GenAI or FinLLM use cases across various

financial fields will be beneficial, particularly in understanding how generative models change existing financial services. For example, the combination of complex tasks such as structure recognition, question answering, and numerical reasoning tasks can provide advantages in assisting automatic report generation or analyzing financial PDF documents.

6.3 Future Work

An extension of this research will focus on exploring the multimodal capability and leveraging LLMs in three key aspects—Dataset, Model, and Application, which are identified as potential areas for future work.

Firstly, the research will proceed with the construction of multimodal datasets and the implementation of LLM-based annotation. In the FedNLP application, challenges arose in handling long-length documents. Leveraging techniques from Document Understanding, this research will extend to collect various types of documents from the Federal Reserve and generate a financial document multimodal dataset that includes figures, tables, and text. Some documents encompass economic data calculations, contributing to document understanding and numerical reasoning tasks. Additionally, in the StockEmotion application, the research utilized annotation leveraging PLM capabilities. Considering the impressive results demonstrated by LLMs, there have been attempts towards LLM-based data annotation. This research will extend to review LLM-based annotation techniques and further collect the recent data and make it on a larger scale.

Secondly, the research will focus on the development of new models for document understanding and stock market prediction tasks. In the FinDoc application, the research implemented a model using a Transformer-based pre-trained docAI model for document layout analysis and key information extraction. For future research, a new pre-trained model will be explored, aiming to demonstrate joint learning capabilities from text, layout, and visual features with low computation costs. Moreover, in the StockEmotion application, the research will explore the integration of various deep learning models with Knowledge Graph-based models for stock market prediction tasks. This task involves handling diverse multimodalities, including text,

emotion, emojis, and time series data. The initial research demonstrated the advantages of leveraging different modalities. This will be expanded to explore specific details that include implementing joint learning at different parts of the network, determining optimal embedding sizes, assessing the impact of concatenation before or after the network, experimenting with window sizes, and conducting back-testing experiments. Additionally, more baseline models will be researched to demonstrate performance variations when using text-only, text-price combination, and text-emotion-price together.

Last but not least, the research will enhance the demo applications to improve the visualization of model outputs, aiming to facilitate interpretation. In the FinDoc application, the QA system demonstrated the advantage of RAG with LLM for handling internal financial data by generating links to the relevant pages in the answer. The research will extend this functionality to display tables or charts, aiding end-users in improving trust in the model. Furthermore, the research will enhance system functionality, allowing users to choose different pre-training models through the user interface, enabling them to observe diverse outcomes.

Bibliography

[1] Muhammad Abdul-Mageed and Lyle Ungar. 'Emonet: Fine-grained emotion detection with gated recurrent neural networks'. In: *Proceedings of the 55th annual meeting of the association for computational linguistics (volume 1: Long papers)*. 2017, pp. 718–728.

[2] Julio Cesar Salinas Alvarado, Karin Verspoor and Timothy Baldwin. 'Domain adaption of named entity recognition to support credit risk assessment'. In: *Proceedings of the Australasian Language Technology Association Workshop 2015*. 2015, pp. 84–90.

[3] Nurudin Alvarez-Gonzalez, Andreas Kaltenbrunner and Vicenç Gómez. 'Uncovering the Limits of Text-based Emotion Detection'. In: *Findings of the Association for Computational Linguistics: EMNLP 2021*. 2021, pp. 2560–2583.

[4] Saleema Amershi et al. 'Modeltracker: Redesigning performance analysis tools for machine learning'. In: *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems*. New York, NY, USA: ACM, 2015, pp. 337–346.

[5] Srikar Appalaraju et al. 'Docformer: End-to-end transformer for document understanding'. In: *Proceedings of the IEEE/CVF international conference on computer vision*. 2021, pp. 993–1003.

[6] Dogu Araci. 'Finbert: Financial sentiment analysis with pre-trained language models'. In: *arXiv preprint arXiv:1908.10063* (2019).

[7] Youngmin Baek et al. 'Character region awareness for text detection'. In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 2019, pp. 9365–9374.

[8] Ben S Bernanke and Kenneth N Kuttner. 'What explains the stock market's reaction to Federal Reserve policy?' In: *The Journal of finance* 60.3 (2005), pp. 1221–1257.

[9] Alan S Blinder et al. ‘Central bank communication and monetary policy: A survey of theory and evidence’. In: *Journal of economic literature* 46.4 (2008), pp. 910–45.

[10] Johan Bollen, Huina Mao and Xiaojun Zeng. ‘Twitter mood predicts the stock market’. In: *Journal of computational science* 2.1 (2011), pp. 1–8.

[11] Rishi Bommasani et al. ‘On the opportunities and risks of foundation models’. In: *arXiv preprint arXiv:2108.07258* (2021).

[12] Tom Brown et al. ‘Language models are few-shot learners’. In: *Advances in neural information processing systems* 33 (2020), pp. 1877–1901.

[13] Sébastien Bubeck et al. ‘Sparks of artificial general intelligence: Early experiments with gpt-4’. In: *arXiv preprint arXiv:2303.12712* (2023).

[14] Erik Cambria et al. ‘SenticNet 6: Ensemble application of symbolic and subsymbolic AI for sentiment analysis’. In: *Proceedings of the 29th ACM international conference on information & knowledge management*. 2020, pp. 105–114.

[15] Iñigo Casanueva et al. ‘Efficient Intent Detection with Dual Sentence Encoders’. In: *Proceedings of the 2nd Workshop on Natural Language Processing for Conversational AI*. 2020, pp. 38–45.

[16] Tommaso Caselli et al. ‘DALC: the Dutch Abusive Language Corpus’. In: *Proceedings of the 5th Workshop on Online Abuse and Harms (WOAH 2021)*. 2021, pp. 54–66.

[17] Yupeng Chang et al. ‘A survey on evaluation of large language models’. In: *arXiv preprint arXiv:2307.03109* (2023).

[18] Gary Charness, Uri Gneezy and Michael A Kuhn. ‘Experimental methods: Between-subject and within-subject design’. In: *Journal of Economic Behavior & Organization* 81.1 (2012), pp. 1–8.

[19] Ankush Chatterjee et al. ‘SemEval-2019 task 3: EmoContext contextual emotion detection in text’. In: *Proceedings of the 13th international workshop on semantic evaluation*. 2019, pp. 39–48.

[20] Chung-Chi Chen, Hen-Hsen Huang and Hsin-Hsi Chen. ‘Issues and perspectives from 10,000 annotated financial social media data’. In: *Proceedings of The 12th language resources and evaluation conference*. 2020, pp. 6106–6110.

[21] Mark Chen et al. ‘Evaluating large language models trained on code’. In: *arXiv preprint arXiv:2107.03374* (2021).

[22] Tianqi Chen and Carlos Guestrin. ‘Xgboost: A scalable tree boosting system’. In: *Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining*. New York, NY, USA: ACM, 2016, pp. 785–794.

[23] Zhiyu Chen et al. ‘ConvFinQA: Exploring the Chain of Numerical Reasoning in Conversational Finance Question Answering’. In: *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*. 2022, pp. 6279–6292.

[24] Zhiyu Chen et al. ‘FinQA: A Dataset of Numerical Reasoning over Financial Data’. In: *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*. 2021, pp. 3697–3711.

[25] Kyunghyun Cho et al. ‘Learning Phrase Representations using RNN Encoder-Decoder for Statistical Machine Translation’. In: *EMNLP*. 2014.

[26] Aakanksha Chowdhery et al. ‘Palm: Scaling language modeling with pathways’. In: *Journal of Machine Learning Research* 24.240 (2023), pp. 1–113.

[27] Hyung Won Chung et al. ‘Scaling instruction-finetuned language models’. In: *arXiv preprint arXiv:2210.11416* (2022).

[28] Yi-Ling Chung et al. ‘CONAN-COunter NArratives through Nichesourcing: a Multilingual Dataset of Responses to Fight Online Hate Speech’. In: *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*. 2019, pp. 2819–2829.

[29] Kevin Clark et al. ‘Electra: Pre-training text encoders as discriminators rather than generators’. In: *arXiv preprint arXiv:2003.10555* (2020).

[30] Jacob Cohen. ‘A coefficient of agreement for nominal scales’. In: *Educational and psychological measurement* 20.1 (1960), pp. 37–46.

[31] Keith Cortis et al. ‘Semeval-2017 task 5: Fine-grained sentiment analysis on financial microblogs and news’. In: *Proceedings of the 11th international workshop on semantic evaluation (SemEval-2017)*. 2017, pp. 519–535.

[32] Kimberlé Crenshaw. ‘Demarginalizing the intersection of race and sex: A black feminist critique of antidiscrimination doctrine, feminist theory and antiracist politics’. In: *u. Chi. Legal f.* (1989), p. 139.

[33] Thomas Davidson et al. ‘Automated hate speech detection and the problem of offensive language’. In: *Proceedings of the International AAAI Conference on Web and Social Media*. Vol. 11. 2017.

[34] Dorottya Demszky et al. ‘GoEmotions: A Dataset of Fine-Grained Emotions’. In: *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*. 2020, pp. 4040–4054.

[35] Yang Deng et al. ‘PACIFIC: Towards Proactive Conversational Question Answering over Tabular and Textual Data in Finance’. In: *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*. 2022, pp. 6970–6984.

[36] Jacob Devlin et al. ‘BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding’. In: *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*. Minneapolis, Minnesota: Association for Computational Linguistics, 2019, pp. 4171–4186.

[37] Xiao Ding et al. ‘Deep learning for event-driven stock prediction’. In: *Proceedings of the 24th International Conference on Artificial Intelligence*. 2015, pp. 2327–2333.

[38] Yihao Ding et al. ‘Form-NLU: Dataset for the Form Natural Language Understanding’. In: *Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval*. 2023, pp. 2807–2816.

[39] Yihao Ding et al. ‘V-Doc: Visual questions answers with Documents’. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2022, pp. 21492–21498.

[40] Xin Du and Kumiko Tanaka-Ishii. ‘Stock embeddings acquired from news articles and price history, and an application to portfolio optimization’. In: *Proceedings of the 58th annual meeting of the association for computational linguistics*. 2020, pp. 3353–3363.

[41] Darren Duxbury et al. ‘How emotions influence behavior in financial markets: a conceptual analysis and emotion-based account of buy-sell preferences’. In: *The European Journal of Finance* 26.14 (2020), pp. 1417–1438.

