ASHESI UNIVERSITY

COMPARATIVE ANALYSIS OF OPEN-SOURCE LARGE LANGUAGE MODELS FOR SENTIMENT ANALYSIS AND PROMPT ENGINEERING

UNDERGRADUATE THESIS

B.Sc. Computer Science

Fredrick Kiarie Njoki

2024

ASHESI UNIVERSITY

Comparative Analysis of Open-Source Large Language Models for Sentiment Analysis and Prompt Engineering

UNDERGRADUATE THESIS

Undergraduate Thesis submitted to the Department of Computer Science, Ashesi University, in partial fulfillment of the requirements for the award of Bachelor of Science degree in Computer Science.

Fredrick Kiarie Njoki

August 2024

Declaration

I hereby declare that this undergraduate thesis is the result of my own original work and that no part of it has been presented for another degree in this university or elsewhere.

Candidate's Signature:

.....

Candidate's Name:

.....

Date:

.....

I hereby declare that the preparation and presentation of this thesis were supervised in accordance with the guidelines on supervising of undergraduate thesis laid down by Ashesi University.

Supervisor's Signature:

.....

Supervisor's Name:

.....

Date:

.....

Acknowledgments

First, I would like to thank God for giving me the courage to take up and complete this project.

I would also like to express my profound gratitude to my supervisor, Mr. Sampson Dankyi Asare, for guiding me on this project and supporting me from its beginning, tirelessly and wholly.

In addition, I would like to thank my friends and family for their unwavering support and for encouraging me to step out of my comfort zone and work extra hard on this project.

I am sincerely thankful for the little things that meant a lot and went a long way to make this project successful.

I am immensely indebted to all of them for their unwavering support.

Abstract

Machine Learning models such as Random Forests and Naïve Bayes are trained and used for sentiment analysis. Large language models (LLMs) are currently used in many tasks because of their advanced attention architecture and the large amount of data they have been trained on. Closed source LLMs like ChatGPT, from OpenAI, and Gemini, from Google, are being explored and used for various tasks. However, they do not allow the user to fully interact and fine-tune them for performing specific tasks. Hence, some users opt for open-source LLMs such as BERT and GPT-2 because they can fine-tune them to their desired tasks. Despite their widespread use, they have not been explored fully to find out how they perform against each other in sentiment analysis and prompt engineering. This paper examines how such open-source large language models perform in analyzing sentiments and their responsiveness to prompts, specifically offering more insights and details about the sentiments aside from giving the sentiment's polarity. The study aims to contribute to the knowledge of the effectiveness of the large language models in sentiment analysis and prompt engineering. This will also inform the reader of the choice between the various LLMs for their use.

Keywords:

Machine learning, Artificial Intelligence, large language model, sentiment analysis, prompt engineering

Table of Contents

Chapter 1: Introduction

1.1 Background

Open-source refers to a program whose source code is available for use or modification by users [6]. Large language models are deep learning algorithms that can perform various natural language processing (NLP) tasks [19]. They use transformer models and are trained using massive datasets. Sentiment analysis is the process of analyzing digital text to determine if the emotional tone of the message is positive, negative, or neutral [12]. Prompt engineering is structuring text that can be interpreted and understood by a generative Artificial Intelligence (AI) model [13]. Consider a multinational company such as Amazon. They have many products and serve many customers around the world. They receive feedback and reviews about the products and services they offer. However, because of the number of reviews and feedback they receive, the people working at the customer service would have to spend a lot of time reviewing the feedback.

This is where Machine Learning models come in. Machine Learning models help customer service agents with the tedious work of reviewing customer feedback. The models can analyze emotions to determine the general feedback about a product, whether positive, negative, or neutral. Large language models are trained using large amounts of data. They can determine if feedback is positive, negative, or neutral and give important insights and details from the review. Those details can involve determining which geographical location people mostly complain about a product, among other information. Large language models have wide applications apart from sentiment analysis. Users need to provide instructions to get a response from the large language models. The structuring of the instructions, called prompts, matters for the kind of response the model would provide. For example, given the suitable prompts, an LLM can provide more details about a particular review given by a

user. An Amazon customer service agent can use an LLM to analyze customer reviews and use prompts to generate information like the specific complaint or product mentioned in a review. Despite their wide usage of open-source LLMs, their comparative performance for sentiment analysis and prompt engineering has not been fully explored. This paper seeks to explore that.

