applied sciences

MDPI

Article

Fine-Tuning Retrieval-Augmented Generation with an Auto-Regressive Language Model for Sentiment Analysis in Financial Reviews

Miehleketo Mathebula *, Abiodun Modupe and Vukosi Marivate

Department of Computer Science, University of Pretoria, Lynnwood Road, Pretoria 0002, South Africa; abiodun.modupe@up.ac.za (A.M.); vukosi.marivate@up.ac.za (V.M.)
* Correspondence: miehleketo.mathebula@tuks.co.za

Abstract: Sentiment analysis is a well-known task that has been used to analyse customer feedback reviews and media headlines to detect the sentimental personality or polarisation of a given text. With the growth of social media and other online platforms, like Twitter (now branded as X), Facebook, blogs, and others, it has been used in the investment community to monitor customer feedback, reviews, and news headlines about financial institutions' products and services to ensure business success and prioritise aspects of customer relationship management. Supervised learning algorithms have been popularly employed for this task, but the performance of these models has been compromised due to the brevity of the content and the presence of idiomatic expressions, sound imitations, and abbreviations. Additionally, the pre-training of a larger language model (PTLM) struggles to capture bidirectional contextual knowledge learnt through word dependency because the sentence-level representation fails to take broad features into account. We develop a novel structure called language feature extraction and adaptation for reviews (LFEAR), an advanced natural language model that amalgamates retrieval-augmented generation (RAG) with a conversation format for an auto-regressive fine-tuning model (ARFT). This helps to overcome the limitations of lexicon-based tools and the reliance on pre-defined sentiment lexicons, which may not fully capture the range of sentiments in natural language and address questions on various topics and tasks. LFEAR is fine-tuned on Hellopeter reviews that incorporate industry-specific contextual information retrieval to show resilience and flexibility for various tasks, including analysing sentiments in reviews of restaurants, movies, politics, and financial products. The proposed model achieved an average precision score of 98.45%, answer correctness of 93.85%, and context precision of 97.69% based on Retrieval-Augmented Generation Assessment (RAGAS) metrics. The LFEAR model is effective in conducting sentiment analysis across various domains due to its adaptability and scalable inference mechanism. It considers unique language characteristics and patterns in specific domains to ensure accurate sentiment annotation. This is particularly beneficial for individuals in the financial sector, such as investors and institutions, including those listed on the Johannesburg Stock Exchange (JSE), which is the primary stock exchange in South Africa and plays a significant role in the country's financial market. Future initiatives will focus on incorporating a wider range of data sources and improving the system's ability to express nuanced sentiments effectively, enhancing its usefulness in diverse real-world scenarios.

Keywords: large language models; sentiment analysis; retrieval-augmented generation; prompt engineering; conversational fine-tuning; retrieval augmented generation assessment; auto-regressive LLM

check for updates

Citation: Mathebula, M.; Modupe, A.; Marivate, V. Fine-Tuning Retrieval-Augmented Generation with an Auto-Regressive Language Model for Sentiment Analysis in Financial Reviews. *Appl. Sci.* **2024**, *14*, 10782. https://doi.org/10.3390/app142310782

Academic Editors: Carlos J. Costa and Manuela Aparicio

Received: 15 October 2024
Revised: 13 November 2024
Accepted: 14 November 2024
Published: 21 November 2024

1. Introduction

Social media platforms such as Twitter (now X), Facebook (now Meta), YouTube, and TikTok are critical in today's digital age for consumers to share experiences and engage with their communities. In this current data-driven economy, analysing consumer data from social media is vital for stakeholders such as investors, regulators, and financial institutions to grasp societal viewpoints on financial products and services. Messages or comments posted on social media are valid tools that facilitate real-time communication, engagement, and connection with a large global audience. Analysing these data provides valuable insights for businesses, marketers, and researchers interested in understanding public opinions and trends. In South Africa, the penetration of social media in the hospitality sector accounts for 67% of all tourism, while the wholesale, retail, catering, and accommodation sectors contribute 14.4% to the country's GDP [1]. In comparison to the neighbouring country, Zimbabwe, the Digital 2022 report recorded that there were 4.65 million internet users in Zimbabwe in January 2022 and 1.55 million social media users, which translates to 10.2% of the total population [2]. Moreover, social media are a key source of information for citizens and activists in Zimbabwe. In Northern Africa, approximately 40.4% of the population utilises social media, compared to 41.6% in Southern Africa [3]. In Central Africa, just under 10% of the population engages with social media, marking the lowest rate in Africa and globally [4]. A study by Macro-Monitor in 2022 [5] identified comparable patterns in the extent to which American households, homeowners, and business professionals utilise social media for information exchange on financial institutions. The study found that 52% of respondents born in the US after 1998 shared information on social media. This proportion decreases as households age drops to 16% when the household owner was born before 1960.

Sentiment analysis is crucial to understanding the users' post or comments, which comprise text, emojis, and hashtags, which can provide valuable information about the user's polarity towards specific needs, topics, or events. Sentiment analysis requires data labelling for the training of the machine learning (ML) algorithms. These data labels should be of high quality so that the ML classifier can be closer to the ground truth. One popular method to obtain the gold standard labelled dataset is using crowdsourcing or workers for their annotation [6,7]. However, the use of crowdsourcing for data annotation offers a more affordable solution, allows a variety of perspectives and experiences, and is ultimately very scalable. In contrast, the accuracy and consistency of the annotations can vary widely with low-quality annotations and lead to misinterpretations and errors; e.g., there is no contextual understanding. Amazon Mechanical Turk (MTurk) [8] is another common way to generate a high-quality labelled dataset. This approach is less reliable and accurate for understanding context, annotating text, and learning word dependencies. Additionally, because online social media texts are rich in slang, emojis, hashtags, idioms, and acronyms, using the MTurk approach will produce low-quality annotations. Although the cost of using MTurk may be very low, the time needed to develop the interface and tackle the spammer problem is not negligible; validation and correction costs to ensure minimal quality are very high [9].

Lexicon-based methods (LBMs) are often employed to determine sentiment in text since they are simpler and faster than supervised learning approaches. Popular lexicon-based methods include VADER, SentiWordNet, and LIWC [10,11]. For example, VADER is a rule-based method that is efficient and beats other approaches to social media data, particularly Twitter (now X). It also considers emojis when calculating sentiment scores. However, VADER does not consider the type of emoji, e.g., the specific emoji used, and utilises the values or scores to determine the polarity of text or posts containing such emojis. The limitation of the lexicon-based approach for sentiment analysis is their inability to handle sarcasm, irony, and other forms of figurative text or language. These algorithms rely on a predefined list of words from LIWC, WSD, and others by finding their associated sentiment score to determine or rank the expression in a natural language found in the given text. However, the polarity outcomes from most of these approaches are incorrect or inaccurate in reflecting the subjectivity of the provided text or social media post, let alone the products and services of a financial institution. Often, words have multiple non-literal meanings that these approaches may struggle to comprehend. As a result, the lexicon-based approach frequently misclassifies sentiment because it has difficulty understanding the context and tone of the text in order to convey subjective feelings in a manner that humans can understand naturally. Also, the recent pre-trained large language models (PTLMs) include Word2Vec [12] and GloVe [13], but these methods ignore the context of the words [14]. Although Word2Vec and GloVe are rich in the linguistic structure of the given text or sentence, their knowledge pertaining to lexical polysemy remains unclear. Similarly, bidirectional encoder representations from transformers (BERT)s [15] capture the entire sequence of words in a sentence at once [16], but it is still not clear whether or not BERT models preserve information about lexical polysemy and sense distinctions [16]. PTLMs are effective in generating human-like text compared to lexicon-based models, which makes them appropriate to extract sentiments from financial data to facilitate investor decision making [16] but not to evaluate financial institutions based on product and service feedback to influence the market decision or electronic word-of-mouth (eWOM) [11]. However, PTLMs still have difficulty capturing sentiment analysis tasks for financial products because they rely on pre-defined sentiment lexicons, which might not be able to capture all types of sentiment expressed in natural language. Also, the models were not trained to understand the intricate details of financial language [17,18].

Moreover, PTLMs for sentiment analysis tasks are evaluated based on accuracy, F1 score, recall, and precision. While these metrics are generally accepted within the broader LBMs and machine learning communities, they often fail to capture the intensity and granularity of a statement in human understanding, particularly in financial text (e.g., online social media text) in order to determine whether the sentiment is positive, negative, or neutral [19,20]. To address this gap, our study uses new metrics called retrieval-augmented generation assessment (RAGAS) combined with sentiment intensity, the level of detail in sentiment classification, and the Vendi score with auto-regressive language models (ARLMs) such as Llama-2 (available online: https://llama.meta.com/llama2/ (accessed on 18 September 2024)), Llama-3 (available online: https://llama.meta.com/ (accessed on 18 September 2024)), OpenAI GPT-3.5-Turbo (available online: https://platform.openai.com/docs/models/gpt-3-5-turbo (accessed on 18 September 2024)) and OpenAI GPT-4o-Mini (available online: https://platform.openai.com/docs/models/gpt-4o-mini(accessed on 18 September 2024)) that have the capability to predict the next sequence of words in a sentence or text, and they have significantly expedited recent developments in NLP tasks like sentiment analysis.

In this paper, we present a language feature extraction and adaptation for reviews (LFEAR) based on ARLMs, which we combine with retrieval-augmented generation (RAG) and fine-tuning in conversation structure. RAG aids in better understanding natural language in text, while the conversation structure uses ARLMs to capture word dependencies to improve the overall performance of NLP such as sentiment analysis. In response to challenges in sentiment analysis of financial product reviews, LFEAR is designed to be adaptable and high-performing across various domains, extending its effectiveness beyond its original target. LFEAR employs techniques such as fine-tuning in conversation format, efficient prompts, and continuous learning, then trains on products and reviews based on South African financial institutions to achieve 92.76% accurate outcomes on the long form of sentiment on financial reviews on Hellopeter. These enhancements underscore the critical necessity for sentiment analysis models that are detailed yet adaptable beyond their original training parameters. The primary contributions of this study can be summarised as follows:

- We introduce LFEAR, a brand-new advanced inference model that uses conversational fine-tuning to auto-regressive LLMs like Llama-2, Llama-3, GPT-3.5-Turbo, and GPT-4o Mini to fit the complex analysis of South African financial product reviews.
- LFEAR amalgamates, with prompt engineering, RAG and continuous learning to handle sarcasm features and calculate sentiment in other domains, such as financial news, political discourse, restaurant reviews, hotel feedback, and movie ratings.
- We use the majority voting method to evaluate LFEAR on a wide range of different datasets and measure the relationship between public opinion and institutional performance. This demonstrates that the LFEAR is adaptable and strategic relevance across various contexts.

- LFEAR improves sentiment analysis performance evaluations by incorporating RAGAS, sentiment intensity, granularity classification, and vendi score, enhancing customer responses and bridging the gap in current financial sentiment analysis metrics.

The remainder of this paper is organised as follows. Section 2 provides a brief review of the background and related work, highlighting previous studies on sentiment analysis in the financial sector. Section 3 details our methodology and describes the proposed framework for data collection, preprocessing, and model training. In Section 4, we present the performance evaluation of the models and offer a comparative analysis based on various metrics. In section 5, we discuss the results of the study, outline the applications of LFEAR, and outlines directions for future research Finally, Section 6 concludes the study.

2. Related Works

There are an excess of data in the twenty-first century, which has resulted in a digital infusion of technology, paving the way for the growth of big data [21]. People all over the world are becoming more electronically sophisticated, using devices like digital cameras with sensor capability, smartphones, and communication tools to gain access to social media for disseminating information within their community, and this has increased the number of data processing actuators [22,23]. In the age of big data, the use of sentiment analysis has proven effective for categorising public attitudes into various moods and assessing public mood [24]. In the subsequent section, we look at the sentiment analysis in the financial sector in Section 2.1. Thereafter, we explore the lexicon or rule-based approaches in Section 2.2 that leverage predefined sentiment dictionaries or rules to assign sentiment scores. These approaches are easy to put into practice, require few computational resources, and are tailored to specific domains. However, these approaches may face difficulties in capturing subtle sentiments and need the inclusion of new words and terms in lexicons when dealing with new domains. On the other hand, most ML approaches extract information from labelled data and offer higher accuracy. However, they need significant training on data specific to each domain and have to retrain models when dealing with new domains. In Section 2.3, we look at deep learning approaches that have the capability to learn intricate patterns automatically, understand context, and demonstrate outstanding performance. Nevertheless, they demand significant resources and lack interpretation capability. Finally, we show the LFEAR model, which helps improve natural language models through conversation structure and RAG.

2.1. Financial Sentiment Analysis

As researchers sought to understand the nuanced expressions and idiosyncratic language of finance-related writing, the field of financial sentiment analysis began to go into business. Yet even the best-performing model we have to date, the pre-trained language model FinBERT, which massively outperforms our most universal models when we perform any task related to finance, has proven unable to keep up with the language of the slowly but constantly evolving world of finance itself [25]. Meanwhile, a recent paper from Gite et al. [26] proposed a method of improving stock price prediction that fuses the stock market's sentiment, as expressed in the headlines of relevant news articles, with price-relevant data about the stocks themselves. Yet while this approach aims for only a narrow slice of the market, it has been shown to be highly reliant on historical data and offers an inadequate framework for the kind of real-time shifts in market sentiment that high-frequency traders obsess over.

Changing market conditions affect financial sentiment models, necessitating the development of models that can adapt to these changes without relying solely on static, pre-labelled datasets. Sharaff et al. [27] have taken the LSTM framework with word embeddings, which had already been converted into a financial news sentiment tool and have pushed its performance into an even more favourable realm relative to many current models. This is, no doubt, due to the crucial role that context and temporal dependencies play when attempting to extract meaning from the often dense and terse language of such articles. However, what appears at first blush to be a model with high performance is one with very limited applicability to the fast-paced world of finance due to its heavy reliance on long-term, semi-static historical datasets. Mishev et al. [28] compared various methods of sentiment analysis, especially the most advanced models. Even though today's transformer-based models perform really well, the authors pointed out that some genuinely fixable problems need to be addressed. They mainly involved the kind and amount of data available for training these models. But they also highlighted something else that we thought was important, which is the issue of what one does with the model's output, particularly in real-time decisions that could have significant financial implications.

Additionally, present-day models often overlook the specialised financial vocabulary, which encompasses not only the distinctive terminology of the domain but also the abbreviations and shifts in sentiment driven by the context that make up the world of finance. Liu et al. [29] performed a sentiment analysis on social media data using a novel approach and, in the process, extracted investor sentiment from these otherwise undervalued reservoirs of information. After conducting a thorough validation of their tools, they found that the optimal way of interpreting sentiments from these networks is not as "positive", "neutral", or "negative", but instead in more contextually relevant categories, which are "buy", "hold" or "sell". Ardekani et al. [30] introduced a financial sentiment model that uses contextual language processing, but their work still does not address the inherent difficulties tied to performing aspect-based sentiment analysis, which is absolutely crucial for interpreting the multi-faceted nature of financial data.

2.2. Lexicon-Based Approaches

Lexicon-based approaches are popular for detecting sentiment in text because they are simpler and faster than supervised learning methods. Yue et al. [31] conducted a study comparing the performance of supervised and unsupervised machine learning techniques for sentiment analysis from a given set of Twitter messages. Their findings revealed that supervised methods generally exhibit superior accuracy to unsupervised approaches like lexicon-based algorithms. Nonetheless, acquiring sufficient labelled training data for supervised methods can be costly and time-consuming.

Kiritchenko et al. [32] used a lexicon to find (a) the mood of short, casual text messages like tweets and SMS on the SemEval-2013 dataset (message-level task) and (b) the mood of a word or phrase within a message (term-level task). On top of that, the researcher employs commonly used statistical-style features such as word, character, elongated, punctuation, and POS tag counts, as well as common Twitter-specific attributes such as emoticons and hashtag counts developed to handle negation. The scheme is tested by selecting 1,455 high-frequency terms from the Sentiment140 Corpus and the Hashtag SentimentCorpus, which includes 1.6 million tweets with positive and negative sentiment labels. The data consist of regular English words, Twitter-specific terms (e.g., emoticons, abbreviations, and creative spellings), and negated natural expressions manually annotated using MaxDiff (available online: https://saifmohammad.com/WebPages/lexicons.html (accessed on 1 October 2024)) to assign a score to the most prominent words in the individual tweet and the SMS [33]. The approach achieved a macro-averaged F-score of 69.02% in the message-level task and 88.93% in the term-level assignment. A linear-kernel support vector machine (SVM) was used to train the SMS messages. The system achieved an F-score of 70.45% for the message-level task and an F-score of 89.50% for the term-level task. It came in second for detecting the sentiment of terms within SMS messages (F-score of 88.00·, 0.39 points behind the first-ranked system).

Bradley, M.M., and Lang, P.J. [34] introduced affective norms for English words (ANEWs), which provide emotional assessments for many English words. The positivity and intensity levels of stimuli impact emotional responses, influencing how we process and perceive emotional information. The author of [35] presented AFINN-96, which consists of 2477 distinct words, with an addition of 15 phrases that were not used. The author simplified the process by focusing only on valence when scoring the words, excluding factors like subjectivity/objectivity, arousal, and dominance, and assigning scores manually. The author of [35] used ANEW as a classification-based fuzzy model as a basis for expressing stochastic with five label terms for classifying tweets into five fuzzy opinion categories, very negative, negative, neutral, positive, and very positive, which allowed for a more nuanced understanding of sentiment in social media data. While ANEW includes many words commonly used in English, it lacks depth due to the evolving nature of language use in online communication and social media posts. ANEW does not account for the use of negations in words, making evaluation challenging as negations do not always reverse the meaning of each word, especially when adverbs are involved, leading to increased complexity. The author of [36] introduced a novel approach for analysing the sentiments from tweets that include a mix of adjectives, adverbs, and verbs to determine the sentiment score, and the actual polarity of the text or online communication is classified using a linear function that incorporates emotion intensity.

Hutto and Gilbert (2014) [37] proposed a valence-aware dictionary and sentiment reasoner (VADER), a lexicon- and rule-based sentiment analysis tool that is specifically attuned to sentiments expressed from online text collected from social media or the Internet. VADER (available online: http://www.nltk.org/_modules/nltk/sentiment/vader.html (accessed on 17 September 2024)) supports the handling of emoticons, idioms, punctuation, negation, emphasis, and contrasts. VADER also considers the impact of words in ALL CAPS to emphasise their meaning. Depending on the original sentiment of the word, the overall polarity is adjusted by 0.733. Additionally, VADER can identify negated sentences and evaluate sentiment shifts brought on by contrastive conjunctions like "but". However, the empirical validation of the VADER is based on multiple independent human judges that incorporate a "goldstandard" sentiment lexicon that is especially attuned to microblog-like contexts.

Çılgın et al. [38] used VADER to analyse and perform opinion mining on Twitter data. In addition to the binary classification system that almost all other Twitter-based sentiment analysis models provided, their model also provided a multi-classification system. It was observed that VADER was an apt selection to analyse large sets of data. This model's limitations were that a small percentage of the actual data were employed, a general lexicon was used to categorise the data, and the data were not trained. Newman and Joyner [39] used VADER to analyse student evaluations of teaching from three sources. The positive/negative valences of the comments were compared, frequently used keywords in comments were identified, and the impact of those comments containing said keywords was determined.

Elbagir et al. [40] used VADER to analyse sentiments in social media posts using individual words and sentences. The text of tweets was preprocessed to remove unwanted characters and words such as punctuation, unicode problems, URLs, emails, currency symbols, and numerals. Jain et al. [41] used natural language processing (NLP) with VADER to predict the sentiment from social media data such as X, Facebook, and Reddit, and the results are interpreted and explained using heatmaps. Using VADER to evaluate sentiments in the Twitter dataset, the author achieved 69.52% accuracy, 64.88% precision, 85.10% recall, and 73.63% F1-score for the tweet, respectively, after normalisation. In their study, Borg et al. [42] utilised a linear SVM as a machine learning classification model and VADER to predict customer feedback sentiment for the Huge Swedish Telecom Corporation by analysing a dataset of 168,010 emails. The author employed the Swedish sentiment Lexicon and VADER for sentiment analysis, achieving an F1 score of 83.4% and a mean AUC of 0.896. Moreover, the author in [42] identified a pattern in email discussions that could potentially predict the emotions of unseen emails.