[42] Paul Ekman. ‘An argument for basic emotions’. In: *Cognition & emotion* 6.3-4 (1992), pp. 169–200.

[43] Mai ElSherief et al. ‘Hate lingo: A target-based linguistic analysis of hate speech in social media’. In: *Proceedings of the International AAAI Conference on Web and Social Media*. Vol. 12. 1. 2018.

[44] Bjarke Felbo et al. ‘Using millions of emoji occurrences to learn any-domain representations for detecting sentiment, emotion and sarcasm’. In: *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*. 2017, pp. 1615–1625.

[45] Paula Fortuna and Sérgio Nunes. ‘A survey on automatic detection of hate speech in text’. In: *ACM Computing Surveys (CSUR)* 51.4 (2018), pp. 1–30.

[46] Paula Fortuna et al. ‘A hierarchically-labeled portuguese hate speech dataset’. In: *Proceedings of the Third Workshop on Abusive Language Online*. 2019, pp. 94–104.

[47] Antigoni Founta et al. ‘Large scale crowdsourcing and characterization of twitter abusive behavior’. In: *Proceedings of the International AAAI Conference on Web and Social Media*. Vol. 12. 2018.

[48] Jianliang Gao et al. ‘Graph-based stock recommendation by time-aware relational attention network’. In: *ACM Transactions on Knowledge Discovery from Data (TKDD)* 16.1 (2021), pp. 1–21.

[49] Lei Gao and Ruihong Huang. ‘Detecting Online Hate Speech Using Context Aware Models’. In: *Proceedings of the International Conference Recent Advances in Natural Language Processing, RANLP*. 2017, pp. 260–266.

[50] Yidi Ge et al. ‘Beyond negative and positive: Exploring the effects of emotions in social media during the stock market crash’. In: *Information Processing & Management* 57.4 (2020), p. 102218.

[51] Daniela Gerz et al. ‘Multilingual and Cross-Lingual Intent Detection from Spoken Data’. In: *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*. 2021, pp. 7468–7475.

[52] Shantanu Godbole and Sunita Sarawagi. ‘Discriminative methods for multi-labeled classification’. In: *Pacific-Asia conference on knowledge discovery and data mining*. Springer. 2004, pp. 22–30.

[53] John Griffith, Mohammad Najand and Jiancheng Shen. ‘Emotions in the stock market’. In: *Journal of Behavioral Finance* 21.1 (2020), pp. 42–56.

[54] Maarten Grootendorst. ‘BERTopic: Neural topic modeling with a class-based TF-IDF procedure’. In: *arXiv preprint arXiv:2203.05794* (2022).

[55] Yu Gu et al. ‘Domain-specific language model pretraining for biomedical natural language processing’. In: *ACM Transactions on Computing for Healthcare (HEALTH)* 3.1 (2021), pp. 1–23.

[56] Yue Guo, Zian Xu and Yi Yang. ‘Is ChatGPT a Financial Expert? Evaluating Language Models on Financial Natural Language Processing’. In: *The 2023 Conference on Empirical Methods in Natural Language Processing*. 2023.

[57] Mahmoud El-Haj. ‘Multiling 2019: Financial narrative summarisation’. In: *Proceedings of the Workshop MultiLing 2019: Summarization Across Languages, Genres and Sources*. 2019, pp. 6–10.

[58] Bernd Hayo and Matthias Neuenkirch. ‘Do Federal Reserve communications help predict federal funds target rate decisions?’ In: *Journal of Macroeconomics* 32.4 (2010), pp. 1014–1024.

[59] Thorsten Hens and Anna Meier. ‘Behavioral finance: the psychology of investing’. In: *Credit Suisse* (2015).

[60] Jordan Hoffmann et al. ‘Training compute-optimal large language models’. In: *arXiv preprint arXiv:2203.15556* (2022).

[61] Teakgyu Hong et al. ‘Bros: A pre-trained language model focusing on text and layout for better key information extraction from documents’. In: *Proceedings of the AAAI Conference on Artificial Intelligence*. Vol. 36. 10. 2022, pp. 10767–10775.

[62] Benjamin Hoover, Hendrik Strobelt and Sebastian Gehrmann. 'exBERT: A Visual Analysis Tool to Explore Learned Representations in Transformer Models'. In: *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics: System Demonstrations*. Association for Computational Linguistics, 2020, pp. 187–196.

[63] Chao-Chun Hsu et al. 'EmotionLines: An Emotion Corpus of Multi-Party Conversations'. In: *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*. 2018.

[64] Edward J Hu et al. 'LoRA: Low-Rank Adaptation of Large Language Models'. In: *International Conference on Learning Representations*. 2021.

[65] Jun Hu and Wendong Zheng. 'A deep learning model to effectively capture mutation information in multivariate time series prediction'. In: *Knowledge-Based Systems* 203 (2020), p. 106139.

[66] Ziniu Hu et al. 'Listening to chaotic whispers: A deep learning framework for news-oriented stock trend prediction'. In: *Proceedings of the eleventh ACM international conference on web search and data mining*. 2018, pp. 261–269.

[67] Yupan Huang et al. 'Layoutlmv3: Pre-training for document ai with unified text and image masking'. In: *Proceedings of the 30th ACM International Conference on Multimedia*. 2022, pp. 4083–4091.

[68] Muhammad Okky Ibrohim and Indra Budi. 'Multi-label hate speech and abusive language detection in Indonesian twitter'. In: *Proceedings of the Third Workshop on Abusive Language Online*. 2019, pp. 46–57.

[69] Guillaume Jaume, Hazim Kemal Ekenel and Jean-Philippe Thiran. 'Funsd: A dataset for form understanding in noisy scanned documents'. In: *2019 International Conference on Document Analysis and Recognition Workshops (ICDARW)*. Vol. 2. IEEE. 2019, pp. 1–6.

[70] Sarfaraz Javed et al. 'Impact of Federal Funds Rate on Monthly Stocks Return of United States of America'. In: *International Journal of Business and Management* 14.9 (2019), p. 105.

[71] Rasmus Jørgensen et al. ‘MultiFin: A Dataset for Multilingual Financial NLP’. In: *Findings of the Association for Computational Linguistics: EACL 2023*. 2023, pp. 864–879.

[72] Alexander Jung. ‘Have minutes helped to predict fed funds rate changes?’ In: *Journal of Macroeconomics* 49 (2016), pp. 18–32.

[73] Haoqiang Kang and Xiao-Yang Liu. ‘Deficiency of Large Language Models in Finance: An Empirical Examination of Hallucination’. In: *arXiv preprint arXiv:2311.15548* (2023).

[74] Myungkoo Kang et al. *Hate speech in Asia and Europe: Beyond hate and fear*. Routledge, 2020.

[75] Jared Kaplan et al. ‘Scaling laws for neural language models’. In: *arXiv preprint arXiv:2001.08361* (2020).

[76] Zixuan Ke et al. ‘Continual Pre-training of Language Models’. In: *The Eleventh International Conference on Learning Representations*. 2022.

[77] Yoon Kim. ‘Convolutional Neural Networks for Sentence Classification’. In: *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing, EMNLP*. ACL, 2014, pp. 1746–1751.

[78] Shimon Kogan et al. ‘Predicting risk from financial reports with regression’. In: *Proceedings of Human Language Technologies: The 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics*. The Association for Computational Linguistics, 2009, pp. 272–280.

[79] Josua Krause, Adam Perer and Kenney Ng. ‘Interacting with predictions: Visual inspection of black-box machine learning models’. In: *Proceedings of the 2016 CHI conference on human factors in computing systems*. ACM, 2016, pp. 5686–5697.

[80] Werner Kristjanpoller, Anton Fadic and Marcel C Minutolo. ‘Volatility forecast using hybrid neural network models’. In: *Expert Systems with Applications* 41.5 (2014), pp. 2437–2442.

[81] Ritesh Kumar et al. ‘Aggression-annotated Corpus of Hindi-English Code-mixed Data’. In: *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*. 2018.

[82] Quoc Le and Tomas Mikolov. 'Distributed representations of sentences and documents'. In: *International conference on machine learning*. Beijing, China: JMLR.org, 2014, pp. 1188–1196.

[83] Jean Lee et al. 'A Survey of Large Language Models in Finance (FinLLMs)'. In: *arXiv preprint arXiv:2402.02315* (2024).

[84] Jean Lee et al. 'Fednlp: an interpretable nlp system to decode federal reserve communications'. In: *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*. 2021, pp. 2560–2564.

[85] Jean Lee et al. 'K-MHaS: A Multi-label Hate Speech Detection Dataset in Korean Online News Comment'. In: *Proceedings of the 29th International Conference on Computational Linguistics*. 2022, pp. 3530–3538.

[86] Jean Lee et al. 'StockEmotions: Discover Investor Emotions for Financial Sentiment Analysis and Multivariate Time Series'. In: *arXiv preprint arXiv:2301.09279* (2023).

[87] Sangah Lee et al. 'KR-BERT: A Small-Scale Korean-Specific Language Model'. In: *ArXiv* abs/2008.03979 (2020).

[88] Brian Lester, Rami Al-Rfou and Noah Constant. 'The Power of Scale for Parameter-Efficient Prompt Tuning'. In: *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*. 2021, pp. 3045–3059.

[89] Patrick Lewis et al. 'Retrieval-augmented generation for knowledge-intensive nlp tasks'. In: *Advances in Neural Information Processing Systems* 33 (2020), pp. 9459–9474.

[90] Jiangnan Li et al. 'Past, present, and future: Conversational emotion recognition through structural modeling of psychological knowledge'. In: *Findings of the Association for Computational Linguistics: EMNLP 2021*. 2021, pp. 1204–1214.

[91] Jiazheng Li et al. 'Maec: A multimodal aligned earnings conference call dataset for financial risk prediction'. In: *Proceedings of the 29th ACM International Conference on Information & Knowledge Management*. 2020, pp. 3063–3070.

[92] Quanzhi Li and Sameena Shah. 'Learning stock market sentiment lexicon and sentiment-oriented word vector from stocktwits'. In: *Proceedings of the 21st Conference on Computational Natural Language Learning (CoNLL 2017)*. 2017, pp. 301–310.