1.2 Problem Statement

The research problem is comparing and analyzing the performance of various open-source large language models in analyzing sentiment analysis and how they respond to various prompts. The adoption of large language models is increasing, and companies are incorporating AI models into their businesses. Businesses and people who want to analyze sentiment and determine the emotion or other vital details in the sentiment utilize AI models, especially when they have a lot of sentiments to analyze. Therefore, they need to know the most effective AI models to use, primarily open-source, because of the ease of fine-tuning them. Moreover, as people in various fields continue to use AI models, there is a need to know the proper prompts to give the models to get the desired responses. This paper seeks to address those needs by comparing the performance of various open-source large language models for sentiment analysis and prompt engineering.

1.3 Significance of the Study

This study is significant because readers will know how various open-source large language models perform in analyzing sentiments and how they generate responses given various prompts. That knowledge will enable people to decide on the go-to AI models for tasks such as sentiment analysis because they will know which models perform better. Moreover, readers will know the effectiveness of some open-source large language models in generating responses given different prompts.

1.4 Brief Theoretical/Conceptual Framework

The conceptual framework of the study focuses on sentiment analysis and prompt engineering. The two concepts are derived from machine learning and natural language processing. Knowledge of various fields like computer science is used in machine learning and natural language processing to enable AI models to analyze sentiments. Prompt engineering is used to maximize the capabilities of large language models like chatGPT in generating responses. Integration of these concepts is used in this paper to study the comparative analysis of some open-source large language models in sentiment analysis and their responsiveness to prompts.

1.5 Research Objectives

The main objective of this research is to compare the performance of four open-source large language models for sentiment analysis and two for prompt engineering. The sub-objectives include:

- To fine-tune Distilbert, GPT-2, XLNet, and Electra models for sentiment analysis.
- To evaluate the performance of each of the four models in terms of accuracy, precision, recall, and F1-score metrics.
- To perform prompt engineering using Llama and Mistral models to extract insights from the predicted sentiments.

1.6 Research Questions

This research seeks to answer the following questions:

- How do Distilbert, GPT-2, XLNet, and Electra compare in terms of precision, recall, accuracy, and F1-score when performing sentiment analysis?
- How does the relevance of insights extracted from the predicted sentiments through prompt engineering compare across Llama and Mistral models?

1.7 Expected Contributions

This study will provide insights into the performance of various open-source large language models in sentiment analysis. Readers will know how those models perform and which are better. Furthermore, the study will provide insights into the responsiveness of some open-source large language models given various prompts. When analyzing the responses, the study will also inform readers of some of the best prompt engineering strategies to get desired responses from the AI models.

Chapter 2: Related Work

Mathew and Bindu extensively reviewed natural language processing (NLP) sentiment analysis techniques, focusing on pre-trained models [11]. They highlighted the evolution and effectiveness of pre-trained models such as BERT and RoBERTa. They emphasized their ability to significantly enhance sentiment analysis tasks due to their advanced architectures and training on large datasets. The review emphasized the benefits of using these models, particularly their capacity to capture contextual information and manage complex linguistic nuances, which are critical for accurate sentiment analysis. This study is a foundational reference for understanding the capabilities and advantages of pre-trained models in sentiment analysis [11].

Joshy and Sundar conducted a comparative study to analyze the performance of three prominent pre-trained models, BERT, DistilBERT, and RoBERTa, in sentiment analysis tasks [9]. Their research aimed to provide insights into the efficiency and effectiveness of these models across various sentiment analysis scenarios. They conducted extensive experiments, evaluating the model's performance based on accuracy, precision, recall, and F1-score metrics. The findings revealed that while all three models exhibited high performance, RoBERTa consistently outperformed the others, demonstrating its superior capability in understanding and interpreting sentiment in text. This comparative analysis is crucial for selecting the most suitable model for specific sentiment analysis tasks [9].