Hu et al. [43] proposed a method to summarise customer reviews of electronic products from Amazon and C|net.com based on product features. Nevertheless, many incorrectly spelt words are part of the features that frequently appear in social media content, which can make it challenging for automated systems to accurately identify and summarise customer opinions. Additionally, the constant evolution of language and slang used in online reviews adds another layer of complexity to the task of extracting meaningful insights from customer feedback.

Baccianella et al. [44] proposed SentimentNet. 3.0, a lexical resource specifically designed to enable sentiment classification and opinion mining applications. SentimentNet 3.0 is an upgraded version of SentiWordNet 1.0, a lexical resource made publicly available for research purposes licensed to over 300 research groups and used in a wide range of research projects throughout the world [44]. SentiWordNet 1.0 and 3.0 are the outcomes of automatically annotating all WordNet synsets for positivity, negativity, and neutrality.

Moshkin et al. [45] used fuzzy ontology subgraphs based on lexical dictionaries to find morphological features in VKontakte text fragments, such as word, smile, or style. They determined sentiment by analysing features of the subject area, focusing on syntagmatic units rather than individual words for compatibility. The lexical ontology was assessed using SentiWordNet 3.0, which is built on WordNet 3.0. The method was tested using ML algorithms such as the Naive Bayes (NB) classifier, linear regression, and SVM on 420 VKontakte posts and comments. The average accuracy achieved was 78.33%, 65.24%, and 75.25%, respectively. The study revealed that the NB classifier performed better than the other ML algorithms tested.

Sadhasivam et al. [46] retrieved a dataset from an official product review site. The data were cleaned by eliminating unnecessary elements like stop words, verbs, punctuation, and conjunctions. In the referenced study, the author computes the positive and negative probabilities for each word in the dataset, merges these probabilities, and determines the sentiment based on the higher probability. To determine sentiment, each set of data is converted into a more complex format, and then a mathematical operation is used to identify the strongest indicator of sentiment, with the assistance of SentiWordNet. The dataset is trained using NB, SVM, and Ensemble methods with positive and negative labels, resulting in an accuracy of 78.86%. Nevertheless, the accuracy of the predictions fluctuates depending on the number of classifiers combined for the review output. It is also difficult to precisely interpret how users convey their emotions through emoticons in the reviews.

Selecting an appropriate feature subset is crucial in sentiment classification. Tools like LIWC can extract psycholinguistic aspects from texts for analysis [47]. For example, Onan et al. [48] introduced a psycholinguistic approach to sentiment analysis on Twitter. LIWC extracts psycholinguistic features like linguistic processes, psychological aspects, personal concerns, spoken categories, and punctuation from texts. These features are then processed using an ML algorithm. The author in [48] tested the proposed approach on English Twitter messages, including 6218 negative, 4891 positive, and 4252 neutral tweets. NB has the highest predicted performance (77.35% accuracy) when using the linguistic feature set. Incorporating ensembles, the Random Subspace Ensemble of NB achieves a classification accuracy of 89.10%. In the same way, Koutsoumpis et al. [49] used five main personality traits and 52 linguistic categories to find links between things like self-reports, reports from others, life outcomes, and behavioural measures of personality for text-based assessments. The results indicate that text-based personality assessment offers precise and dependable insights into an individual's personality traits. As another example, Chen et al. [50] used the computerised LIWC to quantify students' cognitive, emotional, and social engagement in social annotation. Additionally, the author in [50] explored how students with varying levels of engagement differ in their social annotation behaviours. They used a statistical method to analyse data from 91 undergraduate students and 29 reading materials, successfully identifying two different engagement patterns with a high Cohen's Kappa of 0.78. The two different engagement patterns were labelled as "active" and "passive". Despite a comprehensive examination of student interactions in social annotation, the study did not examine the interactions between students and instructors in social annotation or the contribution of the student–instructor relationships to assist in the overall learning outcomes. Li et al. [51] explored the patterns in how students used annotations and how they responded to them in social annotation activities. They examined how students' performance in behavioural, cognitive, emotional, and social areas changed based on their interactions. They gathered 93 undergraduates who were enrolled in an elective course at a large North American university, and the students were tasked with collaboratively annotating the class readings uploaded to Perusall, a social annotation platform, over 7 weeks. To determine exactly how to effectively classify student behaviours into groups, the researcher in [51] used metaclustering analysis based on the number of annotations and response behaviours. For example, they combined multiple clustering solutions to make them more robust and reliable. We looked at the number of annotation and response behaviours and used the K-means algorithm to find the best number of clusters from 905 data instances. Then, we used LIWC [52], a text mining tool to evaluate the levels of students' cognitive activities, specifically focusing on cognitive insight and cognitive discrepancy.

Word Sense Disambiguation (WSD) is a method used to determine the meaning of a word with multiple senses. Rentoumi et al. [53] introduced the use of WSD to assign polarity based on an n-gram graph for analysing sentiment in text related to figurative phrases. This polarity assignment involves using a tool that adjusts the sentiment of figurative phrases in text or online content based on their context. It uses a range of eight surrounding words to determine similarity and generate a gloss vector (GV) in WSD. GV creates a co-occurrence matrix of words, where each cell in the matrix indicates the number of times the words represented by the row and the column occur together in a WordNet gloss. Each word in a WordNet gloss is depicted as a vector in a multi-dimensional space based on its specific row. For each word in the gloss, a context vector is formed by using the respective row in a matrix that shows how often words appear together. Subsequently, a gloss vector is generated as the average of all these context vectors for each word sense. The similarity is determined by analysing the parts of speech (POS) using a tool called the Stanford POS tagger. The next step involves assessing the sentiment by associating WordNet senses with words in the text, which are classified into positive or negative categories. The author used hidden Markov models (HMMs) to figure out how sentences made the author feel, and the results were confirmed in the Affective Text task of SemEval'07 [54]. The results show that this method effectively assigns sentiment to figurative language, indicating its potential for use in different NLP tasks.

Jayakrishnan et al. [55] used WSD and SVM classification on non-English text to find emotions or people's feelings. They obtained a 91.8% per cent success rate, but this rate can go up if semantic and syntactic features are added. Hogenboom et al. [56] used historical stock prices of NASDAQ-100 companies and news articles from Dow Jones Newswires to test graph-based WSD as a sentiment predictor of stock price. They obtained a 53.3% success rate. However, the system cannot evaluate more complex trading strategies, and future research is expected to incorporate a more explicit notion of human sentiment with respect to news articles. Table 1 provides an overview of various lexicon-based methods for sentiment analysis. It shows the reference, the wording size, the attributes used to represent text, an acronym for the proposed model, the source or dataset, and the evaluation metric such as precision (PR), recall (RC), F1-score (F1), and accuracy (ACC) or result obtained for comparison.

Table 1. A comparison of lexicon-based approaches.

Reference	Features	Lexicon	Dataset	Source	Metrics	Results
[32]	Word n-gram	MaxDiff	Twitter and SMS	SemEval-2013 Available online: https://paperswithcode.com/dataset/semeval-2013-task-2 (accessed on 1 October 2024)	F1	69%
[43]	Misspelled words	None	Customer Reviews	HuAndLiu Available online: https://www.cs.uic.edu/~liub/FBS/sentiment-analysis.html (accessed on 1 October 2024)	Precision	85%

Table 1. *Cont.*

Reference	Features	Lexicon	Dataset	Source	Metrics	Results
[44]	Cognitive synonyms	SentiWordNet	SentiWordNet Corpus	SentiWordNet Available online: https://github.com/aesuli/SentiWordNet (accessed on 1 October 2024)	Coverage	90%
[10]	Cognitive and emotional words	LIWC	Psycholinguistic Texts	LIWC Available online: https://www.liwc.net/ (accessed on 1 October 2024)	F1	82%
[34]	Emotional words	ANEW	Affective Word List	ANEW Available online: https://github.com/dwzhou/SentimentAnalysis (accessed on: 1 October 2024)	Norms Evaluation	High
[35]	Emotional words	AFINN-96	Word List Evaluation	AFINN-96 https://www.npmjs.com/package/afinn-96 (accessed on 1 October 2024)	Precision	85%
[53]	Word, sentence and n-gram	WSD, HMMs	Blogs, Editorials	SemEval'07 Available online: https://web.eecs.umich.edu/~mihalcea/affectivetext/#datasets (accessed on 1 October 2024)	F1	70%
[56]	Word Sense Disambiguation	WordNet	Stock prices	WordNet https://wordnet.princeton.edu (accessed on 1 October 2024)	Recall	76.5%
[54]	Emotion Classification	WordNet, SentiWordNet	News Headlines, Blogs	SemEval-2007 Available online: https://web.eecs.umich.edu/~mihalcea/affectivetext/#datasets (accessed on 1 October 202)	F1	64%
[52]	Psychometric Analysis	LIWC-22	Psychometric Data	LIWC Available online: https://www.liwc.net/ (accessed on 1 October 2024)	Cronbach's Alpha	0.85
[49]	Personality Traits	LIWC-22	Text-Based Data	LIWC Available online: https://www.liwc.net/ (accessed on 1 October 2024)	Meta-analysis	
[48]	Sentiment Analysis	None	Twitter Data	Twitter Dataset https://developer.twitter.com/en/docs/twitter-api (accessed on 1 October 2024)	F1	82%
[47]	Linguistic Inquiry	LIWC, Korean LIWC	Korean Linguistic Data	Korean-English Parallel Corpus Available online: https://aihub.or.kr/aidata/7974 (accessed on 1 October 2024)	Accuracy	91%
[57]	Unigram, bigram, trigram	Count frequency	Amazon Reviews	Amazon Dataset Available online: https://registry.opendata.aws/amazon-reviews/ (accessed on 1 October 2024)	Accuracy, Speed	3–5× faster

2.3. Deep Learning-Based Approaches

Recently, it has been observed that the number of people actively involved in social media is rapidly increasing [58]. People are expressing their feelings in the form of reviews, comments, posts, and statuses on various topics [59]. As a result of this tremendous amount of data generated on the Internet, which can be analysed for further research, traditional methods using predefined rules or ML algorithms are not enough to manage the large volume of data available [60,61]. Hence, the adoption of deep learning models in NLP is essential to uncovering valuable insights from these unorganised data. These models have delivered positive outcomes in analysing emotions, condensing text, and translating languages, becoming essential instruments for comprehending and utilising the large volume of text online. Applications of deep learning models have been successful in identifying patterns and trends in text data, providing valuable information for businesses to make informed decisions. Additionally, these models have the potential to revolutionise customer service by automating responses and improving the overall user experience. Furthermore, these models also help businesses personalise their marketing strategies and target specific customer segments more effectively. However, in South Africa, consumer reviews on social media platforms like Hellopeter provide essential insights into the performance and sentiment of financial institutions. Dealing with finance-related documents has presented challenges due to the traditional methods used for sentiment analysis.

The recent advancements in LLMs and NLP have led to better ways of conducting sentiment analysis tasks. The development of LLMs and NLP has led to changes in model structures, pre-training techniques, and the integration of RAG technologies. RAG is a process that obtains relevant information from outside data in real time to improve context and relevance generation. The extra information is then fine-tuned into the model itself or downstream tasks [62]. Additionally, RAG technologies enable the models to interact with structured knowledge sources, allowing them to access and answer questions beyond the scope of the provided data. For instance, Liu et al. [63] proposed a method for text classification wherein the author highlights the importance of models that are easy to understand and interpret, in addition to focusing on performance. Gao et al. [64] proposed a model in which information is retrieved before generating responses, enabling the generated responses to be more precise and relevant in tasks that require a lot of knowledge. Fan et al. (2024) [65] and Hu et al. [66] studied RAG for NLP tasks but mainly financial sentiment analysis. Lewis et al. [67] suggested a retrieval component that looks through huge collections of documents for relevant ones and feeds them into the generative model. This makes the answers more accurate and relevant to the situation. Zhang et al. [68] address how hard it is to use real-time, context-relevant data in financial sentiment analysis and how RAG models help solve these challenges while making outputs easier to understand and more reliable in a world where finances are always changing. These studies highlight the importance of incorporating information retrieval into NLP tasks to improve the quality of generated responses. By simply combining both modelling and retrieval techniques, researchers have been able to achieve more accurate and relevant results in various applications, such as financial sentiment analysis. This has led to significant advancements in the field of NLP. This has led to significant advancements in the field of NLP, ultimately improving decision-making processes and providing valuable insights for businesses.

However, RAG models have their drawbacks. These include efficiently exploring vast corpora while remaining fast and accurate. Furthermore, the challenge of incorporating the returned knowledge into the generating process in a cohesive and contextually meaningful manner persists. Future studies could include improving the retrieval function using real-time data sources and experimenting with merging RAG with reinforcement learning or meta-learning methodologies. For example, Shivaprasad et al. [69] emphasised the necessity of conducting thorough sentiment analysis on product reviews to gauge customer sentiment, which subsequently affects the financial markets. In this context, the use of LLMs and RAG improves the precision of sentiment prediction to support better strategic decisions. Zhao et al. [70] proposed a generalised pre-training framework to enhance the existing RAG model, showcasing the effectiveness of LLMs for sentiment analysis tasks. The researchers amalgamated conversational fine-tuning and RAG approaches, and this new development has made LLM performance much better, especially in sentiment analysis in the financial domain. Vulic et al. [71] presented a method that adjusts LLMs for dialogue systems, showcasing their ability to handle complex and context-specific conversations. Likewise, Alghisi et al. [72] proposed evaluation methods for adapting LLMs to dialogue-based assignments, highlighting the advantages and disadvantages of each approach. These methods are especially useful in financial sentiment analysis because of the precise natural linguistics of human expression in text posted on social media, and the approach allows one to model the human perspective in a natural language so as to capture the full context or nuances of the language used and offer decision-makers insights into customer behaviour and market trends.

Furthermore, there has been widespread use of LLMs, such as ChatGPT, sparking debates about their capabilities and limitations in a variety of industries. For example, in the field of education, Fˋʼutterer et al. [73] demonstrated responses from all over the world to ChatGPT, which revealed a wide range of sentiments, from supportive to alarming. These studies highlight the significance of considering context and other culturally specific factors in the evaluation of AI. Certainly, LLMs have received praise for their achievements across various domains. Duan et al. [74] introduced an innovative hybrid neural network model for analysing financial text data. This model surpasses previous approaches in sentiment analysis by enhancing topic extraction and pre-training techniques. In general, LLMs have performed well in a number of NLP tasks, such as answering questions and aspect sentiment classification (ASC). For example, Ling et al. [75] developed a retrieval-augmented method that makes semantic representations more descriptive, which makes it easier to classify sentiment across different aspects. The approach also has the capability to handle multidomain sentiment classification, which focuses on transferring information from one domain to the next. The models are first trained in the source domain; the knowledge is then transferred and explored in another domain. For example, Chen et al. [76] proposed a weakly supervised multimodal deep learning (WS-MDL) model to predict multimodal sentiments for tweets. The model uses CNN and DynamicCNN (DCNN) to calculate multimodal prediction scores and sentiment consistency scores. Due to the enormous data available on social media in different forms like videos, audio, and photos for expressing sentiment on social media platforms, the conventional approach for text-based sentiment analysis progressed into compound models of multimodal sentiment analysis. Hence, capturing the sentiment perspectives expressed in different modalities became a crucial approach [77]. The number of data available, the quantity of hidden units (nodes) required to solve the problem, and other factors still impact the choice of a particular deep learning model in the field of ASC. Table 2 shows a full comparison of existing literature that was discussed based on different factors, including embedding representation, dataset, deep learning model, and performance metrics. For example, RC, ACC, and F1 are common metrics for sentiment analysis tasks because they show how well models do at analysing and classifying sentiment in textual data.

Table 2. Comparison of related studies based on common characteristics.

Reference	Embedding	Tasks	Method	Metric	Dataset
[78]	Various AI Techniques	Multiple AI Models	Review	Qualitative Analysis	N/A
[79]	LLMs, Transformer	GPT-4, ChatGPT	NLP Tasks	Accuracy, F1 Score	Open-source datasets
[80]	Medical Data Analysis	Various ML Models	Comparative Study	Sensitivity, Specificity	Medical datasets
[81]	LLMs, Transfer Learning	GPT-3, BERT	Transfer Learning	Accuracy, Precision, Recall	Financial datasets
[82]	LoRA Fine-tuning	Llama-2	Fine-tuning	Accuracy, F1 Score	Custom datasets
[83]	LLMs, Sentiment Analysis	GPT-4, BERT	Sentiment Analysis	Accuracy, Precision, Recall	Social media datasets
[84]	ChatGPT, Sentiment Analysis	ChatGPT	Sentiment Analysis	Accuracy, F1 Score	Customer reviews
[8]	LLMs, Reinforcement Learning	GPT-3, GPT-4	RLHF	Accuracy, F1 Score	Various NLP benchmarks
[85]	LLMs, Multimodal Learning	GPT-4, Multimodal Models	Multimodal Tasks	Accuracy, F1 Score	Multimodal datasets
[86]	LLMs, Text Generation	GPT-3, GPT-4	Text Generation	BLEU, ROUGE	Text corpora
[68]	Retrieval-Augmented LLMs	GPT-4, LLaMA	Retrieval-Augmented	Accuracy, F1 Score	Financial news datasets
[79]	LLMs, Ethics	GPT-3, GPT-4	Ethical Analysis	Qualitative Analysis	N/A
[87]	LLMs, Robustness	GPT-3, GPT-4	Adversarial Testing	Robustness Metrics	Adversarial datasets
[88]	LLMs, Translation	GPT-4, Translation Models	Machine Translation	BLEU, METEOR	Multilingual datasets
[89]	LLMs, Evaluation Methods	GPT-3, GPT-4	Comprehensive Survey	Various Metrics	Various datasets

Table 2. *Cont.*

Reference	Embedding	Tasks	Method	Metric	Dataset
[90]	LLMs, Privacy	GPT-4, BERT	Privacy-Preserving	Accuracy, F1 Score	Sensitive datasets
[91]	LLMs, Healthcare	GPT-4, MedGPT	Medical Diagnosis	Accuracy, Recall	Medical records
[92]	User Behavior, AI	User Model, AI Systems	User Study	User Satisfaction, Accuracy	User data
[93]	Sentiment Analysis	SVM, Neural Networks	Comparative Study	Accuracy, Precision, Recall	Movie reviews
[94]	Edge Computing, AI	Edge AI Models	Edge Computing	Latency, Accuracy	IoT datasets
Proposed Method	**LLMs, Conversational Fine-Tuning, RAG**	**Llama-2, Llama-3, GPT-3.5 Turbo, GPT-4o-Mini**	**Sentiment Analysis**	**Accuracy, F1-measure, Ragas, vendi score, polarity,granularity**	**Hellopeter reviews**

In this study, we build on ARLMs, PTLMs, and RAG to establish a new framework called LFEAR. This framework promises to not merely detect but also to classify the meanings of various sentences found in textual forms of social media and in the financial comments made about products and services of financial institutions (e.g., Hellopeter). The LFEAR model retrieves real-time financial information on a dynamic basis by integrating RAG. This satisfies the need for a constant adaptation to not only emerging trends but also to the types of new language that these trends provoke [95]. LFEAR not only makes a more computationally efficient model but also heightens the intensity and granularity of the sentiment classifications. This allows for a much clearer representation of the actual sentiments expressed in the data. It is necessary, especially in the finance industry, to have a precise model that can influence decision-making based on the surrounding environment and the interaction of the customers with the product and service offer [96]. LFEAR leverages a continuous learning framework to incorporate financial information in a structured manner and seeks to build a model that is adaptive, accurate, and precise. LFEAR's goal is to obliterate the limits we have identified in the field of sentiment analysis and offer a model that is comprehensive, adaptive, and accurate enough to serve as a sentiment analysis engine in the financial domain [97].

3. Proposed Method

The proposed LFEAR architecture, illustrated in Figure 1, is an open-source model specifically developed for sentiment analysis with an emphasis on domain adaptation, for instance, for the sentiment analysis of financial reviews from Hellopeter. All these features are incorporated in a four-tiered system where RAG, conversational fine-tuning, and continuous learning are employed as cutting-edge technologies. The input layer handles a variety of data formats, while the preprocessing layer prepares data through cleaning, embedding, and feature selection. The adaptable behavioural layer, which is the core of the proposed architecture, combines RAG and fine-tuned LLMs such as Llama-2, Llama-3, GPT-3.5 Turbo, and GPT-4o Mini to improve contextual relevance. Finally, the model inference interface generates sentiment analysis findings and strategic insights, making this architecture resilient and adaptable to a variety of datasets and user requirements.