[93] Xiang Lisa Li and Percy Liang. 'Prefix-Tuning: Optimizing Continuous Prompts for Generation'. In: *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*. 2021, pp. 4582–4597.

[94] Xianzhi Li et al. 'Are ChatGPT and GPT-4 General-Purpose Solvers for Financial Text Analytics? A Study on Several Typical Tasks'. In: *Proceedings of EMNLP: Industry Track*. 2023, pp. 408–422.

[95] Yinheng Li et al. 'Large Language Models in Finance: A Survey'. In: *Proceedings of the Fourth ACM International Conference on AI in Finance*. 2023, pp. 374–382.

[96] Vladislav Lialin, Vijeta Deshpande and Anna Rumshisky. 'Scaling down to scale up: A guide to parameter-efficient fine-tuning'. In: *arXiv preprint arXiv:2303.15647* (2023).

[97] Pengfei Liu et al. 'Pre-train, prompt, and predict: A systematic survey of prompting methods in natural language processing'. In: *ACM Computing Surveys* 55.9 (2023), pp. 1–35.

[98] Xiao Liu et al. 'P-Tuning: Prompt Tuning Can Be Comparable to Fine-tuning Across Scales and Tasks'. In: *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*. 2022, pp. 61–68.

[99] Yinhan Liu et al. 'Roberta: A robustly optimized bert pretraining approach'. In: *arXiv preprint arXiv:1907.11692* (2019).

[100] Zhuang Liu et al. 'Finbert: A pre-trained financial language representation model for financial text mining'. In: *Proceedings of the twenty-ninth international conference on international joint conferences on artificial intelligence*. 2021, pp. 4513–4519.

[101] Nikola Ljubešić, Tomaž Erjavec and Darja Fišer. 'Datasets of Slovene and Croatian moderated news comments'. In: *Proceedings of the 2nd Workshop on Abusive Language Online (ALW2)*. 2018, pp. 124–131.

[102] Shayne Longpre et al. 'The flan collection: Designing data and methods for effective instruction tuning'. In: *arXiv preprint arXiv:2301.13688* (2023).

[103] Steven Loria. *TextBlob: Simplified Text Processing*. 2017.

[104] Tim Loughran and Bill McDonald. 'When is a liability not a liability? Textual analysis, dictionaries, and 10-Ks'. In: *The Journal of finance* 66.1 (2011), pp. 35–65.

[105] Lefteris Loukas et al. 'FiNER: Financial Numeric Entity Recognition for XBRL Tagging'. In: *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. 2022, pp. 4419–4431.

[106] David O Lucca and Emanuel Moench. 'The pre-FOMC announcement drift'. In: *The Journal of Finance* 70.1 (2015), pp. 329–371.

[107] Scott M. Lundberg and Su-In Lee. 'A Unified Approach to Interpreting Model Predictions'. In: *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017*. Long Beach, CA, USA: Curran Associates, Inc., 2017, pp. 4765–4774.

[108] Macedo Maia et al. 'Www'18 open challenge: financial opinion mining and question answering'. In: *Companion proceedings of the the web conference 2018*. 2018, pp. 1941–1942.

[109] Pekka Malo et al. 'Good debt or bad debt: Detecting semantic orientations in economic texts'. In: *Journal of the Association for Information Science and Technology* 65.4 (2014), pp. 782–796.

[110] Dominique Mariko et al. 'The Financial Document Causality Detection Shared Task (FinCausal 2020)'. In: *Proceedings of the 1st Joint Workshop on Financial Narrative Processing and MultiLing Financial Summarisation*. 2020, pp. 23–32.

[111] Minesh Mathew, Dimosthenis Karatzas and CV Jawahar. 'Docvqa: A dataset for vqa on document images'. In: *Proceedings of the IEEE/CVF winter conference on applications of computer vision*. 2021, pp. 2200–2209.

[112] Puneet Mathur et al. 'Monopoly: Financial prediction from monetary policy conference videos using multimodal cues'. In: *Proceedings of the 30th ACM International Conference on Multimedia*. 2022, pp. 2276–2285.

[113] Rada Mihalcea and Paul Tarau. 'Textrank: Bringing order into text'. In: *Proceedings of the 2004 conference on empirical methods in natural language processing*. Barcelona, Spain: ACL, 2004, pp. 404–411.

[114] Tomas Mikolov et al. ‘Distributed representations of words and phrases and their compositionality’. In: *Proceedings of the 26th International Conference on Neural Information Processing Systems-Volume 2*. Lake Tahoe, Nevada, USA: Curran Associates, Inc., 2013, pp. 3111–3119.

[115] Saif Mohammad et al. ‘Semeval-2018 task 1: Affect in tweets’. In: *Proceedings of the 12th international workshop on semantic evaluation*. 2018, pp. 1–17.

[116] Jihyung Moon, Won Ik Cho and Junbum Lee. ‘BEEP! Korean Corpus of Online News Comments for Toxic Speech Detection’. In: *Proceedings of the Eighth International Workshop on Natural Language Processing for Social Media*. 2020, pp. 25–31.

[117] Hamdy Mubarak, Kareem Darwish and Walid Magdy. ‘Abusive language detection on Arabic social media’. In: *Proceedings of the first workshop on abusive language online*. 2017, pp. 52–56.

[118] Niklas Muennighoff et al. ‘Crosslingual generalization through multitask finetuning’. In: *arXiv preprint arXiv:2211.01786* (2022).

[119] Rajdeep Mukherjee et al. ‘ECTSum: A New Benchmark Dataset For Bullet Point Summarization of Long Earnings Call Transcripts’. In: *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*. 2022, pp. 10893–10906.

[120] Sobia Naseem et al. ‘The investor psychology and stock market behavior during the initial era of COVID-19: a study of China, Japan, and the United States’. In: *Frontiers in Psychology* 12 (2021), p. 16.

[121] Yixin Nie et al. ‘Adversarial NLI: A New Benchmark for Natural Language Understanding’. In: *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*. 2020, pp. 4885–4901.

[122] Hao Niu et al. ‘KeFVP: Knowledge-enhanced Financial Volatility Prediction’. In: *Findings of the Association for Computational Linguistics: EMNLP 2023*. 2023, pp. 11499–11513.

[123] Chikashi Nobata et al. ‘Abusive language detection in online user content’. In: *Proceedings of the 25th international conference on world wide web*. 2016, pp. 145–153.

[124] Nuno Oliveira, Paulo Cortez and Nelson Areal. 'Stock market sentiment lexicon acquisition using microblogging data and statistical measures'. In: *Decision Support Systems* 85 (2016), pp. 62–73.

[125] Nuno Oliveira, Paulo Cortez and Nelson Areal. 'The impact of microblogging data for stock market prediction: Using Twitter to predict returns, volatility, trading volume and survey sentiment indices'. In: *Expert Systems with Applications* 73 (2017), pp. 125–144.

[126] Nedjma Ousidhoum et al. 'Multilingual and Multi-Aspect Hate Speech Analysis'. In: *Proceedings of the 2019 Conference on EMNLP and the 9th IJCNLP (EMNLP-IJCNLP)*. 2019, pp. 4675–4684.

[127] Long Ouyang et al. 'Training language models to follow instructions with human feedback'. In: *Advances in Neural Information Processing Systems* 35 (2022), pp. 27730–27744.

[128] Jangwon Park. *KoELECTRA: Pretrained ELECTRA Model for Korean*. https://github.com/monologg/KoELECTRA. 2020.

[129] Seunghyun Park et al. 'CORD: a consolidated receipt dataset for post-OCR parsing'. In: *Workshop on Document Intelligence at NeurIPS 2019*. 2019.

[130] Sungjoon Park et al. 'Dimensional Emotion Detection from Categorical Emotion'. In: *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*. 2021, pp. 4367–4380.

[131] Jeffrey Pennington, Richard Socher and Christopher D Manning. 'Glove: Global vectors for word representation'. In: *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*. 2014, pp. 1532–1543.

[132] Matthew E. Peters et al. 'Deep Contextualized Word Representations'. In: *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2018*. New Orleans, Louisiana, USA: Association for Computational Linguistics, 2018, pp. 2227–2237.

[133] Robert Plutchik. 'A general psychoevolutionary theory of emotion'. In: *Theories of emotion*. Elsevier, 1980, pp. 3–33.

[134] Fabio Poletto et al. ‘Resources and benchmark corpora for hate speech detection: a systematic review’. In: *Language Resources and Evaluation* 55.2 (2021), pp. 477–523.

[135] Yu Qin and Yi Yang. ‘What You Say and How You Say It Matters: Predicting Stock Volatility Using Verbal and Vocal Cues’. In: *Proceedings of the 57th Conference of the Association for Computational Linguistics, ACL 2019*. Florence, Italy: Association for Computational Linguistics, 2019, pp. 390–401.

[136] Alec Radford et al. ‘Improving language understanding by generative pre-training’. In: (2018).

[137] Alec Radford et al. ‘Language models are unsupervised multitask learners’. In: *OpenAI blog* 1.8 (2019), p. 9.

[138] Colin Raffel et al. ‘Exploring the limits of transfer learning with a unified text-to-text transformer’. In: *Journal of machine learning research* 21.140 (2020), pp. 1–67.

[139] Radim Rehurek and Petr Sojka. ‘Software framework for topic modelling with large corpora’. In: *In Proceedings of the LREC 2010 workshop on new challenges for NLP frameworks*. Citeseer. 2010.

[140] Navid Rekabsaz et al. ‘Volatility Prediction using Financial Disclosures Sentiments with Word Embedding-based IR Models’. In: *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Vancouver, Canada: Association for Computational Linguistics, 2017, pp. 1712–1721.

[141] Marco Túlio Ribeiro, Sameer Singh and Carlos Guestrin. ‘"Why Should I Trust You?": Explaining the Predictions of Any Classifier’. In: *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. ACM, 2016, pp. 1135–1144.

[142] Carlo Rosa. ‘Words that shake traders: The stock market’s reaction to central bank communication in real time’. In: *Journal of Empirical Finance* 18.5 (2011), pp. 915–934.

[143] Victor Sanh, Thomas Wolf and Sebastian Ruder. ‘A Hierarchical Multi-Task Approach for Learning Embeddings from Semantic Tasks’. In: *The Thirty-Third AAAI Conference on Artificial Intelligence, AAAI 2019*. AAAI Press, 2019, pp. 6949–6956.