Ilmania et al. explored the application of deep neural networks for aspect-based sentiment analysis in the Indonesian language [7]. Their research focused on developing models capable of detecting specific aspects within a text and classifying the sentiment associated with each aspect. They proposed two approaches for solving aspect-based

sentiment analysis. The first one used state-of-the-art text classification using a deep neural network for both modules of aspect-based sentiment analysis: aspect detection and sentiment classification. The second one employed an aspect matrix to rescale the word vector of the input sentence, resulting in the aspect matrix using a dense layer for the bag of words input layer. Both approaches obtained competitive results compared to previous Indonesian aspect-based sentiment analysis research using SVM and rule-based methods [5]. Compared to traditional sentiment analysis methods, the study highlighted the importance of aspect-based sentiment analysis in providing more granular insights into sentiment. This approach is particularly valuable for applications requiring detailed sentiment insights related to specific aspects or features within the text [7].

Clarisó and Cabot proposed applying model-driven engineering to support the prompt engineering process [3]. They introduced a domain-specific language (DSL) called Impromptu to define platform-independent prompts. This DSL enables features such as modular prompts, prompt customization for specific platforms, and prompt documentation. Moreover, it is expressive enough to describe different tasks, like text-to-text or text-to-image.

Jha et al.'s approach to addressing the hallucination challenge can be viewed as an adversarial variant of the in-context learning approach, wherein they used formal verification to detect incorrect responses and include that as a part of the prompt in the dialog with the LLM [8]. Their method does require formal modeling to check the responses of the LLM, but they do not need to model the LLM itself formally. The initial experiments reported during the planning task in their paper were encouraging. They indicated that the proposed combination of LLMs and formal verifiers could alleviate the hallucination problem in LLMs, which is essential for safety-critical applications. This study is particularly relevant for prompt engineering, as it highlights the importance of carefully

structured prompts in obtaining reliable and accurate responses from LLMs. This approach enhances the practical usability of LLMs in various applications, including sentiment analysis [8].

Shaikh et al. investigated the application of large language models for analyzing student feedback sentiment [18]. Their study aimed to understand the effectiveness of LLMs in educational contexts, particularly in processing and interpreting student feedback. They evaluated the performance of various LLMs in sentiment analysis tasks and compared their ability to extract meaningful insights from the feedback. The findings indicated that LLMs, such as BERT and GPT-3, showed promising results in accurately analyzing sentiment and providing actionable insights for educators. This study demonstrates the versatility and applicability of LLMs in different domains, highlighting their potential usage beyond conventional sentiment analysis tasks [18].

The study by Rahmania et al. compares VADER, a lexicon-based sentiment analysis tool, and RoBERTa, a transformer-based deep learning model, in the context of retail sentiment analysis [14]. VADER (Valence Aware Dictionary and sEntiment Reasoner) relies on a pre-defined list of sentiment-laden words and rules to determine sentiment scores. It is particularly effective for short texts and social media content where the sentiment is often expressed straightforwardly. In contrast, RoBERTa (A Robustly Optimized BERT Pretraining Approach) leverages deep learning techniques to understand contextual nuances and handle more complex sentiment expressions. Rahmania et al. demonstrated that RoBERTa outperforms VADER in extracting detailed sentiment insights from customer reviews, capturing subtleties that VADER might miss. This comparison highlights the efficacy of transformer-based models in understanding and analyzing sentiments in retail environments, aligning with the project's goal to evaluate advanced models for sentiment analysis.

Daulatkar and Deore explored sentiment analysis related to online learning success in India during the COVID-19 pandemic [3]. Their study utilized sentiment analysis tools to track changes in attitudes toward online education as the pandemic progressed. Analyzing feedback from students and educators revealed shifts from initial skepticism to increased acceptance as stakeholders adapted to new learning modalities. The findings emphasize the dynamic nature of sentiment, especially in response to global events. This research is relevant to understanding how sentiment analysis can capture evolving opinions responding to significant changes, such as those seen during the pandemic. It enhances the importance of adapting sentiment analysis techniques to reflect real-time changes in public opinion. This factor can improve the accuracy and applicability of sentiment analysis in different contexts.

Tran and Matsui focused on public opinion mining using large language models (LLMs) to analyze COVID-19-related tweets [20]. Their study utilized models such as GPT-3 to process and analyze large volumes of social media data, extracting meaningful sentiment patterns and trends. The ability of LLMs to handle vast datasets and understand context has made them powerful tools for sentiment analysis. This research highlights the advantages of LLMs in managing extensive and diverse text data, providing real-time insights into public sentiment. For the current project, using LLMs like LLaMA and Mistral for prompt engineering and sentiment analysis aligns with the need to handle large and complex datasets, offering a robust approach to understanding and categorizing sentiment.