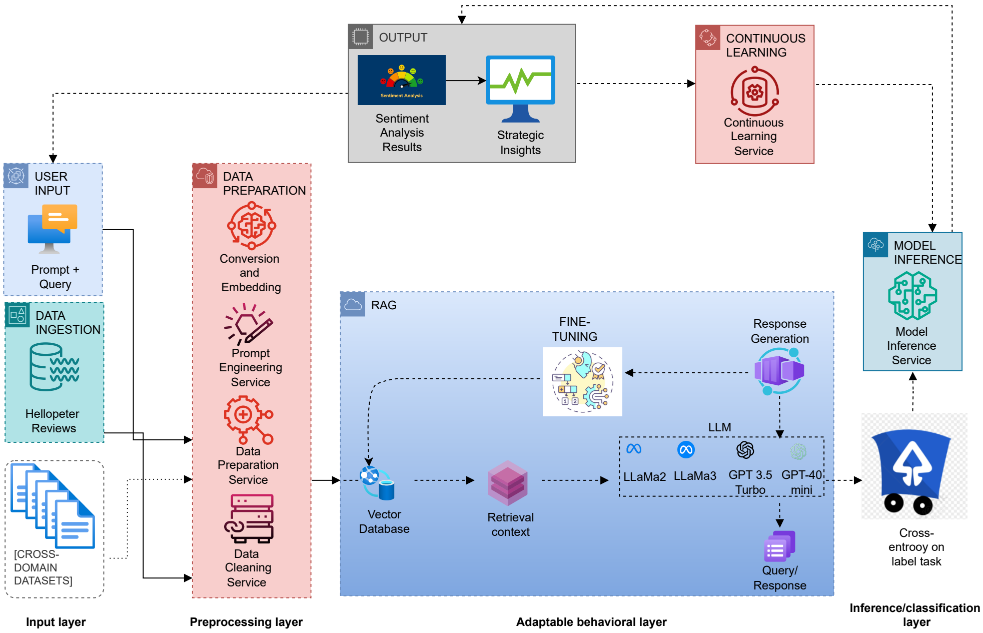


Figure 1. Proposed LFEAR model for sentiment analysis.

To thoroughly verify the proposed LFEAR model, we derive a fundamental mathematical equation that represents the model's essential elements: data ingestion, retrieval-augmented generation, fine-tuning, inference, and ensemble voting. This equation unifies the architectural parts of the model, showing how it generates and combines predictions.

Let:

- $\mathcal{D}_{\text{input}}$: Initial input data, including prompts and queries, ingested from both Hellopeter and cross-domain datasets.
- V: The vector database containing embeddings generated from the input data.
- $\text{RAG}(\cdot)$: Function that produces a context-aware representation by retrieving relevant embeddings from V.
- $\mathcal{M}_i$: Each individual model within the ensemble, where $i = 1, 2, \ldots, N$ (models such as Llama-2, Llama-3, GPT-3.5 Turbo, etc.).
- w_i: Weight for model $\mathcal{M}_i$, optimised based on its cross-domain performance.
- $f_i(r)$: Sentiment output (positive or negative) from model $\mathcal{M}_i$ for review r.
- $S(r)$: The aggregated sentiment score derived from ensemble predictions.

The overall sentiment prediction for the LFEAR model is then defined as

$$S(r) = \arg \max_{s \in \{\text{positive}, \text{negative}\}} \sum_{i=1}^{N} w_i \cdot \delta\big(f_i(\text{RAG}(\mathcal{D}_{\text{input}})) = s\big) \tag{1}$$

where

- $S(r)$ is the final sentiment label for a given review r.
- $\delta(\cdot)$ is an indicator function that returns 1 if the condition inside is true (i.e., if the model $\mathcal{M}_i$ predicts sentiment s after being fine-tuned on context from RAG), and 0 otherwise.
- The RAG component $\text{RAG}(\mathcal{D}_{\text{input}})$ serves as the contextualised input for each model in the ensemble, enhancing relevance and accuracy.

This equation combines the output from each model in the ensemble, weighted by its relevance and accuracy, to produce a robust final sentiment classification $S(r)$.

3.1. Input Layer

The input layer is also the first layer of the proposed architecture, which deals with various inputs from the users and domain datasets. It has a very important function to perform in starting the sentiment analysis process by incorporating user prompts, queries, and other data sets. The layer effectively integrates real-time customer feedback from other sites, including Hellopeter, with cross-domain data sources such as financial news, tweets of politicians, and restaurant reviews for enhanced analysis. This makes the integration of the multiple modes very effective in making the model very general and effective for use in different sentiment analysis applications.

3.1.1. Data Ingestion

The data ingestion process brings in user feedback mainly from the Hellopeter platform, where customers in South Africa leave reviews of their financial institutions. These reviews span a number of banks, such as Standard Bank, Nedbank, Absa, Capitec, and First National Bank. The model takes in the reviews as data from API endpoints in real time, meaning the data are in sync with current sentiment trends. This means that the model is working with a near-real-time reflection of what customers think about these institutions. Key aspects of maintaining the model's data quality are discussed in Section 4.1.

3.1.2. Cross-Domain Datasets

We exclusively utilise cross-domain datasets from non-financial sectors, like hospitality and media, for our model's generalisability evaluation. This operation assesses the model's skill at sentiment labelling across different, previously unencountered situations. Orbital checks ensure the model's fitness when applied to diverse and novel contexts, following the rules of domain adaptation and out-of-domain generalisation [98]. We cross-test the model on datasets from a range of industries to ensure the model is not overfitting to the finance sector and is instead resilient and adaptable to the type of diverse sentiment expression one is likely to encounter across any industry [99].

3.1.3. User Input

The user input component allows people to perform real-time sentiment analysis. Users submit natural language prompts or queries to achieve this. We enhance the contextual relevance of these inputs by first classifying them as either positive or negative sentiments. However, some queries consist of conflicting intents or ambiguous sentiments, which prompts the model to use a newly implemented intent-conflict-detection mechanism to better handle these types of inputs. Instructions that contradict each other and sentiment inconsistencies in the prompts that are flagged by this mechanism make the LLM easier to understand overall [100]. First, our model uses a default mechanism to determine the conflict type and set the sentiment type. This has worked to make our model more robust in a variety of ways. Consistency in decision-making across different input types is a big step toward a systematic mechanism for achieving accurate sentiment predictions across a range of inputs, as shown in Algorithm 1. Two primary input types are distinguished:

Prompt: A prompt is a pre-defined instruction set that standardises how the model interprets and classifies sentiment. Defining sentiment polarity, contextual boundaries, and focus areas e.g., customer satisfaction, prompts limit ambiguity and creates a framework for consistent, context-aware sentiment analysis.

Query: A query represents the specific input submitted by the user for sentiment classification. Each query operates within the framework set by the prompt and is enhanced by RAG-derived context, enabling the model to capture subtle sentiment cues. Examples include

- Query: "Analyse the following customer feedback as positive or negative: The service was fast and good, but the charges are very expensive".
- Query with Context: "The following are some similar reviews: The staff were helpful but the time taken to serve was long. Please classify whether the review is positive or negative".

Algorithm 1 User Input Processing for Sentiment Analysis with Intent Conflict Detection

```
Require: User input $Q$ (query) in natural language
Require: Pre-defined prompt $P$ for sentiment classification context
Ensure: Sentiment classification output (positive or negative)
Initialise the prompt $P$ with defined parameters (e.g., polarity, context boundaries, focus areas)
Define $RAG\_context \leftarrow$ Retrieve relevant context from previously collected sentiment data
Step 1: Query Structuring and Preprocessing
Structure input $Q$ based on the established prompt $P$ for sentiment classification
Clean and normalise $Q$ (e.g., remove unnecessary characters, standardise format)
Step 2: Intent Conflict Detection and Resolution
Initialise conflict flag $conflict\_detected \leftarrow$ False
if $Q$ contains contradictory instructions or sentiment cues then
    Set $conflict\_detected \leftarrow$ True
    Resolve Conflict:
    Apply predefined rules for intent conflict resolution or prompt user for clarification
    if clarification received from user then
        Update $Q$ based on resolved intent
    else
        Proceed with modified query resolution rules (e.g., prioritise primary sentiment)
Step 3: Sentiment Classification with Contextual Augmentation
Formulate final input $Q_{final} \leftarrow Q + RAG\_context$
Apply model to $Q_{final}$ for sentiment classification based on $P$ parameters
Obtain sentiment output $S \in \{\text{positive, negative}\}$
Step 4: Output Generation and Response
Generate final sentiment output $S$ with relevant insights
if $conflict\_detected$ then
    Append conflict resolution note to output for transparency
return $S$
```

3.2. Preprocessing Layer

The preprocessing layer initiates the sentiment analysis process by transforming raw data into a refined format suitable for embedding and model prediction. This layer includes three essential components: data cleaning, data preparation, and conversion and embedding.

3.2.1. Data Cleaning

The first step in preprocessing is data cleaning, which removes unnecessary elements from the raw text to create a structured input. This process includes stripping HTML tags, removing special characters, eliminating stopwords, and clearing unnecessary whitespace. Additionally, all text is converted to lowercase, and words are reduced to their root forms through lemmatisation. These steps are critical in reducing background noise, allowing the model to focus on meaningful linguistic patterns for sentiment classification. The function representing this cleaning procedure is shown below:

$$\text{Cleaned Text} = \text{Lemmatise}(\text{RemoveStopwords}(\text{Normalise}(\text{Original Text}))) \tag{2}$$

Each step in this cleaning function transforms the raw text into a structured format, making it suitable for further analysis and ultimately enhancing the accuracy of model predictions.

3.2.2. Data Preparation

Data preparation is the process that enhances models. It ensures the cleanliness, consistency, and anonymisation of the input data. The process of cleaning up the data involves eliminating any unnecessary or potentially harmful elements from the raw data prior to their integration into the algorithm. It also means making the necessary bridged connection across different datasets. A clean, consistent, and well-anonymised dataset is also standardisation-ready. Additional steps include contraction expansion, e.g., *"can't"* to *"cannot"*, and accent removal for improved readability. SpaCy is used for tokenisation and lemmatisation, reducing words to their root forms, which supports consistent analysis and strengthens the reliability of sentiment classification [101,102]. Through these preprocessing steps, data preparation ensures uniform, relevant input, promoting higher model accuracy and robustness.

3.2.3. Prompt Engineering

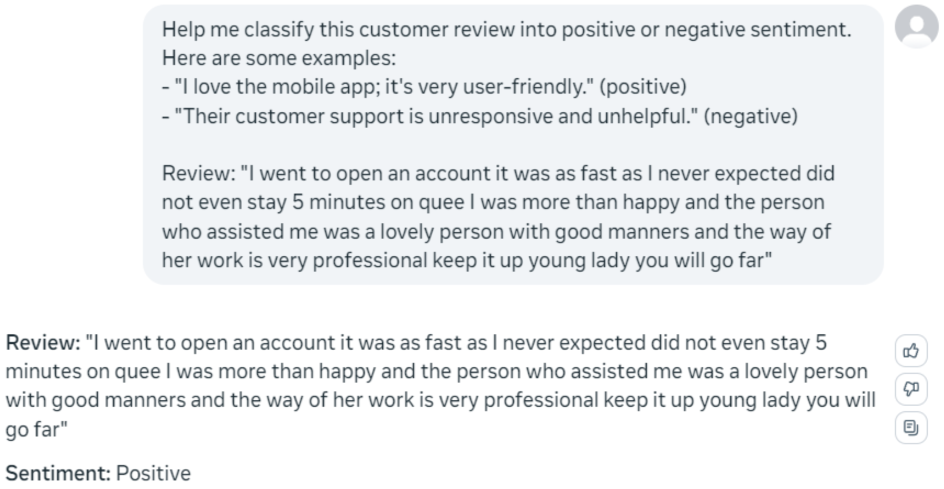
Prompt engineering is one of the important aspects in driving the model to the relevant sentiment classification, particularly in domains such as financial product reviews. The design of prompts uses state-of-the-art methods like few-shot learning, chain-of-thought reasoning, and ReAct (reason and act). These strategies are employed in a very strategic way to increase the model's capacity to understand sentiment in detailed reviews. The detailed process of our prompt engineering strategy is outlined in Algorithm 2.

Algorithm 2 LFEAR Prompt Engineering Techniques

```
Require: Input text $T$ (customer review in natural language)
Require: Pre-defined prompts for Few-shot Learning, Chain-of-Thought Reasoning, and
    ReAct techniques
Ensure: Sentiment classification output $S \in \{\text{positive, negative}\}$
Step 1: Apply Few-shot Learning
Construct few-shot learning prompt $P_{fewshot}$ with labeled example reviews
Generate initial classification prediction $S_{fewshot}$ using $P_{fewshot}$
if $S_{fewshot}$ indicates ambiguity or mixed sentiment then
    Proceed to Chain-of-Thought Reasoning for further clarification
else
    return $S_{fewshot}$ as final sentiment classification
Step 2: Apply Chain-of-Thought Reasoning
Construct chain-of-thought prompt $P_{CoT}$ with guiding criteria (e.g., customer service,
product features, satisfaction level)
Generate refined classification prediction $S_{CoT}$ based on $P_{CoT}$
if $S_{CoT}$ is inconclusive or requires real-time adaptation then
    Proceed to ReAct prompt for adaptive reasoning
else
    return $S_{CoT}$ as final sentiment classification
Step 3: Apply ReAct (Reason and Act)
Construct ReAct prompt $P_{ReAct}$, focusing on aspects such as reasoning and decision-
making
Generate final adaptive sentiment prediction $S_{ReAct}$ based on $P_{ReAct}$ and review of $T$
if $S_{ReAct}$ is confirmed with high confidence then
    return $S_{ReAct}$ as final sentiment classification
else
    Flag $T$ for manual review if sentiment classification remains unclear
```

Few-shot Learning: This technique is based on the fact that the model is trained on a subset of well-selected examples within the prompt and can generalise and classify new inputs correctly even with a small number of training samples [103], as shown in Figure 2. An example prompt used is as follows:

> Help me classify this customer review into positive or negative sentiment. Here are some examples:
>
> - "I love the mobile app; it's very user-friendly". (positive)
> - "Their customer support is unresponsive and unhelpful". (negative)



Help me classify this customer review into positive or negative sentiment.
Here are some examples:
- "I love the mobile app; it's very user-friendly." (positive)
- "Their customer support is unresponsive and unhelpful." (negative)

Review: "I went to open an account it was as fast as I never expected did not even stay 5 minutes on quee I was more than happy and the person who assisted me was a lovely person with good manners and the way of her work is very professional keep it up young lady you will go far"

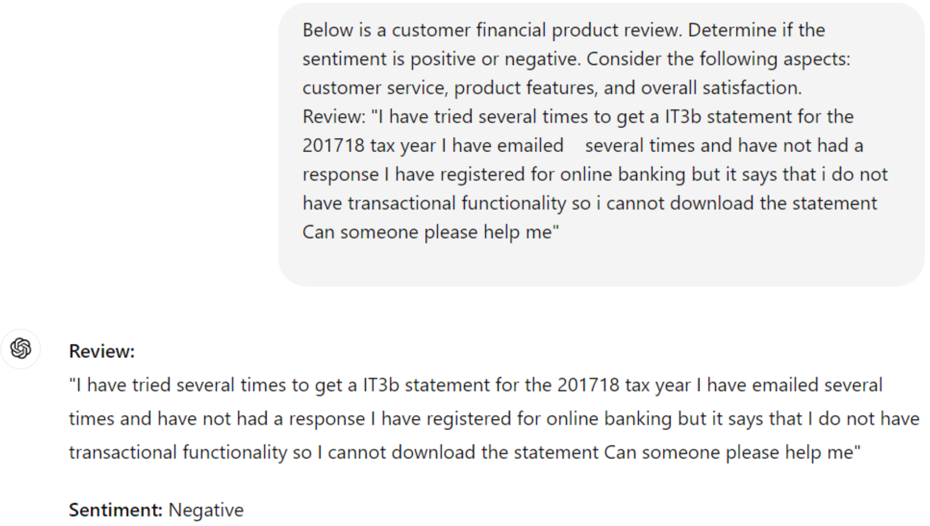
Review: "I went to open an account it was as fast as I never expected did not even stay 5 minutes on quee I was more than happy and the person who assisted me was a lovely person with good manners and the way of her work is very professional keep it up young lady you will go far"

Sentiment: Positive

Figure 2. Few-shot learning with the Meta-Llama-3 model.

Chain-of-Thought Reasoning: This enhances the model's interpretability through a logical flow when classifying sentiment [104], as shown in Figure 3. This allows the model to consider elements such as service delivery, product attributes, and satisfaction levels before arriving at a classification. The prompt used is as follows:

> Below is a customer financial product review. Determine if the sentiment is positive or negative. Consider the following aspects: customer service, product features, and overall satisfaction.



Below is a customer financial product review. Determine if the sentiment is positive or negative. Consider the following aspects: customer service, product features, and overall satisfaction.
Review: "I have tried several times to get a IT3b statement for the 201718 tax year I have emailed several times and have not had a response I have registered for online banking but it says that i do not have transactional functionality so i cannot download the statement Can someone please help me"

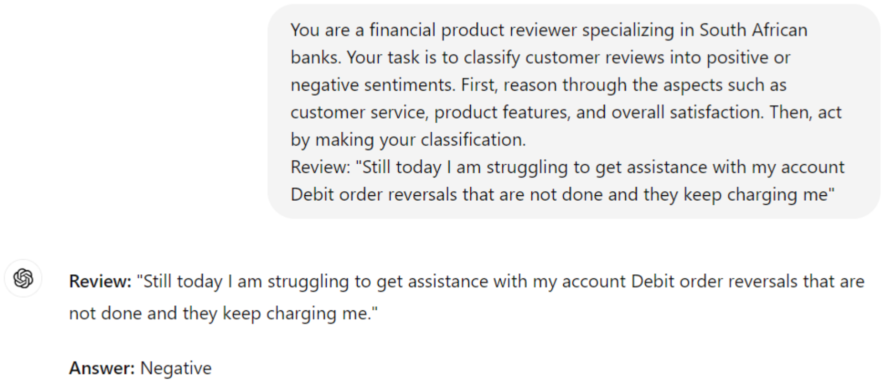
Review:
"I have tried several times to get a IT3b statement for the 201718 tax year I have emailed several times and have not had a response I have registered for online banking but it says that I do not have transactional functionality so I cannot download the statement Can someone please help me"

Sentiment: Negative

Figure 3. Chain-of-thought reasoning with the OpenAI GPT-4o Mini model.

ReAct (Reason and Act): ReAct combines reasoning and adaptation, where the model adapts its output in real-time as it processes the query[105], as shown in Figure 4. The prompt used is as follows:

> You are a financial product reviewer specialising in South African banks. Your task is to classify customer reviews into positive or negative sentiments. First, reason through aspects such as customer service, product features, and overall satisfaction. Then, act by making your classification.



You are a financial product reviewer specializing in South African banks. Your task is to classify customer reviews into positive or negative sentiments. First, reason through the aspects such as customer service, product features, and overall satisfaction. Then, act by making your classification.
Review: "Still today I am struggling to get assistance with my account Debit order reversals that are not done and they keep charging me"

Review: "Still today I am struggling to get assistance with my account Debit order reversals that are not done and they keep charging me."

Answer: Negative

Figure 4. Reason and Act with OpenAI GPT-3.5 Turbo Model.

Through the combination of these techniques, the prompt engineering guarantees that the model can capture sentiment and at the same time can learn the context under which sentiments are expressed so as to enhance the overall sentiment analysis. This is especially the case in financial reviews where the use of subtle expressions and even mixed feelings are often used.

3.2.4. Conversion and Embedding

After data cleaning and preprocessing, the data are transformed into numerical representations that are called embeddings. These embeddings are obtained using the Sentence-Transformer model known as *all-MiniLM-L6-v2* [106]. The model aims at capturing the semantic meaning of the text so that vectors that represent similar sentiments are close in this high-dimensional space. The values of the embedding dimension for each document are 384. The generated embeddings are then saved in a Facebook AI Similarity Search (FAISS) index [107] for efficient search.