[144] Victor Sanh et al. 'DistilBERT, a distilled version of BERT: smaller, faster, cheaper and lighter'. In: *arXiv preprint arXiv:1910.01108* (2019).

[145] Ramit Sawhney et al. 'Cryptocurrency Bubble Detection: A New Stock Market Dataset, Financial Task & Hyperbolic Models'. In: *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. 2022, pp. 5531–5545.

[146] Ramit Sawhney et al. 'Deep attentive learning for stock movement prediction from social media text and company correlations'. In: *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. 2020, pp. 8415–8426.

[147] Teven Le Scao et al. 'Bloom: A 176b-parameter open-access multilingual language model'. In: *arXiv preprint arXiv:2211.05100* (2022).

[148] Evan A Schnidman and William D MacMillan. *How the Fed Moves Markets: Central Bank Analysis for the Modern Era*. Springer, 2016.

[149] Agam Shah, Suvan Paturi and Sudheer Chava. 'Trillion Dollar Words: A New Financial Dataset, Task & Market Analysis'. In: *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (ACL)*. 2023.

[150] Raj Shah et al. 'When FLUE Meets FLANG: Benchmarks and Large Pretrained Language Model for Financial Domain'. In: *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*. 2022, pp. 2322–2335.

[151] Soumya Sharma et al. 'Financial Numeric Extreme Labelling: A dataset and benchmarking'. In: *Findings of the Association for Computational Linguistics: ACL 2023*. 2023, pp. 3550–3561.

[152] Soumya Sharma et al. 'FinRED: A dataset for relation extraction in financial domain'. In: *Companion Proceedings of the Web Conference 2022*. 2022, pp. 595–597.

[153] Baoguang Shi, Xiang Bai and Cong Yao. 'An end-to-end trainable neural network for image-based sequence recognition and its application to scene text recognition'. In: *IEEE transactions on pattern analysis and machine intelligence* 39.11 (2016), pp. 2298–2304.

[154] Abu Awal Md Shoeb and Gerard de Melo. 'Emotag1200: Understanding the association between emojis and emotions'. In: *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. 2020, pp. 8957–8967.

[155] S. Siegel and N.J. Castellan. *Nonparametric Statistics for the Behavioral Sciences*. McGraw-Hill, 1988. ISBN: 9780070573574.

[156] Ankur Sinha and Tanmay Khandait. 'Impact of news on the commodity market: Dataset and results'. In: *Advances in Information and Communication: Proceedings of the 2021 Future of Information and Communication Conference (FICC), Volume 2*. Springer. 2021, pp. 589–601.

[157] SKTBrain. *KoBERT: Korean BERT pre-trained cased*. https://github.com/SKTBrain/KoBERT. 2019. (Visited on 12/01/2012).

[158] Mohammad S Sorower. 'A literature survey on algorithms for multi-label learning'. In: *Oregon State University, Corvallis* 18 (2010), pp. 1–25.

[159] Yejun Soun et al. 'Accurate Stock Movement Prediction with Self-supervised Learning from Sparse Noisy Tweets'. In: *2022 IEEE International Conference on Big Data (Big Data)*. IEEE. 2022, pp. 1691–1700.

[160] Hendrik Strobelt et al. 'Lstmvis: A tool for visual analysis of hidden state dynamics in recurrent neural networks'. In: *IEEE transactions on visualization and computer graphics* 24.1 (2017), pp. 667–676.

[161] Hendrik Strobelt et al. 'S eq 2s eq-v is: A visual debugging tool for sequence-to-sequence models'. In: *IEEE transactions on visualization and computer graphics* 25.1 (2018), pp. 353–363.

[162] Yu Sun et al. 'Ernie 2.0: A continual pre-training framework for language understanding'. In: *Proceedings of the AAAI conference on artificial intelligence*. Vol. 34. 05. 2020, pp. 8968–8975.

[163] Hao Tan and Mohit Bansal. 'LXMERT: Learning Cross-Modality Encoder Representations from Transformers'. In: *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*. 2019, pp. 5100–5111.

[164] Yixuan Tang et al. 'FinEntity: Entity-level Sentiment Classification for Financial Texts'. In: *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*. 2023, pp. 15465–15471.

[165] Ian Tenney et al. 'The Language Interpretability Tool: Extensible, Interactive Visualizations and Analysis for NLP Models'. In: *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*. 2020, pp. 107–118.

[166] Romal Thoppilan et al. 'Lamda: Language models for dialog applications'. In: *arXiv preprint arXiv:2201.08239* (2022).

[167] Hugo Touvron et al. 'Llama: Open and efficient foundation language models'. In: *arXiv preprint arXiv:2302.13971* (2023).

[168] Ashish Vaswani et al. 'Attention is all you need'. In: *Advances in neural information processing systems* 30 (2017).

[169] Bertie Vidgen and Leon Derczynski. 'Directions in abusive language training data, a systematic review: Garbage in, garbage out'. In: *Plos one* 15.12 (2020), e0243300.

[170] Jesse Vig and Yonatan Belinkov. 'Analyzing the Structure of Attention in a Transformer Language Model'. In: *Proceedings of the 2019 ACL Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*. ACL, 2019, pp. 63–76.

[171] Dongsheng Wang et al. 'DocLLM: A layout-aware generative language model for multimodal document understanding'. In: *arXiv preprint arXiv:2401.00908* (2023).

[172] Jiapeng Wang, Lianwen Jin and Kai Ding. 'LiLT: A Simple yet Effective Language-Independent Layout Transformer for Structured Document Understanding'. In: *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. 2022, pp. 7747–7757.

[173] Kunze Wang et al. 'Detect All Abuse! Toward Universal Abusive Language Detection Models'. In: *Proceedings of the 28th International Conference on Computational Linguistics*. 2020, pp. 6366–6376.

[174] Neng Wang, Hongyang Yang and Christina Wang. 'FinGPT: Instruction Tuning Benchmark for Open-Source Large Language Models in Financial Datasets'. In: *NeurIPS 2023 Workshop on Instruction Tuning and Instruction Following*. 2023.

[175] Sida I Wang and Christopher D Manning. 'Baselines and bigrams: Simple, good sentiment and topic classification'. In: *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*. 2012, pp. 90–94.

[176] William Yang Wang and Zhenhao Hua. 'A Semiparametric Gaussian Copula Regression Model for Predicting Financial Risks from Earnings Calls'. In: *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics, ACL 2014*. The Association for Computer Linguistics, 2014, pp. 1155–1165.

[177] Zeerak Waseem and Dirk Hovy. 'Hateful symbols or hateful people? Predictive features for hate speech detection on Twitter'. In: *Proceedings of the NAACL student research workshop*. 2016, pp. 88–93.

[178] Zeerak Waseem et al. 'Understanding Abuse: A Typology of Abusive Language Detection Subtasks'. In: *Proceedings of the First Workshop on Abusive Language Online*. 2017, pp. 78–84.

[179] Jason Wei et al. 'Chain-of-thought prompting elicits reasoning in large language models'. In: *Advances in Neural Information Processing Systems* 35 (2022), pp. 24824–24837.

[180] Jason Wei et al. 'Finetuned Language Models are Zero-Shot Learners'. In: *International Conference on Learning Representations*. 2021.

[181] Henry Weld et al. 'CONDA: a CONtextual Dual-Annotated dataset for in-game toxicity understanding and detection'. In: *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*. 2021, pp. 2406–2416.

[182] James Wexler et al. 'The what-if tool: Interactive probing of machine learning models'. In: *IEEE transactions on visualization and computer graphics* 26.1 (2019), pp. 56–65.

[183] Edmure Windsor and Wei Cao. 'Improving exchange rate forecasting via a new deep multimodal fusion model'. In: *Applied Intelligence* (2022), pp. 1–17.

[184] Thomas Wolf et al. 'Transformers: State-of-the-Art Natural Language Processing'. In: *Proceedings of the 2020 Conference on EMNLP: System Demonstrations*. Association for Computational Linguistics, Oct. 2020, pp. 38–45.

[185] Huizhe Wu et al. ‘Hybrid deep sequential modeling for social text-driven stock prediction’. In: *Proceedings of the 27th ACM international conference on information and knowledge management*. 2018, pp. 1627–1630.

[186] Shijie Wu et al. ‘Bloomberggpt: A large language model for finance’. In: *arXiv preprint arXiv:2303.17564* (2023).

[187] Ellery Wulczyn, Nithum Thain and Lucas Dixon. ‘Ex machina: Personal attacks seen at scale’. In: *Proceedings of the 26th international conference on world wide web*. 2017, pp. 1391–1399.

[188] Qianqian Xie et al. ‘PIXIU: A Comprehensive Benchmark, Instruction Dataset and Large Language Model for Finance’. In: *Advances in Neural Information Processing Systems* 36 (2024).

[189] Frank Xing et al. ‘Financial sentiment analysis: an investigation into common mistakes and silver bullets’. In: *Proceedings of the 28th international conference on computational linguistics*. 2020, pp. 978–987.

[190] Yang Xu et al. ‘LayoutLMv2: Multi-modal Pre-training for Visually-rich Document Understanding’. In: *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*. 2021, pp. 2579–2591.

[191] Yiheng Xu et al. ‘Layoutlm: Pre-training of text and layout for document image understanding’. In: *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. 2020, pp. 1192–1200.

[192] Yumo Xu and Shay B Cohen. ‘Stock movement prediction from tweets and historical prices’. In: *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. 2018, pp. 1970–1979.

[193] Hongyang Yang, Xiao-Yang Liu and Christina Dan Wang. ‘FinGPT: Open-Source Financial Large Language Models’. In: *arXiv preprint arXiv:2306.06031* (2023).

[194] Jingfeng Yang et al. ‘Harnessing the power of llms in practice: A survey on chatgpt and beyond’. In: *arXiv preprint arXiv:2304.13712* (2023).

[195] Linyi Yang et al. ‘Generating Plausible Counterfactual Explanations for Deep Transformers in Financial Text Classification’. In: *Proceedings of the 28th International Conference on Computational Linguistics*. 2020, pp. 6150–6160.

[196] Linyi Yang et al. ‘Html: Hierarchical transformer-based multi-task learning for volatility prediction’. In: *Proceedings of The Web Conference 2020*. Taipei, Taiwan: ACM / IW3C2, 2020, pp. 441–451.