Basiri and Kabiri addressed the challenges of sentence-level sentiment analysis in Persian, focusing on the linguistic and syntactic features unique to the Persian language [2]. Their study developed models tailored to Persian, improving sentiment classification accuracy by incorporating language-specific characteristics. The research by Basiri and Kabiri is significant for understanding how language-specific features affect sentiment

analysis [2]. Adapting sentiment analysis techniques to different languages involves addressing unique linguistic elements, which is crucial for ensuring accurate analysis across various languages. This insight is valuable for projects aiming to develop or refine models for sentiment analysis in multiple languages.

Li et al. proposed a sentiment information-based Chinese text sentiment analysis model [10]. Their approach integrates sentiment lexicons with machine-learning algorithms to enhance classification accuracy. By combining linguistic resources with advanced modeling techniques, Li et al. improved the performance of sentiment analysis for Chinese texts. This research emphasizes incorporating linguistic and cultural nuances into sentiment analysis models. For the current project, understanding how sentiment information can be integrated into models for different languages provides a foundation for developing more accurate and contextually aware sentiment analysis systems.

Zhang examined the application of deep learning technologies in Japanese sentiment analysis [21]. The study utilized neural network-based models to capture complex sentiment patterns in Japanese text, demonstrating the effectiveness of deep learning in handling language-specific challenges. Zhang's findings highlight the benefits of deep learning models in analyzing texts with intricate syntactic and semantic structures. This aligns with the project's focus on leveraging advanced models like transformers and neural networks to improve sentiment analysis accuracy and handle diverse linguistic features.

Ramanathan and Meyyappan conducted a sentiment analysis on Twitter feedback related to tourism in Oman [15]. Their study employed text-mining techniques to analyze tourists' opinions, providing insights into satisfaction levels and improvement areas in tourism services. They examined the effect of four factors to determine the sentiment analysis of tweets about Oman tourism. The factors were domain-specific ontology, entity-

specific opinion extraction, combined lexicon-based approach, and conceptual semantic analysis. They found out that the new approach, including conceptual semantic sentiment analysis, expressively improves the performance of sentiment analysis. This research illustrates the practical applications of sentiment analysis in the tourism industry, demonstrating how it can enhance customer experience and service quality. For the current project, understanding how sentiment analysis can be applied to real-world scenarios, such as tourism, highlights the versatility and impact of these techniques across different domains.

Rohani and Shavaa introduced the SentiRobo approach, which utilizes machine learning techniques for sentiment analysis [16]. Their work demonstrated the potential of machine learning models, such as support vector machines and ensemble methods, to handle diverse sentiment analysis tasks effectively. The SentiRobo approach emphasizes the advantages of machine learning in managing complex and large-scale sentiment analysis problems. This aligns with the project's goal to explore and evaluate various models for sentiment analysis, showcasing the potential of machine learning techniques to enhance accuracy and performance.

Though the researchers have explored sentiment analysis using various machine learning models, analyzing the performance of some open-source large language models and comparing them is necessary. Moreover, the researchers examined prompt engineering strategies to obtain desired responses from AI models such as ChatGPT. However, there is a need to compare the performance of various open-source large language models in prompt engineering. This paper explores sentiment analysis and prompt engineering more, focusing on DistilBERT, GPT-2, XLNet, and ELECTRA models for sentiment analysis, as well as LLaMA and Mistral for prompt engineering.

Chapter 3: Methodology

3.1 Model Selection

Six open-source large language models were chosen for the project: four for sentiment analysis and two for prompt engineering. The models were selected based on their architecture and capabilities. The models for sentiment analysis are DistiBERT, GPT-2, XLNet, and ELECTRA. The models for prompt engineering are LLaMA and Mistral.

3.1.1 DistilBERT

DistlBERT is a smaller, faster, and lighter version of BERT (Bidirectional Encoder Representations from Transformers). As a streamlined and distilled variant of BERT, DistilBERT emerges as a compelling alternative to most transformers [21]. It retains 97% of BERT's language understanding while being 60% faster and 40% smaller [17]. DistilBERT is well-suited for sentiment analysis due to its efficiency and performance.