The embeddings are derived in the following manner. Here, let $X = \{x_1, x_2, \ldots, x_n\}$ be the cleaned and preprocessed set of reviews, where each x_1 is a textual input. The embedding model ϕ transforms each x_i into a fixed-size vector $\mathbf{v}_i \in \mathbb{R}^{384}$:

$$\mathbf{v}_i = \phi(x_i), \tag{3}$$

where $\mathbf{v}_i$ is the embedding vector for review x_i.

The FAISS index is used to index the embedding vectors. Given a query vector $\mathbf{q}$, the retrieval process finds the k nearest neighbours by minimising the Euclidean distance d between $\mathbf{q}$ and all stored vectors $\mathbf{v}_i$:

$$d(\mathbf{q}, \mathbf{v}_i) = \|\mathbf{q} - \mathbf{v}_i\|_2. \tag{4}$$

The algorithm returns the k documents with the smallest distance d, indicating the text with the most semantically similar reviews. This embedding and retrieval approach allows the sentiment analysis model to be both scalable and capable of delivering high-quality results by adding contextual and semantic information to the input data.

Generation and document retrieval are critical processes in gathering the most relevant context for sentiment analysis. Pre-trained embedding models and FAISS [107] are used to efficiently index documents. The embedding generation stage converts both the query and the documents into dense vectors, obtaining the semantic information required to perform similarity matching. The document retrieval procedure then uses FAISS to discover the documents most comparable to the vector representations by performing a nearest-neighbour search on the top k results. Algorithm 3 provides a step-by-step overview of the entire process, from embedding generation to retrieval.

Algorithm 3 Conversion and Embedding Generation

```
Input: Query $q$, Document Set $X = \{x_1, x_2, \ldots, x_n\}$, Number of neighbours $k$
Output: Top $k$ most relevant documents $D_{\text{top}_k}$
Load pre-trained embedding model $\phi$
Initialise FAISS index $I$
Initialise index-to-document map $M$
for each document $x_i$ in $X$ do
    $v_i \leftarrow \phi(x_i)$    ▷ Generate embedding for document $x_i$
    Add $v_i$ to FAISS index $I$
    $M[i] \leftarrow x_i$    ▷ Store document reference for later retrieval
$q_{\text{embedding}} \leftarrow \phi(q)$    ▷ Generate embedding for the query
$[d_{\text{top}_k}, \text{indices}] \leftarrow I.\text{search}(q_{\text{embedding}}, k)$    ▷ Retrieve top $k$ nearest neighbours
$D_{\text{top}_k} \leftarrow \{M[\text{indices}[j]] \mid j = 1, \ldots, k\}$    ▷ Map retrieved indices back to original documents
return $D_{\text{top}_k}$
```

3.3. Adaptable Behavioural Layer

The adaptable behavioural layer is the core of our model that combines RAG and the fine-tuning of LLMs, including Llama-2, Llama-3, GPT-3. 5 Turbo, and GPT-4o Mini. This layer continuously updates the model's knowledge to domain-specific scenarios, especially when handling product reviews concerning South African financial institutions. The RAG component greatly improves the model's performance by providing the inference process with more context and other relevant information obtained from other sources. To enhance the LLMs' performance, fine-tuning techniques are applied to fine-tune the LLMs for financial reviews, which contain distinctive language and specialised terms, thus enhancing sentiment classification efficiency and context relevance, as shown in Algorithm 4.

3.3.1. Model Selection

The reason for selecting certain models revolved around their ability to process language, retain context, and adapt to domain-specific scenarios. Among the chosen models, Llama-2 and Llama-3 stand out for their efficiency in managing the complex and emergent language patterns of reviews for financial products [108]. The team also selected GPT-3.5 Turbo and its successor, GPT-4o Mini, because they are known to generate contextually aware responses, even when the language being processed is complex [109]. Each model was chosen based on its architecture and pre-training corpus, which together make a strong base for much more adaptation through fine-tuning [110]. Llama-2 and Llama-3, for instance, handle not just ordinary language but also the intricate structures that certain kinds of specialised, domain-specific language take with seeming ease [111]. Their proficiency makes them ideal for tasks where a simple understanding of ordinary sentiment and context will not suffice, as the language in question is not at all simple. Indeed, any model we employ must proficiently handle both ordinary tasks and unconventional ones. The fine-tuning of each model is discussed in detail in Section 3.3.5.

3.3.2. Vector Database Implementation

The vector database is implemented with the help of FAISS [107] to search for similar vectors in high-dimensional space. Word vectors are created using the "all-MiniLM-L6-v2" model [106], a fast and efficient transformer model, which is stored in the vector index for efficient search. The FAISS index is specifically designed for large-scale document search, so the system can easily maintain high accuracy and low latency in document retrieval irrespective of the amount of data being added. This indexing and retrieval mechanism guarantees that the model can use the most relevant information during inference and thereby improve the overall decision-making and flexibility in dynamic scenarios.

We use a single vector database for context retrieval and its potential impact on scalability as dataset volume and diversity grow. To mitigate this, future implementations could employ distributed or hierarchical indexing techniques, which divide data across multiple indices or layers based on relevance or category. These techniques allow scalable expansion by segmenting data retrieval processes, which improves search efficiency without compromising retrieval accuracy. By adopting these advanced indexing methods, the system can handle larger, more heterogeneous datasets while maintaining low latency and high accuracy.

Algorithm 4 Adaptable Behavioural Layer

```
Require: Query $q$, Document corpus $D$, Vector database $V$, Fine-tuned language models
    $\{\text{Llama-2}, \text{Llama-3}, \text{GPT-3.5 Turbo}, \text{GPT-4o Mini}\}$
Ensure: Final sentiment classification output $S$
Step 1: Query Embedding Creation
Encode the input query $q$ into an embedding $\mathbf{q}$ using the pre-trained embedding model
    (e.g., 'all-MiniLM-L6-v2')
Step 2: Vector Database Retrieval (FAISS)
Use FAISS to retrieve the top-$k$ relevant document embeddings $\{d_1, d_2, \ldots, d_k\}$ from $V$
    based on the Euclidean distance between $\mathbf{q}$ and each document embedding $\mathbf{v}_i$ in $V$:
        $d(\mathbf{q}, \mathbf{v}_i) = \|\mathbf{q} - \mathbf{v}_i\|_2$        (5)
Step 3: RAG
for each retrieved document $d_i$ in $\{d_1, d_2, \ldots, d_k\}$ do
    Concatenate $q$ with $d_i$ to form a combined input
    Pass the combined input to each fine-tuned language model to generate an intermediate response $r_i$ with probability:
        $P(r \mid q) = \sum_{d \in D_k} P(r \mid q, d) P(d \mid q)$        (6)
Step 4: Fine-Tuning with Domain-Specific Data
for each model $M$ in $\{\text{Llama-2}, \text{Llama-3}, \text{GPT-3.5 Turbo}, \text{GPT-4o Mini}\}$ do
    Fine-tune $M$ on financial review data, using techniques such as LoRA and 4-bit
        quantisation for efficient adaptation
    Update model parameters with optimised hyper-parameters (learning rate, batch
        size, gradient accumulation) as needed
Step 5: Sentiment Response Generation
for each model $M$ in $\{\text{Llama-2}, \text{Llama-3}, \text{GPT-3.5 Turbo}, \text{GPT-4o Mini}\}$ do
    Generate a sentiment classification output $S_i$ for the query $q$ using model $M$
Step 6: Ensemble Voting for Final Output
Combine the outputs $\{S_1, S_2, S_3, S_4\}$ from each model using majority voting to determine the final sentiment classification $S$
return $S$
```

3.3.3. Retrieval-Augmented Generation (RAG)

The RAG architecture combines document retrieval with generative sentiment analysis to improve the context relevance of output generated by the model. In this framework, relevant documents are first identified from a vector database and then provided to the LLMs for sentiment classification, which is particularly valuable for domain-specific tasks like analyzing financial reviews, as shown in Algorithm 5.

Algorithm 5 RAG with a Specific Retrieval Methodology and Parameter Tuning

```
Input: Query $q$, Document corpus $D$, Pre-trained language model $\phi$
Output: Generated response $r$
Encode query $q$ and documents in $D$ into embeddings using all-MiniLM-L6-v2
Retrieval Methodology: Retrieve top-$k$ relevant documents $D_k$ from FAISS index based
    on cosine similarity; $D$ is indexed with HNSW (Hierarchical Navigable Small World) to
    balance speed and accuracy for high-dimensional vectors
Parameter Tuning: Set $k = 5$ for retrieval depth, cosine similarity for distance metric,
    LLM decoding parameters: beam width = 3, temperature = 0.7, top-$k$ sampling = 40
for each document $d_i$ in $D_k$ do
    Concatenate query $q$ with $d_i$ to create context-enhanced input
    Generate response $r_i$ using language model $\phi$ with parameter-tuned decoding
Aggregate responses $r_i$ by majority voting to form the final response $r$
return $r$
```

The retrieval process begins by encoding both the query and the document corpus into dense embeddings using a pre-trained embedding model (e.g., all-MiniLM-L6-v2). FAISS is then used to efficiently search the vector database, retrieving the top k relevant documents for each query. The selection of k is optimised for each dataset, where an ideal balance between retrieval speed and contextual relevance was found at $k = 5$ based on validation tests. Additionally, cosine similarity was chosen as the distance metric in the FAISS index to improve matching accuracy.

RAG enhances the generative performance of LLMs by integrating these retrieved documents into the generation process. The system consists of two main components: a retriever and a generator. The retriever uses FAISS to find the most relevant documents for a query, and the generator then utilises the contextual information from these documents to generate a sentiment-annotated response. To control the relevance and specificity of generated responses, we set the LLMs' decoding parameters to a beam width of 3, a temperature of 0.7, and top-k sampling at 40.

In RAG, the probability of generating a response r given a query q is defined as

$$P(r \mid q) = \sum_{d \in D_k} P(r \mid q, d) P(d \mid q) \tag{7}$$

This equation sums over the top-k retrieved documents D_k, where $P(r \mid q, d)$ represents the probability of generating the response given the query and a specific document and $P(d \mid q)$ denotes the probability of retrieving the document given the query. The weighted approach allows the model to produce more relevant responses, enhancing its ability to manage complex or specialised data contexts [112]. This combined retrieval-generation approach enables the model to leverage domain-specific knowledge effectively, making RAG well suited for sentiment analysis tasks that rely on context-specific information.

3.3.4. Querying the Vector Database

The system uses similarity-based searching with FAISS to retrieve the top k documents related to a particular query, such as customer complaints and comments on banking services. The procedure begins with embedding creation, where fresh reviews are encoded into embeddings using a pre-trained embedding model (e.g., 'all-MiniLM-L6-v2'), which is efficient in encoding semantic information into dense vectors. These embeddings are added to the FAISS index, enabling efficient similarity-based querying and retrieval [113]. FAISS is widely adopted for large-scale similarity searches with minimal latency, making it ideal for real-time sentiment analysis activities [114].

Each review is stored in a global dictionary linking the index position to the actual review content, allowing easy access to the original review when necessary. A query is encoded into an embedding using the same pre-trained model, and the nearest neighbours are found using the FAISS index [115]. The RAG system then processes the retrieved documents to generate a response, ensuring that only the most relevant documents enhance the response's quality and accuracy. The Euclidean distance measures the similarity between the query embedding $\mathbf{q}$ and each document embedding $\mathbf{v}_i$, as shown below:

$$d(\mathbf{q}, \mathbf{v}_i) = \|\mathbf{q} - \mathbf{v}_i\|_2 \tag{8}$$

3.3.5. Models Fine-Tuning

The fine-tuning of LLMs such as Llama-2, Llama-3, GPT-4o Mini, and GPT-3.5 Turbo was performed with the goal of tailoring these models to the special language and cultural peculiarities of customer evaluations in the South African financial industry. This method used domain-specific data, targeted prompts, and sophisticated techniques such as Low-Rank Adaptation (LoRA) and quantisation to improve model performance in sentiment analysis tasks. The Llama-2 and Llama-3 models (e.g., meta-llama/Llama-2-7b-hf and meta-llama/Meta-Llama-3-8B-Instruct) were fine-tuned using a dataset of annotated customer reviews, reflecting sentiment expressions particular to the South African environment. The fine-tuning was led by a cross-entropy loss function that was intended to reduce classification mistakes. The approach entailed modifying pre-trained models by delivering structured input–output pairs while training. Prompts designed for Few-shot Learning, Chain-of-Thought reasoning, and ReAct were used to align model outputs with task objectives, allowing the models to capture nuanced sentiment cues present in financial reviews, including culturally relevant expressions and industry-specific terminology [37].

To improve computational efficiency, four-bit quantisation was used to reduce resource needs while retaining model performance [116]. The incorporation of LoRA allowed for parameter-efficient fine-tuning by selectively updating just a subset of model weights, making the process more flexible to resource restrictions [117]. The fine-tuning setup used a learning rate of 1×10^{-4}, a batch size of 4, gradient accumulation steps set to 8, and a cosine learning rate schedule. These settings were chosen to provide high validation data correctness while keeping the model flexible to a wide range of input changes and developing consumer sentiment patterns. Parallel fine-tuning was carried out on OpenAI models (*gpt-4o-mini-2024-07-18*) and (*gpt-3.5-turbo-0125*) using a similar technique, with extra flexibility offered by OpenAI's fine-tuning API. Training datasets were specially selected for South African financial reviews, and prompts were designed to aid the models' interpretation processes. This design enabled the models to efficiently read regional idioms, customer expectations, and sentiment subtleties specific to the local market.

The fine-tuning method includes the dynamic modification of hyper-parameters, including the number of epochs, the batch size, and the learning rate. These parameters were automatically tuned depending on dataset properties [118]. The model performance was assessed utilising evaluation criteria such as accuracy, precision, recall, and F1-score, with a focus on generalisation across a variety of review contexts. Both the Llama and GPT models showed considerable gains in sentiment classification accuracy after fine-tuning, demonstrating their utility in domain-specific sentiment analysis. The conversational fine-tuning approaches highlight the necessity of tailoring LLMs to specific industrial settings, especially in sectors where regional and cultural characteristics substantially affect attitude expression. This personalised methodology allows the models to better manage the intricacies of financial assessments, increasing their usefulness as instruments for strategic decision-making in consumer sentiment research.

3.3.6. Response Generation

The gathered and stored documents provide the context necessary for the Llama and GPT models to perform nuanced sentiment analysis that is both richly contextual and domain-specific. Every individual model produces a sentiment classification as "positive" or "negative" that pertains to the query and the relevant retrieved documents. For any sort of ambiguous or nuanced sentiment where the models may not reach a strong consensus, we get around this problem by employing a confidence score that is based on how well the models agree. A confidence score for the final sentiment prediction S_{final} is calculated based on the proportion of models that classified the sentiment similarly [119,120]. This score evaluates the reliability of sentiment classification and evaluates the confidence of our models. If we receive low confidence, we promptly implement further verification to ensure accuracy. Here is how we express the confidence score:

$$\text{Confidence}(S_{\text{final}}) = \frac{1}{n}\sum_{i=1}^{n}\delta(S_i, S_{\text{final}}) \tag{9}$$

where S_{final} represents the majority sentiment classification (positive or negative), S_i is the sentiment prediction from the ith model, n is the total number of models, and $\delta(S_i, S_{\text{final}}$ is a function that returns 1 if $S_i = S_{\text{final}}$ and 0 if $S_i \neq S_{\text{final}}$.

3.4. Inference Layer

The inference layer facilitates the deployment of the trained models for classification and regression tasks. It integrates seamlessly with the overall system, allowing for real-time sentiment analysis and decision-making in a production environment.

We have constructed and enhanced the model's architecture for English financial reviews. This could constrain the model's ability to conduct sentiment analysis and classification in low-resource South African languages. Possible solutions include the future use of multilingual embeddings and tuning on corpora in our national languages by using transfer learning techniques. All these methods could help make the model more broadly useful and indeed robust for understanding the kinds of sentiment expressed in financial reviews in low-resource languages.

3.4.1. Cross-Entropy on Label Task

The system optimises the LLMs using cross-entropy loss on labelled sentiment data. The cross-entropy function penalises incorrect predictions by comparing model-generated outputs against the ground truth labels. The function is defined as

$$\text{CrossEntropy}(y, \hat{y}) = -\sum_{i=1}^{n}(y_i \log(\hat{y}_i) + (1 - y_i)\log(1 - \hat{y}_i)) \tag{10}$$

where y_i represents the ground truth label for the i-th instance, $\hat{y}_i$ is the predicted probability for the i-th instance, and n is the total number of instances. This approach ensures that the model improves its classification accuracy over time by reducing the error between predicted and actual outcomes [121].

3.4.2. Model Inference

The inference process utilises a combination of fine-tuned Llama-2, Llama-3, and GPT models for sentiment classification. Each model generates a sentiment prediction based on the context provided by the RAG system. A majority voting mechanism is then employed to determine the final sentiment label, ensuring increased robustness and minimising individual model bias [122].

$$r_{\text{final}} = \arg\max_{r \in R} \sum_{i=1}^{n}\delta(r_i, r) \tag{11}$$

where

- r_{final} is the final sentiment label.
- R represents the set of all possible sentiment responses (e.g., positive or negative).
- r_i is the sentiment prediction from the i-th model.
- n is the total number of models.
- $\delta(r_i, r)$ is an indicator function, defined as

$$\delta(r_i, r) = \begin{cases} 1, & \text{if } r_i = r \\ 0, & \text{otherwise} \end{cases} \tag{12}$$

The function of indicator, $\delta(r_i, r)$, gives a value of 1 when the model prediction r_i matches the target response r and gives a value of 0 otherwise. The equation is used to calculate r_{final}, which is the sentiment label with the highest number of model votes. This voting mechanism ensures that the sentiment classification is consistent across different model outputs [123].

3.4.3. Continuous Learning

The model includes continuous learning by incorporating cross-domain datasets that go beyond just South African customer reviews, allowing it to better understand a wider range of sentiment expressions. This makes it more scalable, less biased, and able to adapt to global changes in trends and market dynamics. The model updates continuously, using today's update to span recent developments, ensuring it remains current and able to classify sentiment with more resilience and robustness across a range of events and contexts.

Continuous learning parameters θ are updated over time as follows:

$$\theta_{t+1} = \theta_t + \eta \cdot \nabla_\theta \mathcal{L}(\theta_t; \mathcal{D}_{\text{new}}) \tag{13}$$

where θ_t are the model parameters at time t, η is the learning rate, and $\nabla_\theta \mathcal{L}(\theta_t; \mathcal{D}_{\text{new}})$ is the gradient computed over new cross-domain data $\mathcal{D}_{\text{new}}$.

The vector database is also regularly updated by adding new embeddings:

$$V_{t+1} = V_t \cup \{\mathbf{v}_{\text{new}}\} \tag{14}$$

where V_t is the vector database at time t and $\mathbf{v}_{\text{new}}$ is the embedding of recent data.

3.4.4. Output

The final sentiment classification happens through a majority voting process. The output classifications of our multiple fine-tuned models—Llama-2, Llama-3, GPT-4o Mini, and GPT-3.5 Turbo— are then aggregated for the final decision. This ensemble method increases reliability by balancing individual model biases associated with each model's output. The result is a more even and consistent path to the final sentiment classification.

The sentiment score S for each review r is calculated as

$$S = \frac{1}{N} \sum_{i=1}^{N} w_i \cdot f_i(r)$$

where N is the total number of models, w_i is the weight assigned to model i based on its cross-domain performance, and $f_i(r)$ is the sentiment prediction from model i for review r.

To ensure cross-domain consistency, model weights w_i are optimised by minimising the domain-adaptation loss:

$$\mathcal{L}_{\text{adapt}} = \frac{1}{M} \sum_{j=1}^{M} \left(S_{j,\text{Hellopeter}} - S_{j,\text{cross-domain}} \right)^2$$

where M is the number of test reviews across both Hellopeter and cross-domain data, and $S_{j,\text{Hellopeter}}$ and $S_{j,\text{cross-domain}}$ represent the sentiment scores for review j in each dataset.

The final sentiment label S_{final} is determined by majority voting as follows:

$$S_{\text{final}} = \arg \max_{s \in \{\text{positive,negative}\}} \sum_{i=1}^{N} \delta(f_i(r), s)$$

where $\delta(f_i(r), s) = 1$ if $f_i(r) = s$ (indicating that model i predicts sentiment s) and 0 otherwise.