[197] Yi Yang, Yixuan Tang and Kar Yan Tam. ‘InvestLM: A Large Language Model for Investment using Financial Domain Instruction Tuning’. In: *arXiv preprint arXiv:2309.13064* (2023).

[198] Yi Yang, Mark Christopher Siy Uy and Allen Huang. ‘Finbert: A pretrained language model for financial communications’. In: *arXiv preprint arXiv:2006.08097* (2020).

[199] Zhilin Yang et al. ‘Xlnet: Generalized autoregressive pretraining for language understanding’. In: *Advances in neural information processing systems* 32 (2019).

[200] Tomasz Zaleskiewicz and Jakub Traczyk. ‘Emotions and financial decision making’. In: *Psychological perspectives on financial decision making*. Springer, 2020, pp. 107–133.

[201] Marcos Zampieri et al. ‘Predicting the Type and Target of Offensive Posts in Social Media’. In: *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*. 2019, pp. 1415–1420.

[202] Marcos Zampieri et al. ‘SemEval-2020 Task 12: Multilingual Offensive Language Identification in Social Media (OffensEval 2020)’. In: *Proceedings of the Fourteenth Workshop on Semantic Evaluation*. 2020, pp. 1425–1447.

[203] Jiawei Zhang et al. ‘Manifold: A model-agnostic framework for interpretation and diagnosis of machine learning models’. In: *IEEE transactions on visualization and computer graphics* 25.1 (2018), pp. 364–373.

[204] Shengyu Zhang et al. ‘Instruction tuning for large language models: A survey’. In: *arXiv preprint arXiv:2308.10792* (2023).

[205] Xi Zhang et al. ‘Improving stock market prediction via heterogeneous information fusion’. In: *Knowledge-Based Systems* 143 (2018), pp. 236–247.

[206] Wayne Xin Zhao et al. ‘A survey of large language models’. In: *arXiv preprint arXiv:2303.18223* (2023).

[207] Xinyi Zheng et al. ‘Global table extractor (gte): A framework for joint table identification and cell structure recognition using visual context’. In: *Proceedings of the IEEE/CVF winter conference on applications of computer vision*. 2021, pp. 697–706.

[208] Dawei Zhou et al. ‘Domain Adaptive Multi-Modality Neural Attention Network for Financial Forecasting’. In: *WWW ’20: The Web Conference 2020*. ACM / IW3C2, 2020, pp. 2230–2240.

[209] Zhihan Zhou, Liqian Ma and Han Liu. ‘Trade the Event: Corporate Events Detection for News-Based Event-Driven Trading’. In: *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*. 2021, pp. 2114–2124.

[210] Fengbin Zhu et al. ‘TAT-QA: A Question Answering Benchmark on a Hybrid of Tabular and Textual Content in Finance’. In: *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*. 2021, pp. 3277–3287.

APPENDIX A

K-MHaS: A Korean Multi-label Hate Speech Detection Dataset

This appendix explores the extraction of emotional features from text and its application in the information fusion network. It also discusses the analysis of Korean language text. While this research is not focused on finance, it provides valuable insights into text analysis techniques that were applied to other studies (Chapter 4 and Chapter 5). This appendix is an extension of the work ***K-MHaS: A Multi-label Hate Speech Detection Dataset in Korean Online News Comment*** [85] accepted to the COLING 2022. I formulated the research aim, reviewed the previous work, checked the data quality, analyzed the data and the experiment results, and wrote most of the paper.

Online hate speech detection has become an important issue due to the growth of online content, but resources in languages other than English are extremely limited. We introduce K-MHaS[1], a new multi-label dataset for hate speech detection that effectively handles Korean language patterns. The dataset consists of 109k utterances from news comments and provides a multi-label classification using 1 to 4 labels, and handles subjectivity and intersectionality. We evaluate strong baselines on K-MHaS. KR-BERT with a sub-character tokenizer outperforms others, recognizing decomposed characters in each hate speech class.

A1 Introduction

The growth of online content including social media [202], news comments [49], Wikipedia [187], and in-game chat [181] presents challenges in detecting hate speech using advanced Natural Language Processing. Hate speech is a language that attacks or diminishes individuals

[1]The dataset is available at https://github.com/adlnlp/K-MHaS.

or groups based on certain characteristics such as physical appearance, religion, gender, or other attributes, and it can occur across different linguistic styles [45]. Hate speech detection is intrinsically a complex task [173] due to the fuzzy boundary with other overlapping concepts such as abusive language [123], toxic comments [187], or offensive language [33].

Recently, the rise in popularity of Korean TV, movies, and music (e.g. Squid Game, BTS) has led to many young people showing an interest in learning Korean. This phenomenon could result in exposure to harmful content and hate speech in Korean. However, (1) the most common language in hate speech research is English and only limited resources are available in other languages such as Arabic [117], Dutch [16], and Korean [116]. In addition, most datasets are annotated (2) using a single-label classification of particular aspects, even though the subjectivity of hate speech cannot be explained with a mutually exclusive annotation scheme.

We propose K-MHaS, a Korean multi-label hate speech detection dataset that allows overlapping labels associated with intersectionality, a concept from sociology that identifies combined attributes [32]. Our dataset consists of 109,692 utterances from Korean online news comments, labeled with 8 fine-grained hate speech classes. K-MHaS is compatible with previous work on hate speech in other languages, by providing binary classification and multi-label classification from 1(one) to 4(four) labels.

We investigate the K-MHaS dataset by analyzing label distribution, keywords, and label pairs. In addition, we provide strong baseline pre-trained language models using Multilingual-BERT, KoELECTRA, KoBERT, and KR-BERT, and compare the results using six metrics for multi-label classification tasks. Overall, the KoELECTRA model achieves the best performance for all labels, indicating the effects of the pre-training data source. The KR-BERT with a sub-character-level tokenizer outperforms the others on several label pairs, showing that decomposing various Korean characters is essential for the task. Our contribution can be summarized as follows:

- We propose a large-size Korean multi-label hate speech detection dataset that represents Korean language patterns effectively;

Publication	Language	Source	Data size	Labels	M-label
[177]	English	Twitter	16.2k	Sexism, Racism, Neither	N
[33]	English	Twitter	24.8k	Hate Speech, Offensive, Neither	N
[187]	English	Wikipedia comments	115k	Toxic, Severe Toxic, Obscene, Threat, Insult, Identity Hate, Neutral	Y
[68]	Indonesian	Twitter	11k	(a) Individual, Group (b) Religion, Race, Pysical, Gender, Other (c) Weak, Moderate, Strong Hate Speech	P
[46]	Portuguese	Twitter	5.6k	(a) Hate Speech, Not Hate Speech (b) Sexism, Body, Origin, Homophobia, Racism, Ideology, Religion, Health, Other-Lifestyle	P
[126]	English French Arabic	Twitter	6k (EN) 4k (FR) 3k (AR)	Labels for five different aspects (a) Directness, (b) Hostility, (c) Target, (d) Group, and (e) Annotator	P
[116]	Korean	News comments	9k	(a) Hate Speech, Offensive, None (b) Gender, Others, None	N
Ours	**Korean**	**News comments**	**109k**	**(a) Hate Speech, Not Hate Speech (b) Politics, Origin, Physical, Age, Gender Religion, Race, Profanity, Not Hate Speech**	**Y**

TABLE A.1: Comparison of datasets. A "M-label" indicates a multi-label annotation scheme that allows overlapping labels for intersectionality (P = partially applied). The (a) - (e) indicates a layer containing a single label from each aspect.

- We propose a multi-label hate speech annotation scheme, which can handle the subjectivity of hate speech and the intersectionality;
- We evaluate strong baseline experiments on our dataset using Korean-BERT-based language models with six different metrics.

A2 Related Work

A2.1 Hate Speech Terminology

Hate speech detection is an intrinsically complex task due to the fuzzy boundary of definitions and the subjectivity of content that often implies disagreement between the human judges [134]. A recent study addresses hate speech definitions in comparison to other overlapped concepts such as abusive[123], toxic comments [187], or offensive language [33]. However, defining hate speech lies in the inherent vagueness, depending on human subjective interpretation.

Abusive language is an umbrella term that includes hate speech and profanity [45]. Toxic comments are rude messages that are likely to make a person leave a discussion [187], whereas some hate speech can make people discuss more. Offensive language is intended to insult a targeted individual or group [47] and it can be differentiated from hate speech based on subtle linguistic distinctions [33]. However, defining hate speech lies in the inherent vagueness, depending on human subjective interpretation.

A2.2 Hate Speech Classifiers

Classifying hate speech has advanced from being a binary task to a fine-grained annotation schema in order to inspect users' motivation and behaviors. Several aspects in hate speech detection have been established including (a) **target** (e.g. direct, generalized)[43], (b) targeted **entity** (e.g. individual, group)[201] (c) **extent** (e.g. explicit, implicit)[173] (d) **hostility** (e.g. weak, moderate, strong)[68] (e) **category** (e.g. race, gender, religion, politics)[178] and (f) annotator's **emotion** (e.g. anger, disgust, confusion)[126].

Most datasets are annotated using a single-label classification of the particular aspects. Given that hate speech has intersections of similar concepts, this can be problematic. The subjectivity of hate speech cannot be explained in the "black and white" annotation scheme. Multi-level annotation is the most complex scheme [134], which refers to how much detail it contains. It is a mutually exclusive concept that involves several labels from several different aspects, however, it disregards the intersection of subtype aspects [126]. Inspired by intersectionality [32], a multi-label annotation scheme for classifying hate speech has emerged. Intersectionality is a concept for understanding how multiple aspects (e.g. social and political identities) combine to create different modes of hate speech. It brings attention to hate speech detection where linguistic features are subjected to multiple forms of hate within a society [46].

A2.3 Low Resources

The most common language in hate speech research is English. Over time, research has expanded into other low-resource languages including Hindi-English [81], Arabic [117],

Dutch [16], Slovene and Croatian [101]. For linguistic diversity, multilingual tasks [202, 28, 126] are conducted with a small-sized corpus. In terms of source of data, Twitter is the preferred online platform. Other sources such as news comments [49], Wikipedia [187], and gaming platforms [181] are rarely constructed. [116] propose a Korean hate speech dataset with around 10,000 online news comments, classifying two aspects of hate speech and gender bias.