3.1.2 GPT-2

GPT-2 (Generative Pre-trained Transformer 2) is an autoregressive language model developed by OpenAI. OpenAI researchers trained GPT-2 on a massive 40GB dataset called WebText. The GPT-2 small model occupies 500MB of storage, needs 1 vCPU and 2GB of memory, and is suited for smaller datasets [23]. GPT-2 can generate coherent and contextually relevant text and be adapted for classification tasks like sentiment analysis.

3.1.3 XLNet

XLNet is an advanced version of BERT that builds on the Transformer-XL model to enhance its ability to understand linguistic contexts. It introduces improvements like Recurrence Mechanism and Relative Positional Encoding (RPE) to handle long-term dependencies and provide context over extended sequences. XLNet combines bidirectional

context with autoregressive modeling to overcome BERT's limitations, allowing for more accurate and comprehensive predictions [24].

3.1.4 ELECTRA

ELECTRA (Efficiently Learning and Encoder that Classifies Token Replacements Accurately) is a natural language processing model that surpasses BERT by predicting the identities of all tokens in the input rather than just the masked tokens. It requires 25% less computational resources than BERT, making it highly suitable for industrial applications such as hiring, recruitment, chatbots, and targeted advertising [25]. ELECTA is designed to be more efficient in training and inference. It uses a discriminative pretraining method, distinguishing between real input tokens and plausible but synthetically generated tokens. This efficiency and accuracy make ELECTRA a strong candidate for sentiment analysis.

3.1.5 LLaMA (Meta-Llama-3-8B-Instruct)

LLaMA (Large Language Model Meta AI) is known for its robust architecture and ability to generate high-quality text. It is beneficial for prompt engineering, where precise and contextually relevant outputs are crucial. Meta-Llama-3-8B-Instruct is an 8 billion parameter language model developed by Meta that is optimized for dialogue use cases and outperforms many open-source chat models [26].

3.1.6 Mistral (mistralai/Mistral-7B-Instruct-v0.1)

Mistral-7B-Instruct-v0.1 is an instruction-tuned variant of the Mistral 7B model designed to enhance its performance in conversational and task-oriented contexts [27]. It has 7 billion parameters and is engineered for superior performance and efficiency. Mistral is designed to handle complex language tasks with high efficiency and accuracy. Its generating and understanding text capabilities make it ideal for prompt engineering applications.

3.2 Dataset Acquisition

For this project, the Amazon Review Polarity Dataset from Kaggle was selected [1]. This dataset is derived from the Stanford Network Analysis Project (SNAP) and contains a substantial volume of Amazon reviews, offering a comprehensive foundation for sentiment analysis.

The dataset comprises 34,686,770 reviews sourced from 6,643,669 users on 2,441,053 products. These reviews cover 18 years, up to March 2013, and include product and user information, ratings, and plaintext reviews.

J. McAuley and J. Leskovec originally compiled the Amazon reviews dataset for their research on understanding rating dimensions through review text, as detailed in their paper: "Hidden factors and hidden topics: understanding rating dimensions with review text."

The dataset is structured to facilitate polarity sentiment analysis by categorizing review scores of 1 and 2 as negative and 4 and 5 as positive. In contrast, reviews with a score of 3 are excluded from the dataset. This binary classification approach results in two classes:

- Class 1: Negative reviews (review scores of 1 or 2)
- Class 2: Positive reviews (review scores of 4 or 5)

The dataset is provided in two main files: 'train.csv' and 'test.csv.' Each file contains the following columns:

- Polarity: indicates the review's sentiment, with 1 for negative and 2 for positive.
- Title: The title or heading of the review.
- Text: The body of the review.

For this project, the primary dataset used was contained in the 'test.csv' file. Half of the reviews from the 'test.csv' file were randomly selected to create a balanced and unbiased sample for sentiment analysis. The rationale behind this randomization was to ensure that the subset of the data used for training and testing the sentiment analysis models represented the entire dataset. This approach helps reduce selection bias and enhances the model's generalizability. The figure below shows the counts of the dataset.

```
sentiment
positive    100076
negative     99923
Name: count, dtype: int64
```

Figure 3.2.1: Dataset count

The dataset is balanced with a difference of 153 reviews between the positive and negative reviews. Figure 3.2.2 below visually shows the distribution of the dataset.