4. Experiments

This section presents the results of the sentiment analysis model, followed by a detailed discussion of the model's performance, cross-domain adaptability, and insights gained from the sentiment classification tasks. The evaluation covers key metrics such as accuracy, precision, recall, F1-score, and RAGAS, highlighting the effectiveness of integrating diverse datasets, continuous learning, and RAG techniques. We also examine the impact of fine-tuning on domain-specific data and discuss the model's robustness, scalability, and ability to generalise across different domains. The discussion extends to how RAGAS is used to assess the relevance, accuracy, and coherence of generated responses, further validating the model's effectiveness in real-world applications.

4.1. Data Acquisition

The primary data were collected from Hellopeter, a popular website that is used to review financial services in South Africa, and the data collected are in the form of customer reviews. Data were retrieved via API endpoints for several major banks, including Standard Bank, Nedbank, Absa, Capitec, and First National Bank, as shown in Table 3. Additionally, cross-domain datasets were integrated into the model to enrich sentiment analysis by providing diverse contexts and sentiment expressions, summarised in Table 4.

Table 3. Hellopeter dataset summary

Dataset	Train	Validation	Test	Total Size	Tokens
Hellopeter Reviews	103,472	12,934	12,935	129,341	8,340,421

Table 4. Cross-domain experimental dataset description

Datasets	Domain	Total Size	Tokens
SemEval-2014 [124]	Restaurants	2503	54,086
IMDB-50k [125]	Movies	50,000	13,974,161
TripAdvisor [126]	Hotels	20,491	2,389,961
SemEval-2016 Stance [127]	Politics	26,750	604,358
Financial PhraseBank [128]	Financial News	2264	50,891

The diverse datasets ensure that the model remains adaptable across various domains while maintaining accuracy and relevance. To efficiently retrieve and process the Hellopeter reviews, we implemented a data extraction algorithm detailed in Algorithm 6. The algorithm fetches review data from the API endpoints and processes them into a structured format suitable for analysis. The reviews are collected from multiple pages while handling noise and invalid data to ensure the quality of the processed output.

Algorithm 6 Fetching and Processing Hellopeter Reviews

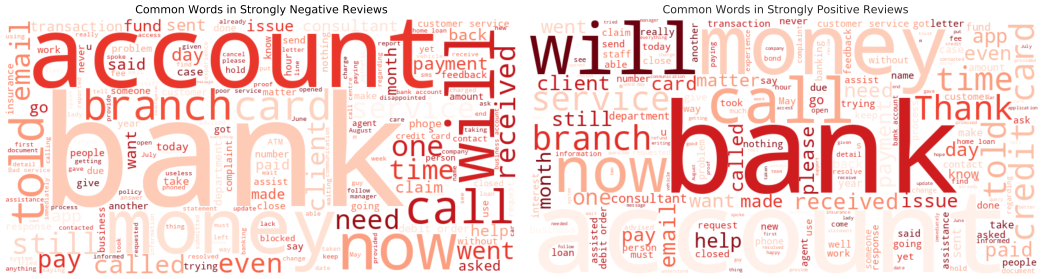
```
Input: List of API endpoints (URLs), Maximum pages to retrieve ($P_{\max}$)
Output: Collection of processed reviews $R_{\text{processed}}$
Initialise $R_{\text{processed}} \leftarrow \{\}$
for each url in URLs do
    $response \leftarrow$ Fetch initial data from $url$
    $P_{\text{total}} \leftarrow \min(\text{Extract total pages}, P_{\max})$
    for $p = 1$ to $P_{\text{total}}$ do
        $data \leftarrow$ Fetch data from page $p$
        if data is valid then
            Extract relevant fields: created_at, review_title, review_content, etc.
            $review \leftarrow$ Combine relevant fields into a single text string
            $R_{\text{processed}} \leftarrow R_{\text{processed}} \cup \{review\}$
return $R_{\text{processed}}$
```

4.2. Exploratory Data Analysis

We took a set of numbers and ran them through a polarity score generator to obtain a favourable look at the emotional distribution of the Hellopeter dataset. Polarity scores shed light on the emotional intensity as well as the variability of customer reviews. With this intelligence, we now have a more nuanced picture from which to judge our proposed model's performance across the range of different emotional intensities that make up the Hellopeter dataset.

4.2.1. Sentiment Word Cloud

Insights into the prevalent themes of customer feedback were gained by creating word clouds for both the strongly negative and strongly positive reviews, as seen in Figure 5. The negative reviews, as shown in Figure 5a, frequently mention the words "account", "money", "pay", and "branch", which point to issues with financial transactions and serious frustrations with customer service. The word cloud for the negative reviews almost screams out the banks' problems in these areas. On the other hand, the positive reviews Figure 5b concentrate on the words "thank", "service", and "help", which seem to suggest no problems at all. Although the clouds do not capture the full context of either group of customers' experiences, they do sum up well the principal areas of concern and satisfaction.



(a) (b)

Figure 5. Word clouds of frequently used terms in negative and positive Hellopeter reviews. (**a**) Negative; (**b**) Positive.

4.2.2. Sentiment Polarity Distribution

We trained the proposed LFEAR on Hellopeter review data. As seen in Figure 6, the polarity scores from these reviews were quite unevenly distributed between negative and positive scores. A greater spread of negative scores skews the distribution, indicating a higher degree of user dissatisfaction. This variability—which is actually our model's higher level of sensitivity to negative sentiment—allows it not only to capture more instances of user dissatisfaction but also to shift those instances against a stable background of positive sentiment. Such insights improve the model's responsiveness to both mild and intense sentiments, strengthening its overall classification performance.

4.2.3. Emotional Phrasing Analysis

We delved deeper into the emotional weight factor, using Scattertext to locate the phrases that are most often linked with either high or low emotional weight. Figure 7 shows where the frequently occurring phrases in our high-emotional-weight reviews (e.g., "terrible experience" or "absolutely amazing") are in relation to the frequently occurring phrases in our reviews that have only moderate emotional weight. Expressions with moderate emotional weight, such as "excellent service", cluster near phrases expressing positive sentiment, while phrases with high emotional weight, such as "worst experience", have a negative sentiment. This distinction is a key part of calibrating the model, allowing it to focus even more closely on better interpreting the range of sentiments that phrases express.

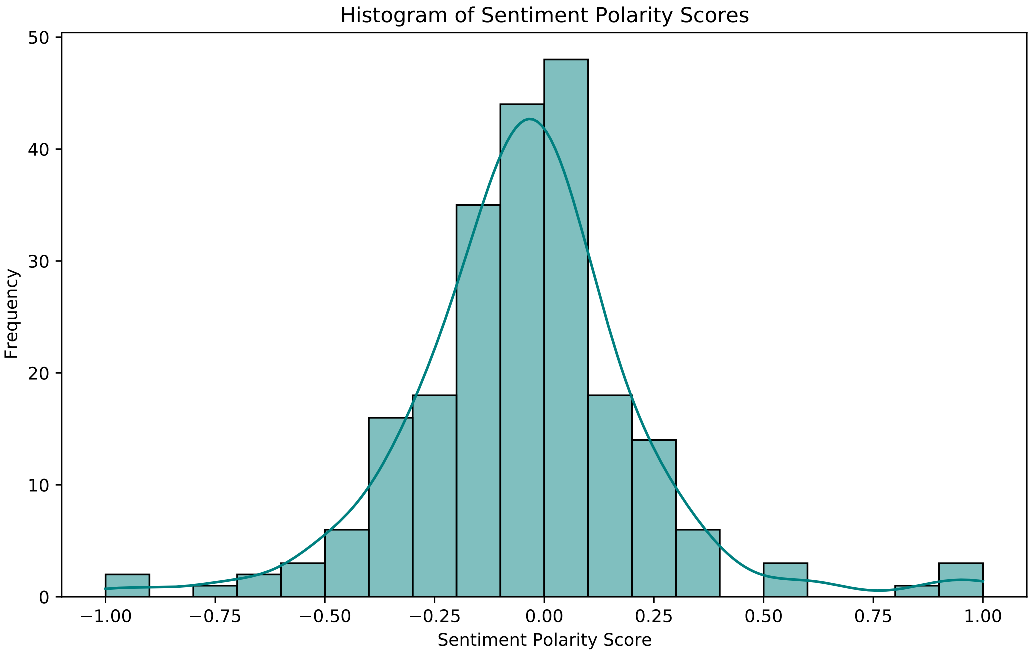


Figure 6. Distribution of sentiment polarity scores in Hellopeter reviews

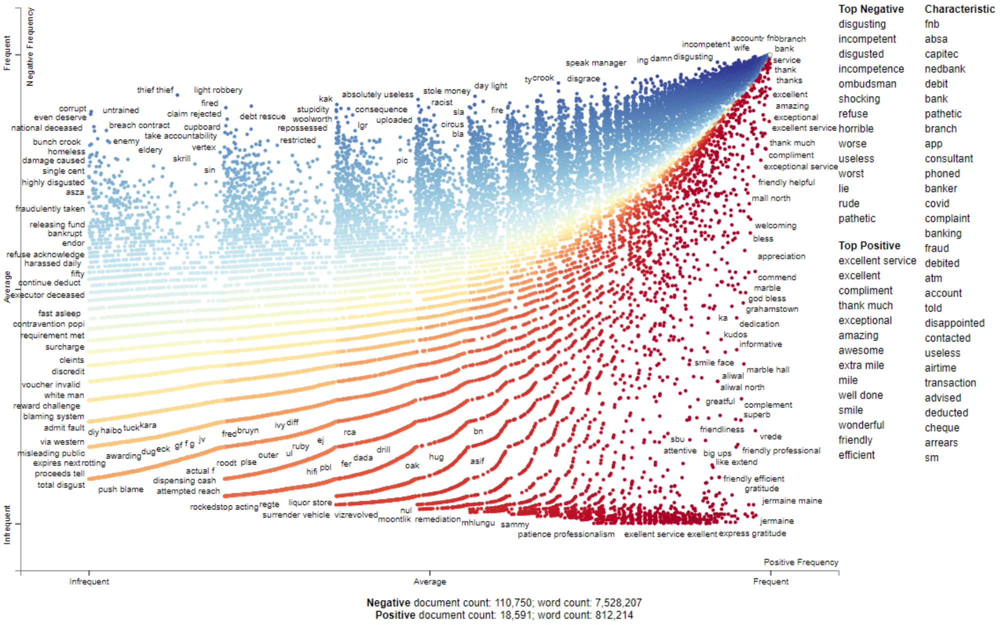


Figure 7. Scattertext visualisation of positive and negative words.

4.3. Experiment Setup

For our experiments, we fine-tuned four state-of-the-art language models, GPT-4o-mini (*gpt-4o-mini-2024-07-18*), GPT-3.5 Turbo (*gpt-3.5-turbo-0125*), Llama2 (*meta-llama/Llama-2-7b-hf*), and Llama3 (*meta-llama/Meta-Llama-3-8B-Instruct*), with a focus on sentiment analysis within the South African financial sector. The GPT-4o-mini model, which features a context window of 128,000 tokens and a maximum output of 16,384 tokens, offered an efficient and cost-effective solution. GPT-3.5 Turbo, with a context window of 16,385 tokens, served as a benchmark for comparison. The Llama models, Llama2 and Llama3, provided robust performance in processing long-context reviews, making them ideal for this domain-specific task.

All experiments were conducted on a Google Cloud Platform (GCP) instance configured with 2 NVIDIA A100 GPUs (40GB each) and an Intel Cascade Lake CPU, running on a Deep Learning VM with CUDA 11.8 and Python 3.10. The models were fine-tuned using a batch size of four per device, eight gradient accumulation steps, and a learning rate of 1×10^{-4}, utilising mixed precision (fp16) for efficiency. The fine-tuning process spanned five epochs, with evaluations conducted every 20% of the training process, ensuring optimal model adaptation and generalisation across varied sentiment expressions.

The detailed hyper-parameters and training configurations for each model are summarised in Table 5.

Table 5. Summary of fine-tuned models and hyper-parameters

Model	Context Window	Max Output Tokens	Training Data	Trained Tokens	Learning Rate	Batch Size	Epochs
GPT-4o-mini-2024-07-18	128,000 tokens	16,384 tokens	Up to October 2023	3,909,592	1.8×10^{-4}	12	3
GPT-3.5-turbo-0125	16,385 tokens	4096 tokens	Up to September 2021	4,617,049	2×10^{-4}	10	3
Llama-2-7b-hf	8192 tokens	4096 tokens	Up to December 2022	-	1×10^{-4}	4	5
Llama-3-8B-Instruct	8192 tokens	4096 tokens	Up to December 2023	-	1×10^{-4}	4	5

4.4. Data Augmentation

The problem of data imbalance is also often encountered in sentiment analysis, especially in such areas as customer feedback, where the amount of negative sentiment usually far exceeds the number of positive sentiments. In order to minimise this imbalance and eliminate potential model bias, we employed specific methods that directly improve the fairness and sample diversity of the training phase. In our dataset, 70% of the reviews were classified as negative, and therefore, during the fine-tuning phase, we used class weighting as well as other forms of sampling.

The primary technique involved dynamically adjusting the loss function to apply higher penalties for classifying the minority of the class positive sentiment wrongly [129]. This approach ensured that the model remained sensitive to underrepresented classes, promoting balanced learning across sentiment categories. Additionally, we employed stratified sampling, which maintains the class distribution during mini-batch creation, helping the model encounter an even mix of positive and negative examples throughout training [130]. By incorporating these approaches, we reduced model bias and enhanced the model's ability to generalise, thus producing better and more equitable sentiment classifications. These techniques are helpful in such areas as financial sentiment analysis, where the presence of bias in the predictions would negatively affect the decision-making process.

4.5. Performance Measurement

The performance of the sentiment analysis model was evaluated using RAGAS, sentiment intensity scoring, granular sentiment classification, Vendi Score, and evaluation metrics such as F-measures and Accuracy.

4.5.1. Evaluation Metrics

To evaluate the performance of our sentiment analysis models, we used standard metrics such as Accuracy (ACC), Precision, Recall, and the F-measure. These metrics provide a comprehensive assessment of the model's effectiveness in predicting sentiment across both positive and negative classes.

The mathematical formulation for Accuracy is

$$\text{Accuracy (ACC)} = \frac{TP + TN}{TP + TN + FP + FN} \tag{15}$$

where

- TP represents True Positives.
- TN represents True Negatives.
- FP represents False Positives.
- FN represents False Negatives.

Precision is calculated as

$$\text{Precision} = \frac{TP}{TP + FP} \tag{16}$$

Recall is defined as

$$\text{Recall} = \frac{TP}{TP + FN} \tag{17}$$

The F-measure (F1-score) is calculated as

$$F\text{-measure} = 2 \times \frac{\text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}} \tag{18}$$

4.5.2. Retrieval-Augmented Generation Assessment

We incorporate specialised RAGAS metrics to evaluate the performance of our RAG models. Specifically, we utilise Answer Relevance, Answer Correctness, Answer Similarity, Answer Recall, and Answer Precision [131]. The Answer Relevance metric assesses the relevance of the generated answer to the given prompt, calculated using the cosine similarity between the embeddings of the generated answer and the original question:

$$\text{Answer Relevance} = \frac{1}{N} \sum_{i=1}^{N} \cos(\mathbf{E}_{a_i}, \mathbf{E}_o) \tag{19}$$

where

- N is the number of generated answers.
- $\mathbf{E}_{a_i}$ is the embedding of the generated answer.
- $\mathbf{E}_o$ is the embedding of the original question.

The Answer Correctness metric evaluates the accuracy of the generated answer by considering both semantic similarity and factual similarity. The correctness score is calculated as

$$\text{Answer Correctness} = \alpha \times \text{Semantic Similarity} + (1 - \alpha) \cdot \text{Factual Similarity} \tag{20}$$

where

- α is the weight assigned to semantic similarity.
- **Semantic Similarity** is the cosine similarity between the embeddings of the generated answer and the ground truth.
- **Factual Similarity** measures the overlap of factual information between the generated answer and the ground truth.

These specialised RAGAS metrics enable us to quantify the performance of our RAG system. By evaluating Answer Relevance, Answer Correctness, Answer Similarity, Answer Recall, and Answer Precision, we can assess the quality of the generated responses within the context of sentiment analysis.

4.5.3. Integrative Sentiment Analysis Metrics

We use three essential measures in our sentiment analysis approach to ensure a deep and precise understanding of our customers' feedback: the intensity of the expressed sentiment, the granular classification of the sentiment, and the Vendi score [132]. We use VADER to calculate the intensity of the expressed sentiment. It assigns one of four levels to each piece of feedback, from strongly positive to strongly negative, capturing both the polarity and the strength of the expressed sentiment. A second key measure in our analysis is the classification level of the sentiment expressed. This is important because, unless one is using a very simplistic model, the sentiment expressed is rarely simply positive or negative. We aim to classify expressed sentiment at a level where one can only classify it wrongly if one is npt being at least somewhat nuanced in terms of what different types of sentiments there might be. Finally, the Vendi score provides an overall accuracy metric, representing the ratio of correct predictions to total samples, which helps assess the model's reliability across all sentiment levels clearly and understandably [133]. Measuring Diversity with the Vendi score is expressed in the mathematical equation below:

$$\mathrm{VS}_k(x_1, \ldots, x_n) = \exp\left(-\sum_{i=1}^{n} \lambda_i \log \lambda_i\right) \tag{21}$$

where

- λ_i represents the i-th eigenvalue of the kernel matrix K/n;
- K is a similarity matrix where each entry $K_{i,j} = k(x_i, x_j)$ is defined by a similarity function $k : X \times X \to \mathbb{R}$.

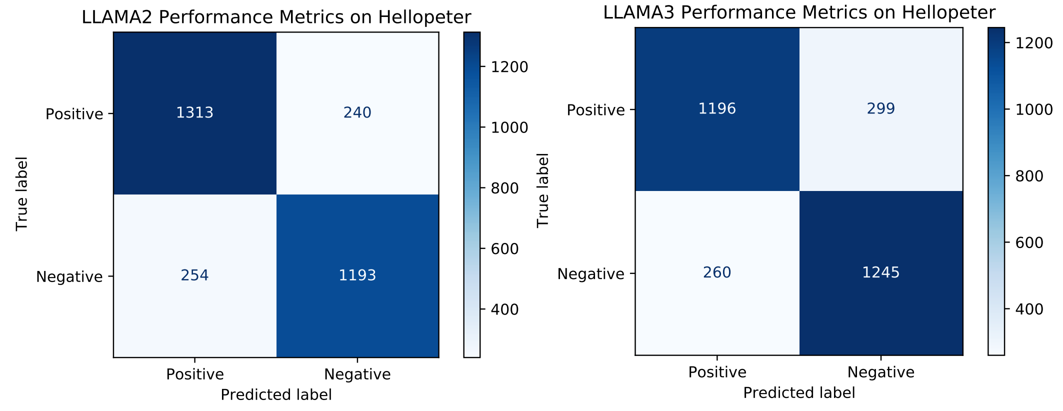
4.6. Performance Evaluation

The performance of our fine-tuned models on the Hellopeter dataset, which served as the primary dataset for fine-tuning, is summarised in Table 6. All models yielded good performance, and it was evident that all models were capable of capturing sentiment in the financial context. The *gpt-4o-mini-2024-07-18* model achieved the highest accuracy of 89.15%, followed by *gpt-3.5-turbo-0125* at 88.95%. The *Llama-2-7b-hf* model, while slightly lower in accuracy at 81.15%, showed competitive performance with a precision of 88.05% and an F1-score of 88.10%. The *Meta-Llama-3-8B-Instruct* model also performed well, with an accuracy of 87.05%, indicating that both Llama models are effective in aligning with the sentiment nuances specific to South African financial reviews.

Table 6. Performance evaluation on the Hellopeter dataset.