A3 Korean Multi-label Hate Speech Detection Dataset (K-MHaS)

Our dataset is based on the Korean online news comments available on Kaggle [2] and Github [3]. The unlabeled raw data was collected between January 2018 and June 2020. In order to curate the data, we randomly select more than 109,692 news comments. Our data preprocessing is designed to tokenize a Korean character and filter the length. We remove URLs and bad characters (e.g. U+1100 to U+11FF - Hangul Jamo) using regular expressions while keeping uppercase and lowercase letters in English and emoji. We discard sentences with fewer than 10 characters as it is often only one word. For the data derived from online comments, we normalized repeated characters by truncating their number of consecutive repetitions to two.

A3.1 Multi-label Annotation

We consider a multi-label annotation scheme to deliver fine-grained hate speech categories and intersectionality from overlapping labels. The annotation scheme has two layers: (a) binary classification (*'Hate Speech' or 'Not Hate Speech'*) and (b) fine-grained classification (*8 labels or 'Not Hate Speech'*). For the fine-grained classification, a 'Hate Speech' class from the binary classification is broken down into 8 classes associated with the hate speech category [4]. As shown in Table A.1, this scheme allows non-exclusive concepts, accounting

[2] https://www.kaggle.com/datasets/junbumlee/kcbert-pretraining-corpus-korean-news-comments

[3] https://github.com/kocohub/korean-hate-speech

[4] Fine-grained labels (matching in Korean): Politics (정치성향차별), Origin (출신차별), Physical (외모차별), Age (연령차별), Gender (성차별), Religion (종교차별), Race (인종차별), and Profanity (혐오욕설)

for the overlapping shades of given categories. We select the 8 hate speech classes in order to reflect the social and historical context as the nature of hate speech is different in each language [74]. For example, the '*politics*' class is chosen due to a significant influence on the style of Korean hate speech.

A3.2 Annotation Instructions

Given the subjectivity of the task and our annotation scheme, we perform a preliminary round to identify the topics of hate speech and develop annotation instructions. We begin with the common categories of hate speech found in literature and match the keywords for each category. After the preliminary round, we investigate the results to merge or remove labels in order to provide the most representative subtype labels of hate speech contextual to the cultural background. Our annotation instruction includes the criteria as follows: **Politics**: hate speech based on political stance; **Origin**: hate speech based on place of origin or identity; **Physical**: hate speech based on physical appearance (e.g. body, face) or disability; **Age**: hate speech based on age; **Gender**: hate speech based on gender or sexual orientation (e.g. woman, homosexual); **Religion**: hate speech based on religion; **Race**: hate speech based on ethnicity; **Profanity**: hate speech in the form of swearing, cursing, cussing, obscene words, or expletives; or an unspecified hate speech category from above; and **Not Hate Speech**.

Our annotation instructions explain a two-layered annotation to (a) distinguish hate and not hate speech, and (b) the categories of hate speech. Annotators are requested to consider given keywords or alternatives of each category within social, cultural, and historical circumstances. For example, a comment using the word *"women"* is not hate speech, whereas, if it is critical of *"women"* or uses language that attacks the group, it is classified as *'gender'*. Notably, we annotate multi-labels if a comment includes several hate speech categories. Since hate speech can be varied, any comments in the form of swearing or cursing are marked as *'profanity'*. For instance, a comment containing hate speech about appearance, political stance, and gender in the profane language (e.g. *"fuck you ugly communist bitch."*)[5] is labeled within *'physical', 'politics', 'gender'* and *'profanity'* classes.

[5] (Korean) "면상도 개 조가치 생겼네 개빨갱이년"

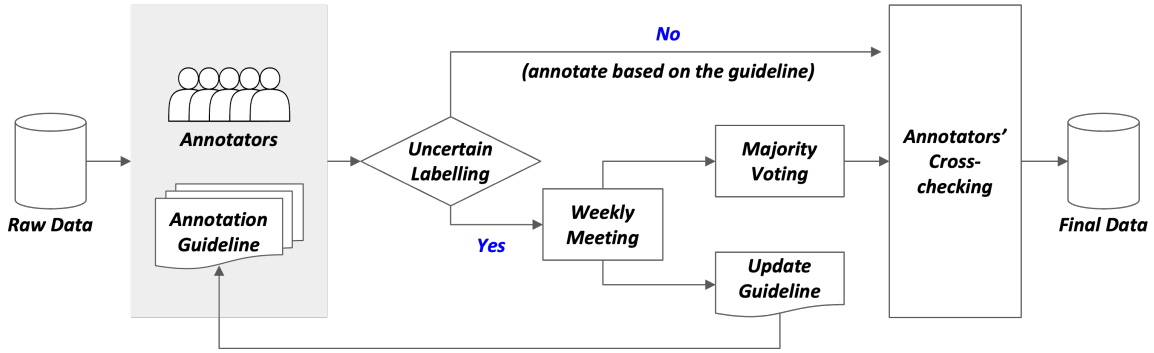


FIGURE A.1: Overview of Annotation Process.

A3.3 Annotation Process

Five native speakers were recruited for manual annotation in both the preliminary and main rounds. During the preliminary round, we facilitated the annotation instructions by conducting an annotators' discussion and providing some examples of keywords for each class. As shown in Figure A.1, we introduced an iterative process that enables faster annotation in the main round. We provided an '*uncertain*' additional field that was used for the unspecified label in annotation guidelines or when the annotator had difficulties in choosing labels. Any '*uncertain*' labeled data was flagged by individual annotators, and then reviewed by five annotators. The final labels were chosen based on the majority vote, and the annotation guidelines were updated to handle similar cases. Additionally, the other labeled data was reviewed, in line with the annotation guideline by two random annotators for the final dataset. The inter-annotator agreement returns an average Cohen Kappa score of 0.892, indicating substantial agreement [155].

A4 Dataset Analysis

K-MHaS dataset contains 109,692 comments as shown in Table A.2. For binary classification, the proportion of the '*hate speech*' (45.7%) and '*not hate speech*' (54.3%) satisfies data balancing. The '*hate speech*' label consists of a single label (33.2%) and multi-labels (12.4%), containing from 2 to 4 labels. Other hate speech datasets reviewed have an approximate ratio

Label Types		**Count (%)**
Total Utterances		**109,692 (100%)**
Multi-label (Hate Speech)	1 label (Single)	36,470 (33.2%)
	2 labels	12,073 (11.0%)
	3 labels	1,440 (1.3%)
	4 labels	94 (0.1%)
Not Hate Speech		59,615 (54.3%)

TABLE A.2: Dataset Statistics. The total is the combination of all '*hate speech*' and '*not hate speech*' label. Together the '*hate speech*' label makes up 45.7% of the data.

Class	**Count - Single (%)**	**Count - Multi (%)**
Politics	6,931 (19.0%)	4,961 (17.2%)
Origin	5,739 (15.7%)	4,458 (15.5%)
Physical	5,443 (14.9%)	3,364 (11.7%)
Age	4,192 (11.5%)	3,178 (11.0%)
Gender	3,348 (9.2%)	4,696 (16.3%)
Religion	1,862 (5.1%)	513 (1.8%)
Race	160 (0.4%)	163 (0.6%)
Profanity	8,795 (24.1%)	7,509 (26.0%)

TABLE A.3: Fine-grained label distributions on *hate speech* labels. A '*not hate speech*' label is not included. A single means 1 label and a multi is the sum of 2, 3, and 4 labels. A Multi-labeled data counts each overlapping class.

of '*hate speech*' to '*not hate speech*' of around 40% [169]. Our dataset is consistent with this figure, where the '*hate speech*' in a single label to '*not hate speech*' ratio is 38%.

The Korean language The Korean language is morphologically rich and the character structure is different from Latin-based language. A brief components used in the paper are as follows:

- **Consonant (자음)**: A consonant is a sound such as 'p', 'f', 'n', or 't' which you pronounce by stopping the air flowing freely through your mouth.
  - initial consonant (초성)
  - bottom consonant (받침)
- **Vowel (모음)**: A vowel is a sound such as the ones represented in writing by the letters 'a', 'e', 'i', 'o', and 'u', which you pronounce with your mouth open, allowing the air to flow through it.

- **Syllable (음절)**: A syllable is a part of a word that contains a single vowel sound and that is pronounced as a unit. So, for example, 'book' has one syllable, and 'reading' has two syllables.
  - Korean romanization : (e.g. [kko#t#baem])
  - Character level : (e.g. 꽃#뱀)
  - Sub-character level : (e.g. 꼬#ㅊ#뱀)

A4.1 Label Distribution

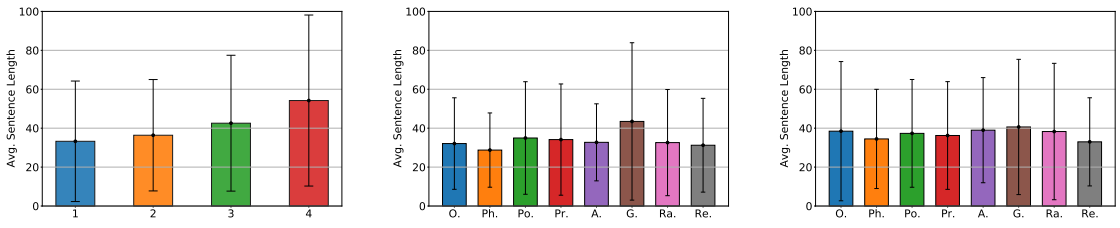
Table A.3 shows the fine-grained label distribution across our K-MHaS. For both single (s) and multi-label (m) distribution, the '*profanity*' class (24.1%-s, 26.0%-m) is more frequent than any other class, indicating that swear words are critical for detecting hate speech. Also, the '*religion*' (5.1%-s, 1.8%-m) and '*race*' (0.4%-s, 0.6%-m) classes are the smallest portions in both distributions, which are significantly more common in other hate speech datasets. This difference could be because Korea is a highly homogenous monoculture with little variation in race and religion [74]. Interestingly, the '*gender*' class (16.3%) occurs at almost twice the frequency in a multi-label distribution, compared to a single-label distribution (9.2%). This indicates that gender-based hate speech is used extensively in combined aspects.