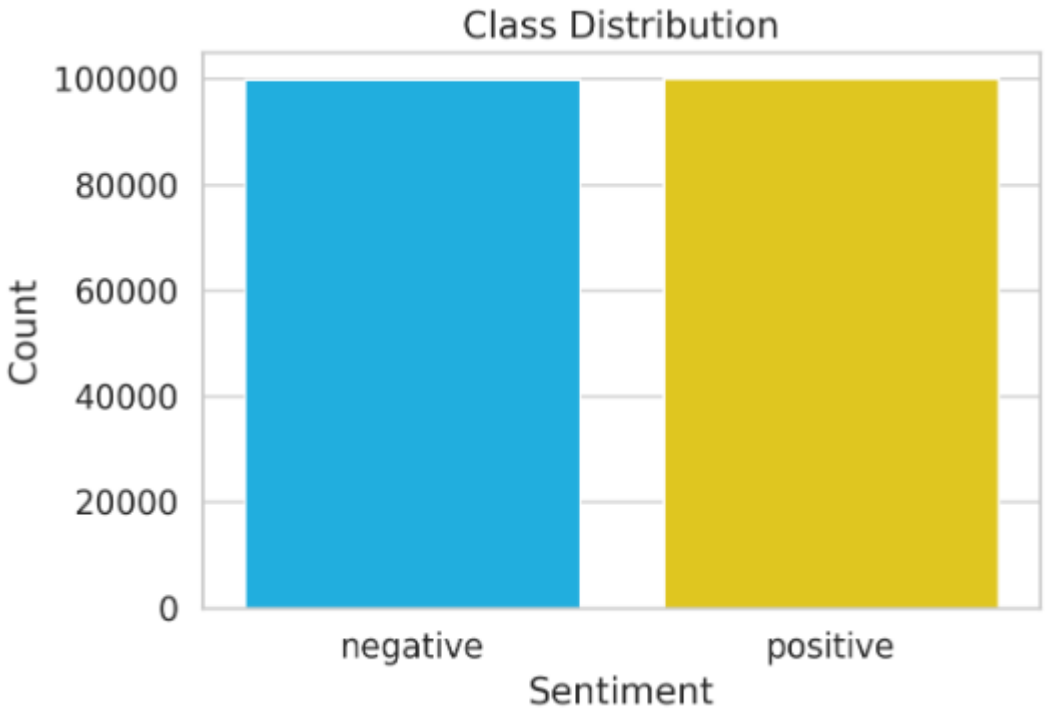

Fig 3.2.2: Bar graph showing the class distribution of the reviews

3.3 Data Preprocessing

Several preprocessing tasks were done to prepare the Amazon Review Polarity Dataset for sentiment analysis and prompt engineering. The dataset was inspected for null values in the polarity, title, and text columns to ensure data integrity. Any records with missing values were removed to maintain consistency. The sentiment values were adjusted and mapped to ensure clarity and consistency. The polarity column, which initially used 1 for negative and 2 for positive sentiments, was adjusted to 0 and 1, respectively. Each review was tokenized, breaking the text into individual tokens (words or subwords). This step was crucial for converting the text data into a format that the large language models can process.

Understanding the distribution of text lengths in the reviews was crucial for deciding the appropriate maximum sequence length for tokenization. This distribution was visualized using a histogram.