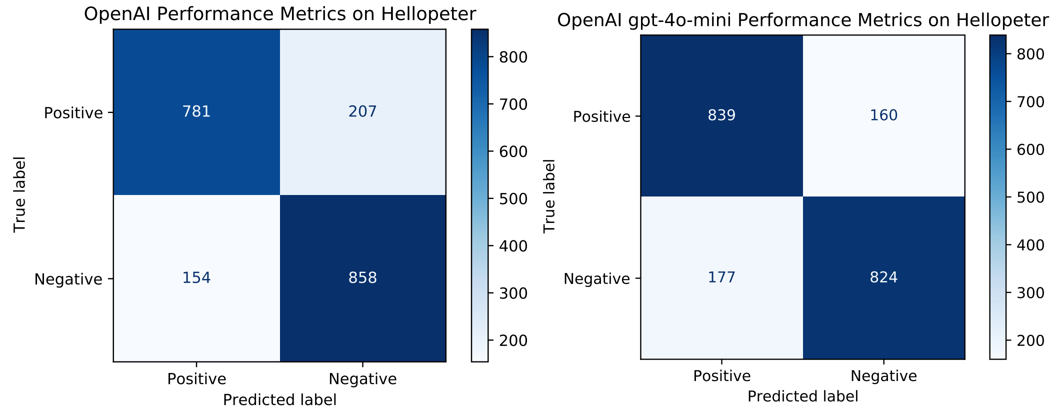
Model	Dataset	Domain	Accuracy	Precision	Recall	F1-Score
Llama-2-7b-hf	Hellopeter	Financial Products	0.8115	**0.8805**	0.8244	0.8810
Meta-Llama-3-8B-Instruct	Hellopeter	Financial Products	0.8705	0.8138	0.8265	0.8558
gpt-3.5-turbo-0125	Hellopeter	Financial Products	0.8895	0.8203	0.8195	0.8793
gpt-4o-mini-2024-07-18	Hellopeter	Financial Products	**0.8915**	0.8316	**0.8315**	**0.8815**

The confusion matrices for the Llama models, as depicted in Figure 8, also show the results on the Hellopeter dataset. Hence, the findings show that the classification is evenly distributed between the positive and negative sentiment classes with a low misclassification level, as highlighted by their high recall and precision. The confusion matrices of the GPT models are presented in Figure 9, and it can be seen that both models achieve comparable results, with the sentiment trends in the financial product reviews being predicted well by both models, with the GPT models having a slightly better accuracy than the Llama models.



(**a**) (**b**)

Figure 8. Confusion matrices for Llama models on the Hellopeter dataset. (**a**) Llama-2-7b-hf confusion matrix; (**b**) Meta-Llama-3-8B-Instruct confusion matrix.



(**a**) (**b**)

Figure 9. Confusion matrices for GPT Models on the Hellopeter Dataset. (**a**) gpt-3.5-turbo-0125 confusion matrix; (**b**) gpt-4o-mini-2024-07-18 confusion matrix.

Table 7 presents the cross-domain evaluation results across multiple datasets, allowing us to assess the adaptability of the models to data they were not primarily fine-tuned on. The results demonstrate that the models retained competitive performance across domains, highlighting their generalisation ability. For instance,the *gpt-3. 5-turbo-0125* model had an accuracy of 85 percent. On the Financial PhraseBank dataset, the score is 95%, and on the SemEval-2014 Restaurant dataset, it is 77. 80%, which shows that it works well in other areas in addition to the financial industry. On the other hand, the *Llama-2-7b-hf* model performed well in terms of adaptability, especially in the IMDB dataset, and an F1-score of 88 was attained, 50% of which is quite good considering the fact that the dataset mainly contains movie reviews.

The cross-domain evaluation results indicate that although there is a small drop in performance as compared to the Hellopeter dataset, the models are still able to generalise well. This flexibility is important in order to make sure that the models work not only in the financial domain in which they were trained but also in different areas such as movie reviews, hotel feedback, political speeches, and restaurant reviews. These results also confirm the effectiveness of the fine-tuning process that allows the models to retain high accuracy, precision, recall, and F1-score in different domains, which proves the applicability of the models in various sentiment analysis tasks.

Table 7. Cross-domain evaluation results on different LLMs.

Model	Dataset	Domain	Accuracy	Precision	Recall	F1-Score
Llama-2-7b-hf	Financial PhraseBank	Financial News	0.8118	0.8043	0.7632	0.7768
Meta-Llama-3-8B-Instruct	Financial PhraseBank	Financial News	0.7897	0.8341	0.7026	0.7207
gpt-3.5-turbo-0125	Financial PhraseBank	Financial News	**0.8595**	**0.8607**	0.8595	0.8600
gpt-4o-mini-2024-07-18	Financial PhraseBank	Financial News	0.7828	0.7785	0.7828	0.7794
Llama-2-7b-hf	IMDB 50K	Movies	0.7034	0.7466	**0.8968**	0.8850
Meta-Llama-3-8B-Instruct	IMDB 50K	Movies	0.7145	0.7455	0.7992	0.8357
gpt-3.5-turbo-0125	IMDB 50K	Movies	0.7330	0.7629	0.7330	0.7237
gpt-4o-mini-2024-07-18	IMDB 50K	Movies	0.7110	0.7609	0.7110	0.7941
Llama-2-7b-hf	TripAdvisor	Hotels	0.7210	0.7964	0.7163	0.7992
Meta-Llama-3-8B-Instruct	TripAdvisor	Hotels	0.7593	0.7872	0.6528	0.8110
gpt-3.5-turbo-0125	TripAdvisor	Hotels	0.8490	0.7849	0.6490	**0.8962**
gpt-4o-mini-2024-07-18	TripAdvisor	Hotels	0.8210	0.7777	0.7210	0.8523
Llama-2-7b-hf	SemEval-2016 Stance	Political	0.7124	0.7336	0.7010	0.6973
Meta-Llama-3-8B-Instruct	SemEval-2016 Stance	Political	0.7146	0.7630	0.8891	0.7223
gpt-3.5-turbo-0125	SemEval-2016 Stance	Political	0.7450	0.7460	0.7450	0.7435
gpt-4o-mini-2024-07-18	SemEval-2016 Stance	Political	0.7140	0.7312	0.7140	0.7040
Llama-2-7b-hf	SemEval-2014	Restaurant	0.7277	0.7637	0.7277	0.7180
Meta-Llama-3-8B-Instruct	SemEval-2014	Restaurant	0.7926	0.7446	0.6926	0.8177
gpt-3.5-turbo-0125	SemEval-2014	Restaurant	0.7780	0.7962	0.7780	0.7745
gpt-4o-mini-2024-07-18	SemEval-2014	Restaurant	0.7736	0.7903	0.7736	0.7703

4.7. Evaluation of Inference Model

The proposed inference model was evaluated on the Hellopeter dataset to assess its effectiveness in the sentiment analysis of financial products. As depicted in Table 8, the model achieved an impressive accuracy of 92.76%, which is much higher than the previously fine-tuned LLMs discussed in the previous sections, and the model achieved a precision of 87.65% and a recall of 88. Its accuracy was 94.06%, while its F1-score was 91.23%, thus proving the model's strong capacity to keep a good balance between precision and recall. These results confirm that the proposed inference model is well suited for sentiment analysis tasks in the financial context and provides accurate and reproducible predictions.

Table 8. Proposed inference model performance evaluation on the Hellopeter dataset.

Model	Dataset	Domain	Accuracy	Precision	Recall	F1-Score
Proposed Inference Model	Hellopeter	Financial Products	0.9276	0.8765	0.8806	0.9123

Figure 10 provides a confusion matrix that visualises the performance of the proposed inference model. The model demonstrates strong classification abilities, as evidenced by the distribution of correct and incorrect predictions.

In addition to traditional performance metrics, the model's effectiveness was further assessed using RAGAS-specific metrics, focusing on aspects such as answer quality and context retrieval. The RAGAS metrics, as shown in Table 9, demonstrate the model's proficiency in producing relevant and contextually accurate responses. The model achieved an average answer similarity of 98.45% and an answer correctness score of 93.85%, accompanied by a context recall of 78.97% and a context precision of 97.69%. These outcomes also provide evidence that the developed inference model outperforms not only in terms of sentiment analysis but also in the ability to generate contextually relevant and semantically accurate responses, thus supporting the model's credibility.

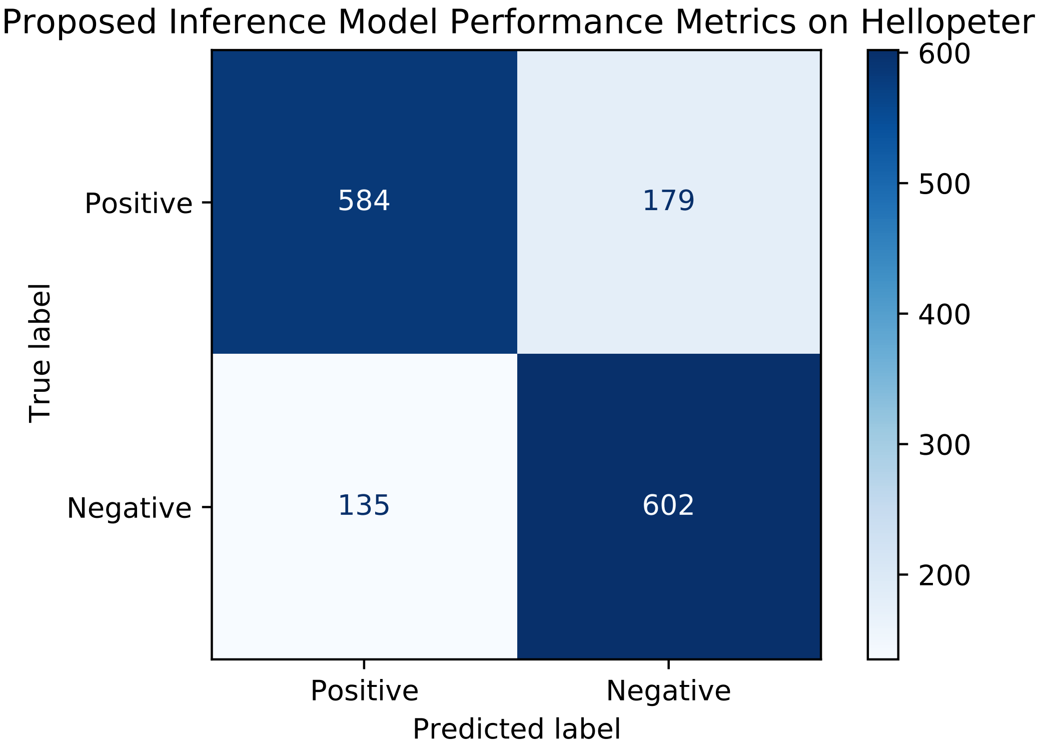


Figure 10. Confusion matrix for the proposed inference model on the Hellopeter dataset.

Table 9. RAGAS metrics for the proposed inference model.

Model	Dataset	Domain	RAGAS Metric	Average Score
LFEAR	Hellopeter	Financial Products	Answer Similarity	0.9845
			Answer Correctness	0.9385
			Answer Relevancy	0.8775
			Context Recall	0.7897
			Context Precision	0.9769

To comprehend the robustness of the model when it comes to differing emotional intensities present within the data, we also assessed it against several sentiment intensity metrics. In Table 10, we display the breakdown of these metrics for LFEAR. What is presented is a distribution of not only the sentiment polarity that is associated with our LFEAR outputs but also the average Vendi scores across several categories of sentiment intensity, including categories like Moderately Negative, Moderately Positive, Strongly Negative, and Strongly Positive. When it comes to results from the model for negative sentiments, we obtain a robust average polarity score of about -0.806 and even more so for our associated Vendi score, which averages about -0.854, indicating that we have a pretty high confidence level in our detection of instances of intense negativity in our dataset. On the other end of the spectrum, when we test some strongly positive sentiment, we obtain virtually the same average sentiment polarity score of 0.806, which suggests that our positive sentiment detection capabilities are almost as effective as our negative ones. These outcomes are illustrated in greater detail in Figure 10. The plot in Figure 11 shows the performance of RAGAS. The performance metrics shown here suggest that RAGAS is not only proficient in understanding kinds of queries that are specific to our context but that it is also excellent in terms of answer quality and precision. Figure 12 visualises the distribution of sentiment polarity and Vendi scores across the Hellopeter dataset, highlighting the model's nuanced performance in handling different sentiment intensities.

We have thoroughly evaluated the proposed inference model using standard performance metrics, RAGAS-specific metrics, and sentiment intensity measures. The results confirm its versatility and accuracy, showcasing a balanced approach to precision and recall—making it a suitable candidate for financial applications that rely on sentiment analysis.

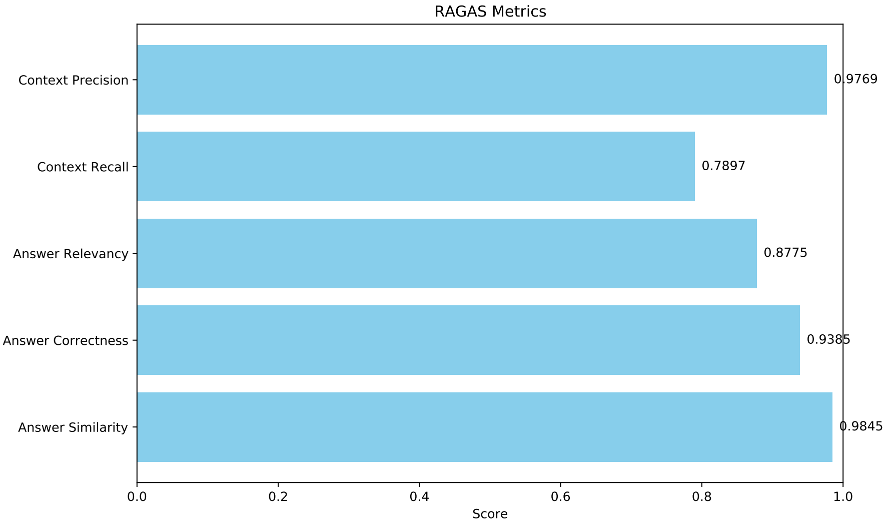


Figure 11. RAGAS performance metrics for the proposed inference model on the Hellopeter dataset.

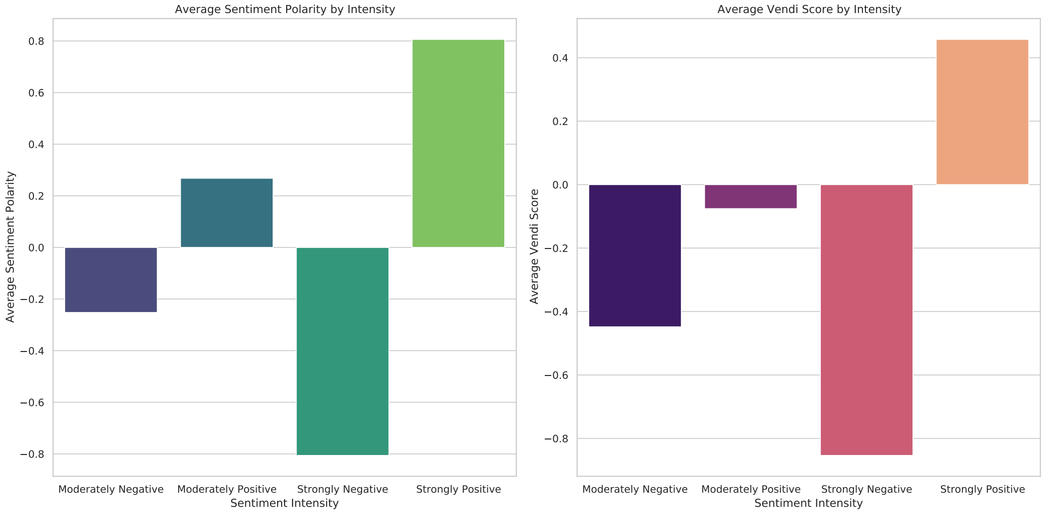


Figure 12. LFEAR distribution of results.

Table 10. Sentiment intensity metrics for the LFEAR model.

Model	Dataset	Sentiment Intensity	Count	Average Sentiment Polarity	Average Vendi Score
LFEAR	Hellopeter	Moderately Negative	1876	−0.252186	−0.448305
		Moderately Positive	1245	0.268085	−0.075834
		Strongly Negative	5121	−0.806115	−0.853706
		Strongly Positive	2703	0.806144	0.457808

5. Results and Discussion

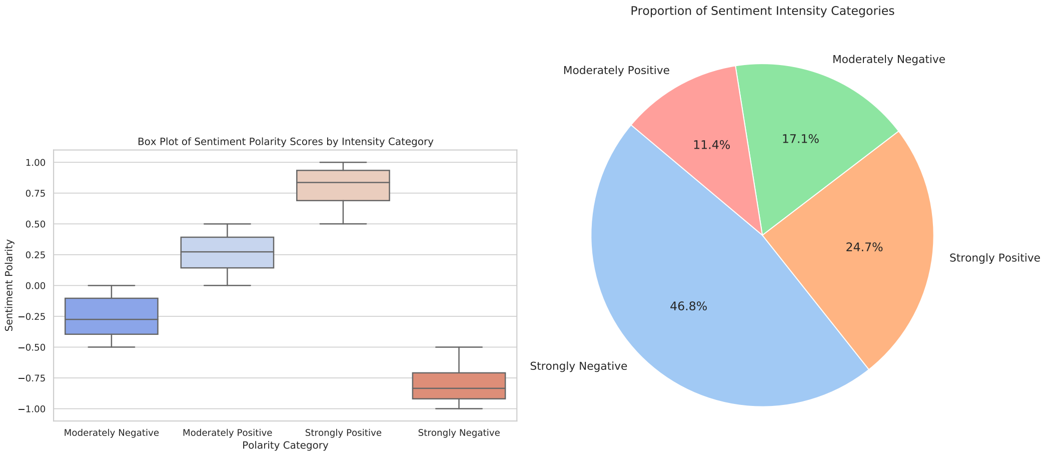
5.1. Results

The findings support the use of the proposed sentiment analysis model, especially in the context of South African financial product reviews where simple praise and criticism do not suffice. The model scored an effective accuracy of 92.76% in the analysis of the

Hellopeter dataset, with both high precision (87.65%) and high recall (88.06%) comparable to F1 (91.23%). LFEAR's impressive performance on Hellopeter and other domains can be explained by the use of customised LLMs, RAG, and continuous learning techniques. The fine-tuning that is performed here on the local and business-related language helps to improve the accuracy of the model toward classifying complex sentiment patterns in the financial domain. The integration of RAG, which dynamically retrieves relevant context from an updated vector database, ensures that sentiment predictions remain contextually aligned with real-time market trends. Such a feature would be particularly important for financial companies that want to adjust their approach alongside the changes in consumers' attitudes.

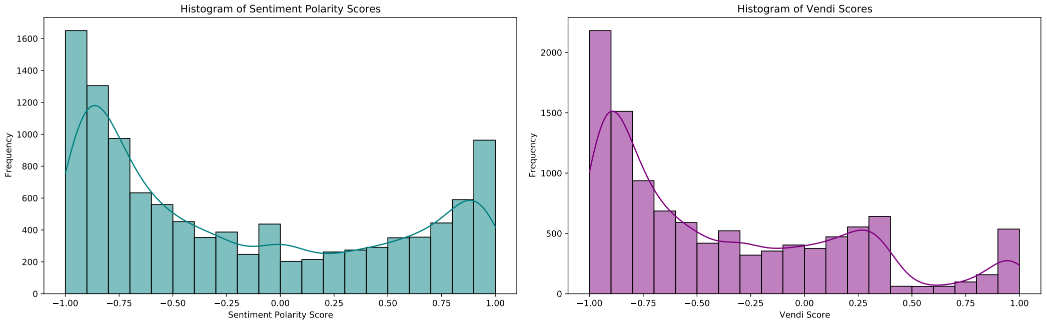
The performance metrics of the LFEAR model, as revealed in Tables 9 and 10, underscore its sturdiness and effectiveness in sentiment analysis, with a particular focus on financial product reviews. The RAGAS metrics are displayed in Table 9, and at their core, they provide a look at how well the LFEAR model answers questions posed to it. The key takeaway from the RAGAS evaluation is that the LFEAR model's answer accuracy is exceedingly high and very close to what one might expect from a human responder. The Answer Correctness metric of 0.9385 shows a very high level of accuracy. The Context Precision of 0.9769 indicates an extraordinary ability to put the answers in the proper context. These admirable levels of performance make the model a prime candidate for real applications where relevant answers and accurate context are not just desirable to have but necessary, especially in applications like finance that require pinpoint sentiment differentiation. The data in Table 10 reinforce the impression that LFEAR handles not just sentiments but also sentiment intensity quite well. In the Hellopeter dataset, the average sentiment polarity for Strongly Negative is -0.806115, and for Strongly Positive, it is 0.806144. These numbers show that LFEAR is not only recognising the emotional expression of this data but also scoring it with excellent precision. The model has an average Vendi score of -0.853706 for Strongly Negative sentiments, which shows it has a really good ability to detect high-intensity dissatisfaction. On a practical level, that is a really good feature to have when modeling customer sentiments; it indicates when your customers are showing service dissatisfaction.