A4.2 Keyword Analysis

To understand the lexical aspects, we list the top 5 keywords for each hate speech category in Table A.4, identifying which tokens are highly associated with each class. In the '*politics*' class, we find that far-right extremism is dominant, and new tokens such as *"catastrophe"* [jae ang](재앙) appear related to the former president's given name ([jae in]) as the two words are near-homophones. Across all classes, one-word tokens are often used in their stem form to modify the meanings of other words. For example, a token [teul] (틀) comes from the word *"denture"* [teulni] (틀니) which is used as an offensive reference to the elderly. In addition, one-word tokens can be used as a prefix (e.g. "dog" [gae] (개)) or a suffix (e.g. "insect" [chung] (충)), and combined with other neutral words to create a new offensive term.

Rank	Politics	Origin	Physical	Age
1	재앙 (1427)	짱깨 (615)	얼굴 (962)	틀 (1918)
2	문재인 (951)	전라도 (596)	돼지 (772)	나이 (599)
3	좌파 (464)	중국 (539)	여자 (294)	노인 (139)
4	좌빨 (402)	쪽 (448)	성형 (216)	충 (112)
5	빨갱이 (367)	짱 (446)	관상 (183)	놈 (106)
Rank	**Gender**	**Religion**	**Race**	**Profanity**
1	여자 (1704)	개독 (526)	흑인 (44)	새끼 (1103)
2	남자 (990)	신천지 (460)	백인 (32)	년 (1014)
3	페미 (172)	사이비 (409)	양키 (32)	지랄 (564)
4	맘충 (138)	종교 (305)	깜둥이 (19)	개 (459)
5	여성 (134)	예수 (227)	놈 (13)	놈 (404)

TABLE A.4: Top 5 keywords associated with each fine-grained label. The number in brackets is the token count. The keyword analysis is from the total dataset and is different from some examples in annotation guidelines.



(A) Label types (B) Class types I (C) Class types II

FIGURE A.2: Average utterance length. (a) label types from 1 to 4 labels. 8 class types (b) in a single label and (c) in multi-labels.

A4.3 Label Pair Analysis

Figure A.2 shows the average length of utterance by label count and class type. The total average length of an utterance is 33 tokens. An increase in the number of labels shows an increasing trend in utterance length, indicating that multi-labeled hate speech contains more linguistic content. The *'gender'* class has relatively longer lengths (43 tokens) compared to other classes in a single label, whereas all multi-label utterances have a similar length. This indicates that the gender class has different linguistic features.

A5 Experiment Setup

A5.1 Data Preparation

We split the data into train/test in the proportions of 0.8/0.2. From the training set, we randomly select 0.1 as a validation set (78,977/8,776/21,939 samples for train/val/test sets, preserving the class proportion). The data passed to the models is the preprocessed sentences and binary label vectors.

A5.2 Baselines

We select four baselines. 1) **Multi-BERT** [184] is pre-trained on Wikipedia in 104 different languages. We adopted the BERT-Base, which uses the WordPiece tokenizer and contains 110M parameters and 119K vocabs. 2) **KoELECTRA** [128] is pre-trained on 34GB Korean news, Korean Wikipedia, Namuwiki (Korean-based wiki) and Modu (Korean corpus data publicly provided by the Korean government). The KoELECTRA-Small-v3 is used with the WordPiece tokenizer and contains 14M parameters and 35K vocabs. 3) **KoBERT** [157] is pre-trained on 54M words from Korean Wikipedia, using the SentencePiece tokenizer, 92M parameters and 8K vocabs. 4) **KR-BERT** [87] is pre-trained on 2.47GB corpus with 233M words from Korean Wikipedia and news. We applied either (1) the character-level tokenizer or (2) the sub-character-level tokenizer[6].

A5.3 Evaluation Metrics

In multi-label classification, the prediction contains a set of labels, which means the prediction can be fully correct, partially correct, or fully incorrect. We propose to use the widely used six metrics [52] for conducting our multi-label classification, including F1-[macro, micro, weighted], Exact Match, AUC and Hamming Loss [158].

[6]The KR-BERT tokenization variants can be found as follows: https://github.com/snunlp/KR-BERT#tokenization

A5.4 Implementation Details

For all baselines, we set the number of epochs as 4 and use a batch size of 32. For other hyper-parameters, we follow the configuration in the official GitHub implementation of the baselines. The source codes or pre-trained models for the baselines are available at the following GitHub addresses: Multilingual BERT[7], KoELECTRA[8], KoBERT[9] and KR-BERT[10].

A6 Results

A6.1 Evaluation for All Labels

The overall performance for all labels is provided in Table A.5. The F1(micro) range between 0.8139 (Multi-BERT), 0.8493 (KoELECTRA) and 0.8500 (KR-BERT-c), while the F1(macro) scores show a range from 0.6912 (Multi-BERT) to 0.7651 (KoBERT) with 4 epochs. We observe that all baselines achieve similar performance, whereas Multi-BERT pre-trained on 104 languages presents a relatively lower performance. The KoELECTRA obtains overall the best or second best among six metrics, although this model has a seven times smaller parameter size (14M) than the average of other models (99M). This indicates the effects of the pre-training data source, considering that the KoELECTRA includes the corpus from Namuwiki and Modu that contain modern slang and buzzwords, while other models generally use Korean Wikipedia.

A6.2 Evaluation for Multi-labels

Table A.8 shows the breakdown F1(micro) for multi-label classification from 1 to 4 labels[11]. A single-label task, achieving 0.8553 and 0.8490 from the KR-BERT-c and KoELECTRA,

[7]https://github.com/google-research/bert
[8]https://github.com/monologg/KoELECTRA
[9]https://github.com/SKTBrain/KoBERT
[10]https://github.com/snunlp/KR-BERT
[11]Further details are shown in Appendix Table A.9.

Model	F1 (macro)	F1 (micro)	F1 (weighted)	E.M.	AUC	H.L. (↓)
BERT	0.6912	0.8139	0.8119	0.7579	0.8878	0.0464
KoELECTRA	0.7245	0.8493	**0.8480**	**0.7994**	**0.9122**	0.0380
KoBERT	**0.7651**	0.8413	0.8424	0.7926	0.9083	0.0401
KR-BERT-c	0.7444	**0.8500**	0.8470	0.7901	0.9028	**0.0368**
KR-BERT-s	0.7245	0.8445	0.8437	0.7825	0.9076	0.0390

TABLE A.5: Overall multi-label classification performance on K-MHaS for the five baseline models at epoch 4 (E.M.:Exact Match, H.L.:Hamming Loss / KR-BERT-*: c = character-level, s = sub-character-level)

Model	F1 (Macro)	F1 (Micro)	F1 (Weighted)	E.M.	AUC	H.L. (↓)
BERT	0.8495	0.8507	0.8505	0.8507	0.8488	0.1493
KoELECTRA	0.8756	0.8766	0.8765	0.8766	0.8750	0.1234
KoBERT	0.8687	0.8692	0.8693	0.8692	0.8696	0.1308
KR-BERT (w. char)	0.8846	0.8850	0.8851	0.8850	**0.8862**	0.1150
KR-BERT (w. sub)	**0.8869**	**0.8879**	**0.8877**	**0.8879**	0.8857	**0.1121**

TABLE A.6: Overall binary classification performance on the K-MHaS dataset for the five pre-trained language models at epoch 4 (E.M.:Exact Match, H.L.:Hamming Loss / KR-BERT (w. *): char = character-level, sub = sub-character-level)

Label	Model	F1 (Macro)	F1 (Micro)	F1 (Weighted)	E.M.	H.L. (↓)
Hate Speech	BER	0.4518	0.8243	0.9037T	0.8243	0.1757
	KoELECTRA	0.4606	0.8540	0.9212	0.8540	0.1460
	KoBERT	0.4666	0.8746	**0.9331**	0.8746	0.1254
	KR-BERT (w. char)	0.4611	0.8558	0.7892	0.8558	0.1442
	KR-BERT (w. sub)	**0.4724**	**0.8953**	0.8458	**0.8953**	**0.1047**
None	BERT	0.4662	0.8733	0.9323	0.8733	0.1267
	KoELECTRA	0.4726	0.8960	**0.9452**	**0.8960**	0.1040
	KoBERT	0.4637	0.8645	0.9273	0.8645	0.1355
	KR-BERT (w. char)	**0.4772**	**0.9126**	0.8709	**0.9126**	**0.0874**
	KR-BERT (w. sub)	0.4687	0.8821	0.8268	0.8821	0.1179

TABLE A.7: A breakdown of binary classification performance on the K-MHaS dataset for the five pre-trained language models at epoch 4 (E.M.:Exact Match, H.L.:Hamming Loss / KR-BERT (w. *): char = character-level, sub = sub-character-level, bi = Bidirectional WordPiece tokenizer)

outperforms other multi-label tasks due to domain similarity. For the multi-label classification, KR-BERT-s achieved the best performance. It uses a sub-character tokenizer that can decompose Hangul(Korean language) syllable characters into sub-characters. Therefore, it provides greater granularity in detecting hate speech words, by identifying the sub-characters from different hate speech categories.[12].

[12](e.g.) 개빠ㄹ갱이년 = 개 ("dog" - *profanity*) + 빠ㄹ갱이 ("communist" - *politics*) + 년 ("bitch" - *gender*)

Metric	# labels	BERT	KoELECTRA	KoBERT	KR-BERT-c	KR-BERT-s
F1 (micro)	1	0.8190	0.8490	0.8320	**0.8553**	0.8392
	2	0.8043	0.8612	**0.8854**	0.8405	0.8703
	3	0.7517	0.7987	0.8290	0.7827	**0.8329**
	4	0.7093	0.7044	0.6832	0.7439	**0.7771**

TABLE A.8: A breakdown of F1 for multi-label classification from 1 to 4 labels.