The quantitative outcomes undoubtedly underscore LFEAR's state-of-the-art performance. They exhibit the model's ability to address the two fundamental research gaps in sentiment analysis—that is, precision and context awareness. Both sentiment misclassification and inadaptability to the real-world context often plague traditional models. By leveraging fine-tuning and RAG integration, LFEAR can not only detect accurate sentiment but also achieve real-time contextual adaptation, leading traditional sentiment analysis models into a new era of performance. This is evidenced in Tables 9 and 10, where the LFEAR model is shown to exhibit consistent appearance and performance across a wide variety of sentiment types and domains—something traditional models have been unable to achieve. To further this model's real-world adaptability, we have incorporated a capacity for continuous learning combined with scalable inference layers. LFEAR's high RAGAS scores, sentiment polarity accuracy, and ability to generalise across domains affirm its scalability, robustness, and suitability as a leading sentiment analysis tool in industry-specific applications. The capacity of LFEAR to handle the different intensities of sentiment is highlighted in Figure 13a,b. These show how well LFEAR distinguishes not just between positive and negative sentiments but also between sentiments of varying strength—a distinction that is crucial for many applications. Indeed, LFEAR appears to achieve state-of-the-art performance in this respect. Figure 14a,b illustrate and provide some further detail regarding this aspect of LFEAR's performance.



(**a**) (**b**)

Figure 13. Sentiment intensity analysis results for the Hellopeter dataset using LFEAR. (**a**) LFEAR Sentiment intensity distribution in the Hellopeter dataset; (**b**) LFEAR proportion of sentiment categories in the Hellopeter dataset.



(**a**) (**b**)

Figure 14. LFEAR performance on the Hellopeter dataset. (**a**) Polarity score distribution; (**b**) Vendi score distribution.

The evaluation of the model across domains leaves no doubt with regard to their robustness and flexibility. While mostly trained on financial review datasets, the model has been shown to work well on other datasets, namely restaurant, movie, hotel, or even political reviews. What is responsible for this kind of flexibility is the fact that the model utilises a cross-domain architecture comprising fine-tuned layers for specific domains alongside a scalable inference layer that is capable of processing diverse sentiment contexts. Additionally, continuous learning capabilities enhance the model's ability to learn over time and reinforce its ability to meet intelligence patterns, which change with the attitudes of people as the system mostly learns from customers. However, the constant efficacy illustrated over the different domains leads us to the conclusion of the validity of the proposed methodology and broadens the areas of its utilisation beyond finance to healthcare, retailing, and public opinion monitoring. The model's scalable and removable structure allows embedding it into practical applications without any performance loss. Its features of continuous learning and the ability to continually update the vector database make it robust to increased volumes of data and changing contexts. Testing the model within live systems, such as in financial firms, would reveal its ability to track trend sentiments relative to important financial indicators (changes in stock price, customer longevity, etc.) and correlate them with a specific time. Also, embedding such a model into customer relationship management (CRM) or advisory services would be beneficial since it could provide intelligence that would help enhance business processes and communication. This flexibility ensures the relevance of the model, improving the interactions and decision-making by stakeholders.

5.2. Limitations and Future Work

While the LFEAR model demonstrates considerable strengths in sentiment analysis, it has certain limitations that warrant further exploration. One limitation lies in the use of a single vector database for context retrieval, which could impact scalability as the volume and diversity of datasets grow. This limitation may lead to decreased retrieval efficiency and accuracy when handling large, heterogeneous datasets with varied sentiment expressions. Furthermore, data access restrictions under the Protection of Personal Information Act (POPIA) constrained our data inputs from certain mobile applications, such as WhatsApp, thus limiting the diversity and richness of sentiment sources within the financial domain. Additionally, while LFEAR effectively interprets general sentiment patterns, it may face challenges with highly nuanced or ambiguous sentiment expressions, which can limit its precision in cases where sentiments are complex or contextually specific. The model's architecture is optimised for English-language financial reviews, and its adaptability to non-English or low-resource languages remains untested, potentially limiting its transferability across different linguistic contexts. These limitations highlight areas where improvements could enhance the robustness and adaptability of the model in dynamic data environments.

Future work should focus on several promising avenues to enhance the LFEAR model's performance and broaden its applicability. First, addressing scalability issues by incorporating advanced retrieval techniques, such as hierarchical indexing, distributed vector databases, or hybrid retrieval models, could improve its capacity to handle extensive and varied datasets. Furthermore, expanding the model's adaptability to multilingual sentiment analysis—particularly in low-resource languages such as indigenous South African languages—could significantly increase its utility in diverse financial markets. Additionally, applying clustering algorithms during preprocessing may improve document classification and retrieval accuracy by grouping similar sentiment patterns before analysis, which can enhance the model's overall efficiency and precision. Finally, exploring meta-learning and reinforcement learning approaches may allow the model to better interpret complex sentiment nuances and adapt to evolving sentiment trends, making it a more versatile tool for sentiment analysis across multiple domains and languages.

6. Conclusions

This paper proposes LFEAR, a novel sentiment analysis architecture suitable for the South African environment, with a focus on the financial sector and consumer reviews. Through the integration of fine-tuned LLMs, RAG, and continuous learning, LFEAR establishes a connection between global knowledge and domain-specific expertise, enabling its widespread application in sentiment interpretation for intricate real-world applications. While sentiment intensity, granularity classification, and the Vendi score have already introduced some enhancements over basic sentiment analysis, LFEAR further contributes to this improvement by incorporating RAGAS. With this addition, we can potentially provide a more comprehensive analysis of LLM performance, extending the current limitations of sentiment analysis and evaluating the various characteristics of customer sentiment in a more sophisticated manner. The model's resilience to a high level of accuracy and applicability to other domains clearly demonstrates its potential for application across a variety of industries, particularly in the field of financial services, to promptly identify customers' issues and to develop a proper response.

The experiments demonstrate that LFEAR has an outstanding performance while depicting its high precision and accuracy across the different intensities and context factors of sentiment analysis. Overall, the results further demonstrate LFEAR's reliability and its application as a highly beneficial technology for accurate sentiment analysis of consumers' scrollbars, taking into account the essence of their sentiment and promoting strategies. Regarding the role of RAG in particular, this model's ability to dynamically include relevant contextual data provides a cutting-edge method for sentiment classification, making LFEAR a useful tool for businesses that want to improve their customer relations, market awareness, and decision-making. Furthermore, LFEAR's ability to extend domain-specific training with a general approach makes it a unique addition to the field of sentiment analysis. Not only does it address the given research questions in financial sentiment but it also illustrates the effectiveness of modern approaches to sentiment analysis. Over time, LFEAR will be a vital reference point in the study and application of sentiment analysis, both for organisations, as researchers design and implement more effective business models designed to enhance stakeholder engagement, and for overall organisational performance.

Author Contributions: Conceptualisation, M.M., A.M. and V.M.; writing—original draft preparation, A.M.; writing—review and editing, A.M.; funding acquisition, V.M. All authors have read and agreed to the published version of the manuscript.

Funding: This research was sponsored by the funded Data Science for Social Impact (DSFI) Group at the University of Pretoria from Google.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The original contributions presented in the study are included in the article, further inquiries can be directed to the corresponding author.

Acknowledgments: The authors would like to thank the ABSA (through their Chair in Data Science) for their steadfast support and funding of the Data Science for Social Impact (DSFSI) Research Group at the Department of Computer Science. We also acknowledge the support of Google, Google Tensorflow, and OpenAI, whose support made this research possible.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Matikiti, R.; Kruger, M.; Saayman, M. The usage of social media as a marketing tool in two Southern African countries. *Dev. S. Afr.* **2016**, *33*, 740–755. [CrossRef]
2. Kemp, S. Digital 2021: Zimbabwe. *Retrieved* **2021**, *8*, 2021.
3. He, W.; Aboderin, I.; Adjaye-Gbewonyo, D. *Africa Aging*; US Government Printing Office: Washington, DC, USA, 2020.
4. Stremlau, N.; Tsalapatanis, A. Social media, mobile phones and migration in Africa: A review of the evidence. *Prog. Dev. Stud.* **2022**, *22*, 56–71. [CrossRef]
5. Shields, A. Banking and Social Media – Channel, Information and Advice. 2024. Available online: https://rfi.global/banking-and-social-media-channel-information-and-advice/ (accessed on 1 October 2024)
6. Hao, S.; Hoi, S.C.H.; Miao, C.; Zhao, P. Active Crowdsourcing for Annotation. In Proceedings of the 2015 IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology (WI-IAT), Singapore, 6–9 December 2015; Volume 2, pp. 1–8. [CrossRef]
7. Nowak, S.; Rüger, S. How reliable are annotations via crowdsourcing: A study about inter-annotator agreement for multi-label image annotation. In Proceedings of the International Conference on Multimedia Information Retrieval, (MIR'10), New York, NY, USA, 29–31 March 2010; pp. 557–566. [CrossRef]
8. Belal, M.; She, J.; Wong, S. Leveraging ChatGPT As Text Annotation Tool For Sentiment Analysis. *arXiv* **2023**, arxiv.2306.17177. [CrossRef]
9. Fort, K.; Adda, G.; Sagot, B.; Mariani, J.; Couillault, A. Crowdsourcing for Language Resource Development: Criticisms About Amazon Mechanical Turk Overpowering Use. H*uman Language Technology Challenges for Computer Science and Linguistics*; Vetulani, Z.; Mariani, J., Eds., Springer: Cham, Switzerland, 2014; pp. 303–314.
10. Bonta, V.; Kumaresh, N.; Janardhan, N. A comprehensive study on lexicon based approaches for sentiment analysis. *Asian J. Comput. Sci. Technol.* **2019**, *8*, 1–6. [CrossRef]
11. Mathebula, M.; Modupe, A.; Marivate, V. ChatGPT as a Text Annotation Tool to Evaluate Sentiment Analysis on South African Financial Institutions. *IEEE Access* **2024**, *12*, 144017–144043. [CrossRef]
12. Mikolov, T.; Sutskever, I.; Chen, K.; Corrado, G.S.; Dean, J. Distributed representations of words and phrases and their compositionality. *arXiv* **2013**, arXiv:1310.4546.
13. Pennington, J.; Socher, R.; Manning, C.D. Glove: Global vectors for word representation. In Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP), Doha, Qatar, 25–29 October 2014; pp. 1532–1543.
14. Shen, Y.; Liu, J. Comparison of Text Sentiment Analysis based on Bert and Word2vec. In Proceedings of the 2021 IEEE 3rd International Conference on Frontiers Technology of Information and Computer (ICFTIC), Virtual Conference, 12–14 November 2021; pp. 144–147. [CrossRef]
15. Devlin, J.; Chang, M.W.; Lee, K.; Toutanova, K. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. *arXiv* **2019**, arxiv:1810.04805.

16. Garí Soler, A.; Apidianaki, M. Let's play mono-poly: BERT can reveal words' polysemy level and partitionability into senses. *Trans. Assoc. Comput. Linguist.* **2021**, *9*, 825–844. [CrossRef]
17. Zhang, W.; Deng, Y.; Liu, B.; Pan, S.J.; Bing, L. Sentiment analysis in the era of large language models: A reality check. *arXiv* **2023**, arXiv:2305.15005.
18. Inserte, P.R.; Nakhlé, M.; Qader, R.; Caillaut, G.; Liu, J. Large Language Model Adaptation for Financial Sentiment Analysis. *arXiv* **2024**, arXiv:2401.14777.
19. Akhtar, M.S.; Ghosal, D.; Ekbal, A.; Bhattacharyya, P.; Kurohashi, S. All-in-One: Emotion, Sentiment and Intensity Prediction Using a Multi-Task Ensemble Framework. *IEEE Trans. Affect. Comput.* **2022**, *13*, 285–297. [CrossRef]
20. Yu, L.C.; Wu, J.L.; Chang, P.C.; Chu, H.S. Using a contextual entropy model to expand emotion words and their intensity for the sentiment classification of stock market news. *Knowl.-Based Syst.* **2013**, *41*, 89–97. [CrossRef]
21. Addo-Tenkorang, R.; Helo, P.T. Big data applications in operations/supply-chain management: A literature review. *Comput. Ind. Eng.* **2016**, *101*, 528–543. [CrossRef]
22. Zaslavsky, A.; Perera, C.; Georgakopoulos, D. Sensing as a Service and Big Data. *arXiv* **2013**, arxiv:1301.0159.
23. Valdeolmillos, D.; Mezquita, Y.; Ludeiro, A.R. Sensing as a service: An architecture proposal for big data environments in smart cities. In Proceedings of the Ambient Intelligence—Software and Applications, 10th International Symposium on Ambient Intelligence, Ávila, Spain, 17–19 June 2020; pp. 97–104.
24. Shayaa, S.; Jaafar, N.I.; Bahri, S.; Sulaiman, A.; Wai, P.S.; Chung, Y.W.; Piprani, A.Z.; Al-Garadi, M.A. Sentiment analysis of big data: methods, applications, and open challenges. *IEEE Access* **2018**, *6*, 37807–37827. [CrossRef]
25. Yang, Y.; Uy, M.C.S.; Huang, A. Finbert: A pretrained language model for financial communications. *arXiv* **2020**, arXiv:2006.08097.
26. Gite, S.; Khatavkar, H.; Kotecha, K.; Srivastava, S.; Maheshwari, P.; Pandey, N. Explainable stock prices prediction from financial news articles using sentiment analysis. *Peerj Comput. Sci.* **2021**, *7*, e340. [CrossRef]
27. Sharaff, A.; Chowdhury, T.R.; Bhandarkar, S. Lstm based sentiment analysis of financial news. *Comput. Sci.* **2023**, *4*, 584. [CrossRef]
28. Mishev, K.; Gjorgjevikj, A.; Vodenska, I.; Chitkushev, L.T.; Trajanov, D. Evaluation of Sentiment Analysis in Finance: From Lexicons to Transformers. *IEEE Access* **2020**, *8*, 131662–131682. [CrossRef]
29. Liu, J.; Xu, F.; Liu, X. Financial Market Sentiment Analysis and Investment Strategy Formulation Based on Social Network Data. *J. Electr. Syst.* **2024**, *20*, 655–660.
30. Ardekani, A.M.; Bertz, J.; Bryce, C.; Dowling, M.; Long, S. FinSentGPT: A universal financial sentiment engine? *Int. Rev. Financ. Anal.* **2024**, *94*, 103291. [CrossRef]
31. Yue, L.; Chen, W.; Li, X.; Zuo, W.; Yin, M. A survey of sentiment analysis in social media. *Knowl. Inf. Syst.* **2019**, *60*, 617–663. [CrossRef]
32. Kiritchenko, S.; Zhu, X.; Mohammad, S.M. Sentiment analysis of short informal texts. *J. Artif. Intell. Res.* **2014**, *50*, 723–762. [CrossRef]
33. Louviere, J.J.; Woodworth, G.G. *Best-Worst Scaling: A Model for the Largest Difference Judgments*; Technical report, Working paper; Cambridge University Press: Cambridge, UK, 1991.
34. Bradley, M.M.; Lang, P.J. Affective norms for English words (ANEW): Instruction Manual and Affective Ratings. Technical Report, Technical Report C-1, The Center for Research in Psychophysiology. 1999. Available online: https://pdodds.w3.uvm.edu/teaching/courses/2009-08UVM-300/docs/others/everything/bradley1999a.pdf (accessed on 1 October 2024).
35. Årup Nielsen, F. A new ANEW: Evaluation of a word list for sentiment analysis in microblogs. *arXiv* **2011**, arxiv:1103.2903.
36. Patil, A.R.; Kumar, A.; Gamanagati, S.; et al. Brain death: diagnostic clues on imaging. *J. Emergencies Trauma Shock* **2012**, *5*, 372–373.
37. Hutto, C.; Gilbert, E. Vader: A parsimonious rule-based model for sentiment analysis of social media text. In Proceedings of the International AAAI Conference on Web and Social Media, Ann Arbor, MI, USA, 1–4 June 2014; Volume 8, 216–225.
38. Çılgın, C.; Baş, M.; Bilgehan, H.; Ünal, C. Twitter sentiment analysis during covid-19 outbreak with vader. *AJIT Acad. J. Inf. Technol.* **2022**, *13*, 72–89.
39. Newman, H.; Joyner, D. Sentiment analysis of student evaluations of teaching. In Proceedings of the Artificial Intelligence in Education: 19th International Conference, AIED 2018, London, UK, 27–30 June 2018; pp. 246–250.
40. Elbagir, S.; Yang, J. Twitter sentiment analysis using natural language toolkit and VADER sentiment. *Proc. Int. Multiconf. Eng. Comput. Sci.* **2019**, *122*, 16.
41. Jain, R.; Kumar, A.; Nayyar, A.; Dewan, K.; Garg, R.; Raman, S.; Ganguly, S. Explaining sentiment analysis results on social media texts through visualization. *Multimed. Tools Appl.* **2023**, *82*, 22613–22629. [CrossRef]
42. Borg, A.; Boldt, M. Using VADER sentiment and SVM for predicting customer response sentiment. *Expert Syst. Appl.* **2020**, *162*, 113746. [CrossRef]
43. Hu, M.; Liu, B. Mining and summarizing customer reviews. In Proceedings of the Tenth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Seattle, DC, USA, 22–25 August 2004; pp. 168–177.
44. Baccianella, S.; Esuli, A.; Sebastiani, F.; et al. Sentiwordnet 3.0: an enhanced lexical resource for sentiment analysis and opinion mining. In Proceedings of the LREC, Valletta, Malta, 17–23 May 2010; Volume 10, pp. 2200–2204.

45. Moshkin, V.; Yarushkina, N.; Andreev, I. The Sentiment Analysis of Unstructured Social Network Data Using the Extended Ontology SentiWordNet. In Proceedings of the 2019 12th International Conference on Developments in eSystems Engineering (DeSE), Kazan, Russia, 7–10 October 2019; pp. 576–580. [CrossRef]
46. Sadhasivam, J.; Kalivaradhan, R.B. Sentiment analysis of Amazon products using ensemble machine learning algorithm. *Int. J. Math. Eng. Manag. Sci.* **2019**, *4*, 508. [CrossRef]
47. Lee, C.; Kim, K.; Lim, J.; Lee, Y. Psychological research using linguistic inquiry and word count (liwc) and korean linguistic inquiry and word count (kliwc) language analysis methodologies. *J. Cogn. Sci.* **2015**, *16*, 132–49.
48. Onan, A. Sentiment analysis on Twitter based on ensemble of psychological and linguistic feature sets. *Balk. J. Electr. Comput. Eng.* **2018**, *6*, 69–77. [CrossRef]
49. Koutsoumpis, A.; Oostrom, J.K.; Holtrop, D.; Van Breda, W.; Ghassemi, S.; de Vries, R.E. The kernel of truth in text-based personality assessment: A meta-analysis of the relations between the Big Five and the Linguistic Inquiry and Word Count (LIWC). *Psychol. Bull.* **2022**, *148*, 843. [CrossRef]
50. Chen, F.; Li, S.; Lin, L.; Huang, X. Identifying temporal changes in student engagement in social annotation during online collaborative reading. *Educ. Inf. Technol.* **2024**, pp. 1–24.
51. Li, S.; Huang, X.; Zhu, G.; Du, H.; Zhong, T.; Hou, C.; Zheng, J. Exploring behavioural patterns and their relationships with social annotation outcomes. *J. Comput. Assist. Learn.* **2024**, *40*, 1389–1399. [CrossRef]
52. Boyd, R.L.; Ashokkumar, A.; Seraj, S.; Pennebaker, J.W. *The Development and Psychometric Properties of LIWC-22*; University of Texas at Austin: Austin, TX, USA, 2022; Volume 10.
53. Rentoumi, V.; Giannakopoulos, G.; Karkaletsis, V.; Vouros, G.A. Sentiment analysis of figurative language using a word sense disambiguation approach. In Proceedings of the International Conference RANLP-2009, Borovets, Bulgaria, 14–16 September 2009; pp. 370–375.
54. Strapparava, C.; Mihalcea, R. SemEval-2007 Task 14: Affective Text. In *Proceedings of the Fourth International Workshop on Semantic Evaluations (SemEval-2007)*; Agirre, E., Màrquez, L., Wicentowski, R., Eds., Association for Computational Linguistics: Prague, Czech Republic, 2007; pp. 70–74.
55. Jayakrishnan, R.; Gopal, G.N.; Santhikrishna, M. Multi-class emotion detection and annotation in Malayalam novels. In Proceedings of the 2018 International Conference on Computer Communication and Informatics (ICCCI), Tamilnadu, India, 4–6 January 2018; pp. 1–5.
56. Hogenboom, A.; Brojba-Micu, A.; Frasincar, F. The impact of Word Sense Disambiguation on stock price prediction. *Expert Syst. Appl.* **2021**, *184*, 115568. [CrossRef]
57. Augustyniak, L.; Szymański, P.; Kajdanowicz, T.; TuligŁowicz, W. Comprehensive Study on Lexicon-based Ensemble Classification Sentiment Analysis. *Entropy* **2016**, *18*, 4. [CrossRef]
58. Yadav, A.; Vishwakarma, D.K. Sentiment analysis using deep learning architectures: A review. *Artif. Intell. Rev.* **2020**, *53*, 4335–4385. [CrossRef]
59. Modupe, A.; Celik, T.; Marivate, V.; Olugbara, O.O. Post-authorship attribution using regularized deep neural network. *Appl. Sci.* **2022**, *12*, 7518. [CrossRef]
60. Aldoseri, A.; Al-Khalifa, K.N.; Hamouda, A.M. Re-thinking data strategy and integration for artificial intelligence: Concepts, opportunities, and challenges. *Appl. Sci.* **2023**, *13*, 7082. [CrossRef]
61. van Giffen, B.; Herhausen, D.; Fahse, T. Overcoming the pitfalls and perils of algorithms: A classification of machine learning biases and mitigation methods. *J. Bus. Res.* **2022**, *144*, 93–106. [CrossRef]
62. Zhao, P.; Zhang, H.; Yu, Q.; Wang, Z.; Geng, Y.; Fu, F.; Yang, L.; Zhang, W.; Jiang, J.; Cui, B. Retrieval-Augmented Generation for AI-Generated Content: A Survey. *arXiv* **2024**, arXiv:2402.19473.
63. Liu, H.; Liu, H.; Liu, H.; Liu, H.; Yin, Q.; Yin, Q.; Wang, W.Y.; Wang, W.Y. Towards Explainable NLP: A Generative Explanation Framework for Text Classification. *arXiv* **2019**, arXiv:1811.00196. [CrossRef]
64. Gao, Y.; Xiong, Y.; Gao, X.; Jia, K.; Pan, J.; Bi, Y.; Dai, Y.; Sun, J.; Guo, Q.; Wang, M.; et al. Retrieval-Augmented Generation for Large Language Models: A Survey. *arXiv* **2023**, arxiv.2312.10997. [CrossRef]
65. Fan, W.; Ding, Y.; bo Ning, L.; Wang, S.; Li, H.; Yin, D.; Chua, T.S.; Li, Q. A Survey on RAG Meeting LLMs: Towards Retrieval-Augmented Large Language Models. *arXiv* **2024**, arXiv:2405.06211.
66. Hu, Y.; Lu, Y. RAG and RAU: A Survey on Retrieval-Augmented Language Model in Natural Language Processing. *arXiv* **2024**, arXiv:2405.06211.
67. Lewis, P.; Lewis, P.; Perez, E.; Perez, E.; Perez, E.; Piktus, A.; Piktus, A.; Piktus, A.; Petroni, F.; Petroni, F.; et al. Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks. *arXiv* **2020**, arXiv:2005.11401.
68. Zhang, B.; Yang, H.; Zhou, T.; Ali Babar, M.; Liu, X.Y. Enhancing financial sentiment analysis via retrieval augmented large language models. *arXiv* **2023**, arXiv:2310.04027.
69. Shivaprasad, T.K.; Shivaprasad, T.K.; Shetty, J.; Shetty, J.; Shetty, J. Sentiment analysis of product reviews: A review. In Proceedings of the International Conference Inventive Communication and Computational Technologies, Coimbatore, India, 10–11 March 2017. [CrossRef]
70. Zhao, Z.; Welsch, R.E. Aligning LLMs with Human Instructions and Stock Market Feedback in Financial Sentiment Analysis. *arXiv* **2024**, arXiv:2410.14926.

71. Vulic, I.; hao Su, P.; Coope, S.; Gerz, D.; Budzianowski, P.; Casanueva, I.; Mrkvsic, N.; Wen, T.H. ConvFiT: Conversational Fine-Tuning of Pretrained Language Models. *arXiv* **2021**, arXiv:2109.10126. [CrossRef]
72. Alghisi, S.; Rizzoli, M.; Roccabruna, G.; Mousavi, S.M.; Riccardi, G. Should We Fine-Tune or RAG? Evaluating Different Techniques to Adapt LLMs for Dialogue. *arXiv* **2024**, arXiv:2406.06399. [CrossRef]
73. Fütterer, T.; Fischer, C.; Alekseeva, A.; Chen, X.; Tate, T.; Warschauer, M.; Gerjets, P. ChatGPT in education: global reactions to AI innovations. *Sci. Rep.* **2023**, *13*, 15310. [CrossRef]
74. Duan, G.; Yan, S.; Zhang, M. A Hybrid Neural Network Model for Sentiment Analysis of Financial Texts Using Topic Extraction, Pre-Trained Model, and Enhanced Attention Mechanism Methods. *IEEE Access* **2024**, *12*, 98207–98224. [CrossRef]
75. Ling, T.; Chen, L.; Liao, C.; Huang, S.; Yu, Z.; Liu, Y. Improving Aspect Sentiment Classification via Retrieving from Training Data. In Proceedings of the 2023 Asia Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC), Taipei, Taiwan, 31 October–3 November 2023; pp. 490–497.
76. Chen, F.; Ji, R.; Su, J.; Cao, D.; Gao, Y. Predicting microblog sentiments via weakly supervised multimodal deep learning. *IEEE Trans. Multimed.* **2017**, *20*, 997–1007. [CrossRef]
77. Alawwad, H.; Alhothali, A.M.; Naseem, U.; Alkhathlan, A.; Jamal, A. Enhancing Textbook Question Answering Task with Large Language Models and Retrieval Augmented Generation. *arXiv* **2024**, arxiv.2402.05128.
78. Hikov, A.; Murphy, L. Information retrieval from textual data: Harnessing large language models, retrieval augmented generation and prompt engineering. *J. AI Robot. Workplace Autom.* **2024**, *3*, 142–150. [CrossRef]
79. Wu, S.; Xiong, Y.; Cui, Y.; Wu, H.; Chen, C.; Yuan, Y.; Huang, L.; Liu, X.; Kuo, T.W.; Guan, N.; et al. Retrieval-Augmented Generation for Natural Language Processing: A Survey. *arXiv* **2024**, arxiv:2407.13193.
80. Miao, J.; Thongprayoon, C.; Suppadungsuk, S.; Garcia Valencia, O.A.; Cheungpasitporn, W. Integrating retrieval-augmented generation with large language models in nephrology: advancing practical applications. *Medicina* **2024**, *60*, 445. [CrossRef]
81. Long, Q.; Wang, W.; Pan, S. Adapt in Contexts: Retrieval-Augmented Domain Adaptation via In-Context Learning. In Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing, Singapore, 6–10 December 2023; pp. 6525–6542.
82. Pathak, A.; Shree, O.; Agarwal, M.; Sarkar, S.D.; Tiwary, A. Performance Analysis of LoRA Finetuning Llama-2. In Proceedings of the 2023 7th International Conference on Electronics, Materials Engineering & Nano-Technology (IEMENTech), Kolkata, India, 18–20 December 2023; pp. 1–4.
83. Huang, Y.; Huang, J. A Survey on Retrieval-Augmented Text Generation for Large Language Models. *arXiv* **2024**, arxiv:2404.10981.
84. Sudirjo, F.; Diantoro, K.; Al-Gasawneh, J.A.; Azzaakiyyah, H.K.; Ausat, A.M.A. Application of ChatGPT in Improving Customer Sentiment Analysis for Businesses. *J. Teknol. Dan Sist. Inf. Bisnis* **2023**. [CrossRef]
85. Es, S.; James, J.; Espinosa-Anke, L.; Schockaert, S. RAGAS: Automated Evaluation of Retrieval Augmented Generation. In Proceedings of the Conference of the European Chapter of the Association for Computational Linguistics, St. Julians, Malta, 17–22 March 2024. [CrossRef]
86. Yang, L.; Yao, Y.; Ton, J.F.; Zhang, X.; Cheng, R.; Klochkov, Y.; Taufiq, M.F.; Li, H. Trustworthy LLMs: a Survey and Guideline for Evaluating Large Language Models' Alignment. *arXiv* **2023**, arXiv:2308.05374. [CrossRef]
87. Salemi, A.; Zamani, H. Evaluating retrieval quality in retrieval-augmented generation. In Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval, Washington, DC, USA, 14–18 July 2024; pp. 2395–2400.
88. Xia, Y.; Zhou, J.; Shi, Z.; Chen, J.; Huang, H. Improving Retrieval Augmented Language Model with Self-Reasoning. *arXiv* **2024**, arxiv:2407.19813.
89. Chang, Y.; Wang, X.; Wang, J.; Wu, Y.; Yang, L.; Zhu, K.; Chen, H.; Yi, X.; Wang, C.; Wang, Y.; et al. A survey on evaluation of large language models. *ACM Trans. Intell. Syst. Technol.* **2024**, *15*, 1–45. [CrossRef]
90. Balaguer, M.A.D.L.; Benara, V.; Cunha, R.L.D.F.; Filho, R.D.M.E.; Hendry, T.; Holstein, D.; Marsman, J.; Mecklenburg, N.; Malvar, S.; Nunes, L.; et al. RAG vs Fine-tuning: Pipelines, Tradeoffs, and a Case Study on Agriculture. *arXiv* **2024**, arXiv:2401.08406 [CrossRef]
91. Soudani, H.; Kanoulas, E.; Hasibi, F. Fine Tuning vs. Retrieval Augmented Generation for Less Popular Knowledge. *arXiv* **2024**, arXiv:2403.01432 . [CrossRef]
92. Danilevsky, M.; Danilevsky, M.; Dhanorkar, S.; Dhanorkar, S.; Li, Y.; Li, Y.; Popa, L.; Popa, L.; Qian, K.; Qian, K.; et al. Explainability for Natural Language Processing. In *Knowledge Discovery and Data Mining*; Springer: New York; NY, USA, 2021. [CrossRef]
93. Chan, S.W.K.; Chan, S.W.K.; Chong, M.W.C.; Chong, M.W.C. Sentiment analysis in financial texts. *Decis. Support Syst.* **2017**, *54*, 53–64. [CrossRef]
94. Jiang, M.T.J.; Jiang, M.T.J.; Wu, S.H.; Wu, S.H.; Chen, Y.; Chen, Y.K.; Gu, Z.X.; Gu, Z.X.; Chiang, C.J.; Chiang, C.J.; et al. Fine-tuning techniques and data augmentation on transformer-based models for conversational texts and noisy user-generated content. In Proceedings of the International Conference on Advances in Social Networks Analysis and Mining, The Hague, Netherlands, 7–10 December 2020. [CrossRef]
95. Agarwal, P.; Gupta, A. Strategic Business Insights through Enhanced Financial Sentiment Analysis: A Fine-Tuned Llama 2 Approach. In Proceedings of the 2024 International Conference on Inventive Computation Technologies (ICICT), London, UK, 19–22 February 2024; pp. 1446–1453. [CrossRef]
96. Hajek, P.; Munk, M. Speech emotion recognition and text sentiment analysis for financial distress prediction. *Neural Comput. Appl.* **2023**, *35*, 21463–21477. [CrossRef]

97. George, M.; Murugesan, R. Improving sentiment analysis of financial news headlines using hybrid Word2Vec-TFIDF feature extraction technique. *Procedia Comput. Sci.* **2024**, *244*, 1–8. [CrossRef]
98. Gururangan, S.; Marasović, A.; Swayamdipta, S.; Lo, K.; Beltagy, I.; Downey, D.; Smith, N.A. Don't stop pretraining: Adapt language models to domains and tasks. *arXiv* **2020**, arXiv:2004.10964.
99. Ruder, S.; Peters, M.E.; Swayamdipta, S.; Wolf, T. Transfer learning in natural language processing. In Proceedings of the 2019 conference of the North American Chapter of the Association for Computational Linguistics: Tutorials, Minneapolis, MN, USA, 2–7 June 2019; pp. 15–18.
100. Gao, J.; Galley, M.; Li, L. Neural Approaches to Conversational AI. *Found. Trends* **2019**, *13*, 127–298. [CrossRef]
101. Nausheen, F.; Begum, S.H. Sentiment analysis to predict election results using Python. In Proceedings of the 2018 2nd international conference on inventive systems and control (ICISC), Piscataway, NJ, USA, 19–20 January 2018; pp. 1259–1262.
102. Kaur, C.; Sharma, A. Social issues sentiment analysis using python. In Proceedings of the 2020 5th International Conference on Computing, Communication and Security (ICCCS), Shanghai, China, 15–18 May 2020; pp. 1–6.
103. Brown, T.B. Language models are few-shot learners. *arXiv* **2020**, arXiv:2005.14165.
104. Wei, J.; Wang, X.; Schuurmans, D.; Bosma, M.; Xia, F.; Chi, E.; Le, Q.V.; Zhou, D.; et al. Chain-of-thought prompting elicits reasoning in large language models. *Adv. Neural Inf. Process. Syst.* **2022**, *35*, 24824–24837.
105. Yao, S.; Zhao, J.; Yu, D.; Du, N.; Shafran, I.; Narasimhan, K.; Cao, Y. React: Synergizing reasoning and acting in language models. *arXiv* **2022**, arXiv:2210.03629.
106. Reimers, N.; Gurevych, I. Sentence-bert: Sentence embeddings using siamese bert-networks. *arXiv* **2019**, arXiv:1908.10084.
107. Johnson, J.; Douze, M.; Jégou, H. Billion-scale similarity search with GPUs. *IEEE Trans. Big Data* **2019**, *7*, 535–547. [CrossRef]
108. Touvron, H.; Martin, L.; Stone, K.; Albert, P.; Almahairi, A.; Babaei, Y.; Bashlykov, N.; Batra, S.; Bhargava, P.; Bhosale, S.; et al. Llama 2: Open foundation and fine-tuned chat models. *arXiv* **2023**, arXiv:2307.09288.
109. Achiam, J.; Adler, S.; Agarwal, S.; Ahmad, L.; Akkaya, I.; Aleman, F.L.; Almeida, D.; Altenschmidt, J.; Altman, S.; Anadkat, S.; et al. Gpt-4 technical report. *arXiv* **2023**, arXiv:2303.08774.
110. Coyne, S.; Sakaguchi, K.; Galvan-Sosa, D.; Zock, M.; Inui, K. Analyzing the performance of gpt-3.5 and gpt-4 in grammatical error correction. *arXiv* **2023**, arXiv:2303.14342.
111. Dubey, A.; Jauhri, A.; Pandey, A.; Kadian, A.; Al-Dahle, A.; Letman, A.; Mathur, A.; Schelten, A.; Yang, A.; Fan, A.; et al. The llama 3 herd of models. *arXiv* **2024**, arXiv:2407.21783.
112. Strubell, E.; Ganesh, A.; McCallum, A. Energy and policy considerations for modern deep learning research. In Proceedings of the 34th AAAI Conference on Artificial Intelligence (AAAI-20), New York, NY, USA, 7–12 February 2020; Volume 34, pp. 13693–13696.
113. Jegou, H.; Douze, M.; Schmid, C. Product quantization for nearest neighbor search. *IEEE Trans. Pattern Anal. Mach. Intell.* **2010**, *33*, 117–128. [CrossRef]
114. Guo, R.; Sun, P.; Lindgren, E.; Geng, Q.; Simcha, D.; Chern, F.; Kumar, S. Accelerating large-scale inference with anisotropic vector quantization. In Proceedings of the International Conference on Machine Learning, Virtual Event, 13–18 July 2020; pp. 3887–3896.
115. Chen, W.; Ma, X.; Zeng, J.; Duan, Y.; Zhong, G. Hierarchical quantization for billion-scale similarity retrieval on gpus. *Comput. Electr. Eng.* **2021**, *90*, 107002. [CrossRef]
116. Dettmers, T.; Lewis, M.; Belkada, Y.; Zettlemoyer, L. Gpt3. int8 (): 8-bit matrix multiplication for transformers at scale. *Adv. Neural Inf. Process. Syst.* **2022**, *35*, 30318–30332.
117. Hu, E.J.; Shen, Y.; Wallis, P.; Allen-Zhu, Z.; Li, Y.; Wang, S.; Wang, L.; Chen, W. Lora: Low-rank adaptation of large language models. *arXiv* **2021**, arXiv:2106.09685.
118. Perez, E.; Kiela, D.; Cho, K. True few-shot learning with language models. *Adv. Neural Inf. Process. Syst.* **2021**, *34*, 11054–11070.
119. Raffel, C.; Shazeer, N.; Roberts, A.; Lee, K.; Narang, S.; Matena, M.; Zhou, Y.; Li, W.; Liu, P.J. Exploring the limits of transfer learning with a unified text-to-text transformer. *J. Mach. Learn. Res.* **2020**, *21*, 1–67.
120. Abimannan, S.; El-Alfy, E.S.M.; Chang, Y.S.; Hussain, S.; Shukla, S.; Satheesh, D. Ensemble multifeatured deep learning models and applications: A survey. *IEEE Access* **2023**, *11*, 107194–107217. [CrossRef]
121. Bishop, C.M.; Nasrabadi, N.M. *Pattern Recognition and Machine Learning*; Springer: New York, NY, USA 2006; Volume 4.
122. Dietterich, T.G. Ensemble methods in machine learning. In Proceedings of the International Workshop on Multiple Classifier Systems, Cagliari, Italy, 21–23 June 2000; pp. 1–15.
123. Kuncheva, L.I. *Combining Pattern Classifiers: Methods and Algorithms*; John Wiley & Sons: Hoboken, NJ, USA, 2014.
124. Tang, D.; Qin, B.; Liu, T. Aspect Level Sentiment Classification with Deep Memory Network. *arXiv* **2016**, arXiv:1605.08900.
125. Maas, A.; Daly, R.E.; Pham, P.T.; Huang, D.; Ng, A.Y.; Potts, C. Learning word vectors for sentiment analysis. In Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies, 19–24 June 2011, Portland, OR, USA, 2011; pp. 142–150.
126. Bagherzadeh, S.; Shokouhyar, S.; Jahani, H.; Sigala, M. A generalizable sentiment analysis method for creating a hotel dictionary: using big data on TripAdvisor hotel reviews. *J. Hosp. Tour. Technol.* **2021**, *12*, 210–238. [CrossRef]
127. Mohammad, S.M.; Sobhani, P.; Kiritchenko, S. Stance and sentiment in tweets. *ACM Trans. Internet Technol. (TOIT)* **2017**, *17*, 1–23. [CrossRef]
128. Malo, P.; Sinha, A.; Korhonen, P.; Wallenius, J.; Takala, P. Good debt or bad debt: Detecting semantic orientations in economic texts. *J. Assoc. Inf. Sci. Technol.* **2014**, *65*, 782–796. [CrossRef]

129. Lin, T.Y.; Goyal, P.; Girshick, R.; He, K.; Dollár, P. Focal loss for dense object detection. In Proceedings of the IEEE International Conference on Computer Vision, Venice, Italy, 22–29 October 2017; pp. 2980–2988.
130. Buda, M.; Maki, A.; Mazurowski, M.A. A systematic study of the class imbalance problem in convolutional neural networks. *Neural Netw.* **2018**, *106*, 249–259. [CrossRef]
131. Ragas. Metrics. 2024. https://docs.ragas.io/en/latest/concepts/metrics/index.html (accessed on 1 October 2024).
132. Pasarkar, A.P.; Dieng, A.B. Cousins Of The Vendi Score: A Family Of Similarity-Based Diversity Metrics For Science And Machine Learning. *arXiv* **2023**, arXiv:2310.12952.
133. Friedman, D.; Dieng, A.B. The vendi score: A diversity evaluation metric for machine learning. *arXiv* **2022**, arXiv:2210.02410.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.