# Labels	Model	F1 (Macro)	F1 (Micro)	F1 (Weighted)	E.M.	AUC	H.L. (↓)
1	BERT	0.6666	0.8190	0.8202	0.7919	0.9011	0.0406
	KoELECTRA	0.6953	0.8490	0.8508	**0.8263**	**0.9213**	0.0341
	KoBERT	0.7321	0.8320	0.8370	0.8142	0.9110	0.0379
	KR-BERT(w. char)	**0.7336**	**0.8553**	**0.8543**	0.8239	0.9145	**0.0318**
	KR-BERT(w. sub)	0.6985	0.8392	0.8419	0.8062	0.9123	0.0360
2	BERT	0.6389	0.8043	0.8174	0.5580	0.8524	0.0788
	KoELECTRA	0.6777	0.8612	0.8700	0.6511	0.8934	0.0577
	KoBERT	**0.7249**	**0.8854**	**0.8911**	**0.6794**	**0.9112**	**0.0482**
	KR-BERT(w. char)	0.6748	0.8405	0.8451	0.5912	0.8735	0.0642
	KR-BERT(w. sub)	0.6718	0.8703	0.8723	0.6535	0.9000	0.0542
3	BERT	0.5784	0.7517	0.7522	0.2448	0.8040	0.1402
	KoELECTRA	0.6146	0.7987	0.7953	0.3310	0.8362	0.1169
	KoBERT	**0.6523**	0.8290	0.8251	**0.3759**	0.8589	0.1019
	KR-BERT(w. char)	0.5828	0.7827	0.7732	0.2828	0.8239	0.1230
	KR-BERT(w. sub)	0.6164	**0.8329**	**0.8263**	0.3586	**0.8615**	**0.0996**
4	BERT	0.4776	0.7093	0.7029	**0.1200**	0.7610	0.2222
	KoELECTRA	0.4511	0.7044	0.6639	0.0000	0.7680	0.2089
	KoBERT	0.4177	0.6832	0.6460	0.0400	0.7510	0.2267
	KR-BERT(w. char)	0.4837	0.7439	0.7226	**0.1200**	0.7930	0.1867
	KR-BERT(w. sub)	**0.5068**	**0.7771**	**0.7618**	**0.1200**	**0.8120**	**0.1733**

TABLE A.9: A breakdown of multi-label classification performance from 1 to 4 labels on K-MHaS for the five pre-trained language models at epoch 4 (E.M.:Exact Match, H.L.:Hamming Loss / KR-BERT (w. *): char = character-level, sub = sub-character-level)

A6.3 Evaluation for Label-pairs

Table A.10 shows the F1-[macro, micro] scores for curated label pairs based on the proportion in the 2-label classification. It illustrates that the KR-BERT-s model outperforms six label pairs. In particular, it is very effective at detecting the *origin and gender* pairs, achieving the highest F1 micro scores of 0.9494 across all label pairs and models. This model uses the sub-character-level tokenizer that can decompose various Korean characters (Hangul syllables) into sub-characters or graphemes to enable handling the bottom consonant (e.g. "gold-digger" [kko#t#baem] 꼬#ㅊ#뱀) or initial consonant (e.g. [k] ㅋ). This approach can detect new slang even if it is only a minor variation from other neutral words.

Further experiment results are displayed as follows:

Label Pairs	# pairs	F1 (macro)					F1 (micro)				
Overall Performance (F1)		**BERT**	**KoELECTRA**	**KoBERT**	**KR-BERT-c**	**KR-BERT-s**	**BERT**	**KoELECTRA**	**KoBERT**	**KR-BERT-c**	**KR-BERT-s**
		0.6912	0.7245	**0.7651**	0.7444	0.7245	0.8139	0.8493	0.8413	**0.8500**	0.8445
Profanity & Politics	323	0.1959	0.2045	**0.2072**	0.2013	0.2034	0.8379	0.8853	**0.9010**	0.8687	0.8616
Profanity & Physical	311	0.1931	0.2061	0.2115	0.2099	**0.2121**	0.8393	0.9096	0.9331	**0.9369**	0.9334
Profanity & Origin	269	0.1887	0.1989	0.1987	0.1961	**0.2050**	0.8144	0.8731	0.8729	0.8661	**0.9070**
Gender & Origin	242	0.2035	0.2017	0.2134	0.1905	**0.2141**	0.8920	0.8780	0.9440	0.8354	**0.9494**
Politics & Origin	224	0.1962	0.1976	0.1991	0.1872	**0.2013**	0.8666	0.8714	0.8846	0.8295	**0.8918**
Age & Politics	222	0.1996	**0.2114**	0.2104	0.1964	0.2014	0.8765	0.9329	**0.9357**	0.8734	0.8878
Gender & Profanity	181	0.1895	0.1991	0.1957	0.1911	**0.2054**	0.8157	0.8715	0.8542	0.8450	**0.8994**
Gender & Physical	177	0.1160	0.1833	**0.1958**	0.1813	0.1953	0.4562	0.7867	0.8455	0.8045	**0.8585**
Age & Profanity	132	0.1908	0.2102	**0.2139**	0.2063	0.2043	0.8414	0.9240	**0.9459**	0.9105	0.9095
Gender & Age	130	0.1738	0.1781	**0.1903**	0.1517	0.1339	0.7277	0.7452	**0.8159**	0.6368	0.5686

TABLE A.10: F1 score for the top 10 two-label pairs on the K-MHaS dataset for the five pre-trained language models at epoch 4 (# total label pairs = 2,439 / KR-BERT-*: c = character-level tokenizer, s = sub-character-level tokenizer).

Label Triplets	# triplets	F1 (macro)					F1 (micro)				
Overall Performance (F1)		**BERT**	**KoELECTRA**	**KoBERT**	**KR-BERT-c**	**KR-BERT-s**	**BERT**	**KoELECTRA**	**KoBERT**	**KR-BERT-c**	**KR-BERT-s**
		0.6912	0.7245	**0.7651**	0.7444	0.7245	0.8139	0.8493	0.8413	**0.8500**	0.8445
Origin & Politics & Profanity	41	0.2780	0.2935	0.2954	0.2937	**0.3125**	0.8224	0.8739	0.8869	0.8739	**0.9316**
Politics & Profanity & Age	37	0.2781	0.3054	**0.3174**	0.2981	0.2971	0.8205	0.9126	**0.9395**	0.8867	0.8824
Physical & Politics & Profanity	32	0.2483	0.2809	**0.2960**	0.2823	0.2939	0.7296	0.8304	0.8750	0.8421	**0.8764**
Origin & Profanity & Gender	30	0.2545	0.2314	0.2527	0.2463	**0.2886**	0.7467	0.7397	0.7368	0.7671	**0.8712**
Physical & Profanity & Gender	24	0.2151	0.2665	0.2730	0.2459	**0.2811**	0.6306	0.7869	0.8226	0.7521	**0.8413**
Origin & Physical & Profanity	14	0.2692	0.2811	**0.2873**	0.2351	0.2865	0.7532	0.8378	0.8312	0.7273	**0.8421**
Politics & Age & Gender	14	0.2593	**0.2933**	0.2830	0.2319	0.2406	0.7606	**0.8684**	0.8158	0.6970	0.7059
Profanity & Age & Gender	13	0.2686	0.2712	0.2932	0.2593	**0.2695**	0.8182	0.8060	**0.8732**	0.8000	0.8358
Origin & Physical & Gender	12	0.2692	0.2327	0.2407	0.2347	**0.2703**	0.7812	0.6780	0.7458	0.7143	**0.8065**
Origin & Age & Gender	10	0.2736	0.2781	**0.3093**	0.2428	0.2411	0.8235	0.8302	**0.9123**	0.7347	0.7600

TABLE A.11: F1 score for the top 10 three-label pairs on the K-MHaS dataset for the five pre-trained language models at epoch 4 (# total label triplets = 290 / KR-BERT-*: c = character-level WordPiece tokenizer, s = sub-character-level WordPiece tokenizer)

Label Quadruplets	# quadruplets	F1 (macro)					F1 (micro)				
Overall Performance (F1)		**BERT**	**KoELECTRA**	**KoBERT**	**KR-BERT-c**	**KR-BERT-s**	**BERT**	**KoELECTRA**	**KoBERT**	**KR-BERT-c**	**KR-BERT-s**
		0.4776	0.4511	0.4177	0.4837	**0.5068**	0.7093	0.7044	0.6832	0.7439	**0.7771**
Origin & Profanity & Age & Gender	5	**0.3567**	0.2950	0.2346	0.2932	0.3086	**0.8000**	0.6875	0.5806	0.7500	0.7879
Origin & Physical & Profanity & Gender	4	0.2582	0.2804	0.2508	0.3302	**0.3757**	0.5385	0.7200	0.6400	0.7692	**0.8571**
Origin & Physical & Politics & Profanity	3	**0.4000**	0.3333	0.3111	0.3333	0.3667	**0.9091**	0.8000	0.7368	0.8000	0.8571
Origin & Politics & Profanity & Age	3	0.4222	0.3444	0.3667	**0.4444**	**0.4444**	0.8800	0.8000	0.8571	**1.0000**	0.9600
Origin & Politics & Profanity & Gender	2	0.1852	**0.2963**	0.2593	**0.2963**	**0.2963**	0.5455	**0.7692**	0.6667	0.6667	**0.7692**

TABLE A.12: F1 score for the top 5 four-label pairs on the K-MHaS dataset for the five pre-trained language models at epoch 4 (# total label quadruplets = 25 / KR-BERT-*: c = character-level WordPiece tokenizer, s = sub-character-level WordPiece tokenizer)

- Table A.9: a breakdown of multi-label classification performance from 1 to 4 labels;
- Table A.6: overall binary classification performance;
- Table A.7: a breakdown of binary classification performance;
- Table A.7: F1 score for the top 10 three-label pairs in 3-labels classification;
- Table A.12: F1 score for the top 5 four-label pairs in 4-labels classification.

A7 Conclusion

We propose K-MHaS, a new large-sized dataset for Korean hate speech detection with a multi-label annotation scheme. We provided extensive baseline experiment results, presenting the usability of a dataset to detect Korean language patterns in hate speech. In future work, the automatic hate speech moderation and counter-speech can be expanded.

Ethics/Broader Impact Statement The study follows the ethical policy set out in the ACL code of Ethics[13] and addresses the ethical impact of presenting a new dataset. As described in the data section A3, our annotated dataset is based on the online news comments data publicly available on Kaggle and Github. All annotators were recruited from a crowdsourcing platform. They were informed about hate speech before handling the data. Our instructions allowed them to feel free to leave if they were uncomfortable with the content. With respect to the potential risks, we note that the subjectivity of human annotation would impact the quality of the dataset.

[13]https://www.aclweb.org/portal/content/acl-code-ethics