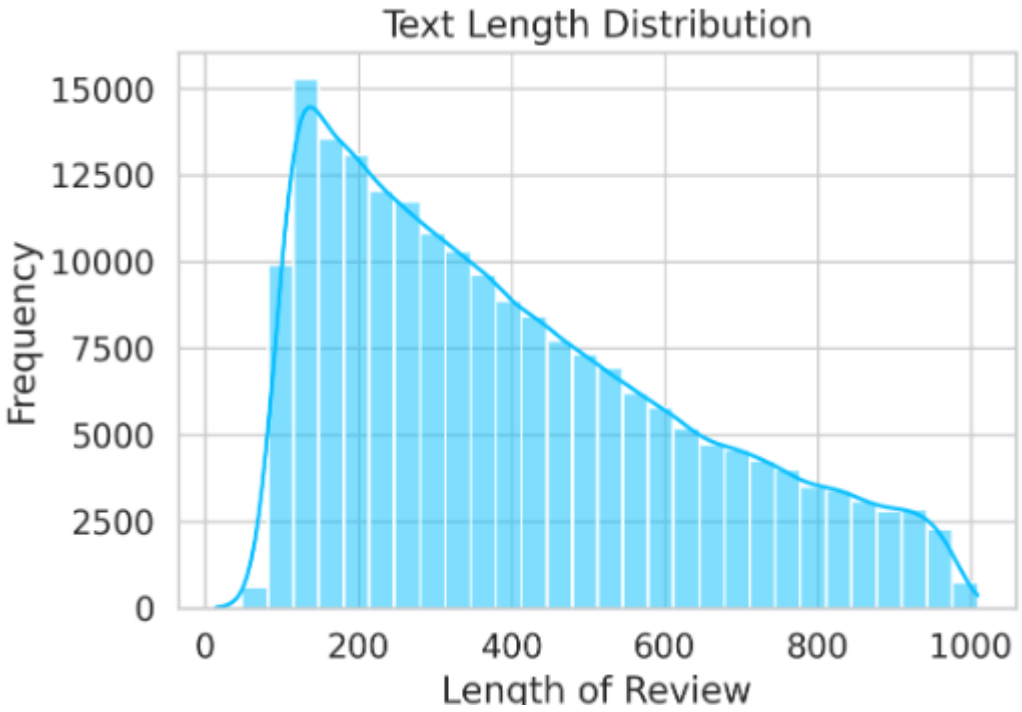

Fig 3.3.1: Histogram showing the text length distribution of the reviews

The histogram in Figure 3.3.1 above illustrates the dataset's review length distribution. The x-axis represents the length of the reviews, measured in terms of the number of characters, while the y-axis indicates the frequency of reviews for each length interval. From the histogram, it can be observed that the most common review length is approximately 150 characters. This peak suggests that many reviews are concise and to the point. The distribution is right-skewed, with a long tail extending towards longer review lengths. This indicates that while most reviews are relatively short, some are significantly longer. Most reviews fall within the range of 0 to 500 characters, with a gradual decline in frequency as the length increases beyond this range.

Understanding the distribution of review lengths is crucial for deciding on the maximum sequence length ('MAX_LEN') for tokenization. Setting the 'MAX_LEN' to around 256 characters based on the histogram could balance capturing sufficient content from each review and ensuring computational efficiency. Reviews longer than this limit can be truncated, while shorter ones can be padded to match the required length, ensuring uniform input sizes for the model.

A word cloud was generated to visualize the most frequent words in the reviews, providing an overview of the dataset's content.

Fig 3.3.2: Word cloud showing the most frequent words in the reviews

The word cloud in Figure 3.3.2 above provides a visual representation of the most frequently occurring words in the dataset of reviews. In this visualization, the size of each word indicates its frequency in the text data – the larger the word, the more often it appears. Words like 'book,' 'one,' 'read,' 'product,' and 'use' appear prominently in the word cloud. This suggests that a significant portion of the reviews are related to books, movies, and products. The frequent occurrence of words like 'love,' 'good,' 'great,' 'buy,' and 'make' indicates that the reviews often discuss the quality and purchasing experience of the items. Terms like 'use,' 'find,' 'work,' and 'recommend' suggest that many reviews are action-oriented, providing insights into how products are used and whether they are recommended. Words such as 'good,' 'great,' 'love,' and 'problem' can indicate sentiment, giving an initial impression of the overall tone of the reviews.

The word cloud is valuable for quickly identifying the dataset's main topics and recurring themes. For sentiment analysis, these insights can help understand the context in which certain words are used, aiding in more accurate sentiment classification. Frequently

occurring words can be given special attention when developing features for the sentiment analysis model, as they are likely to influence the model's predictions significantly.

The tokenized text was then encoded into input_ids and attention masks, which are required inputs for transformer-based language models.

- 'input_ids' represents the tokenized input text converted into numerical IDs.
- 'attention masks' indicate which tokens should be attended to by the model (1 for tokens to be attended to, and 0 for padding tokens).

To ensure uniform input lengths, the tokenized sequences were padded or truncated to a maximum length of 256 tokens (MAX_LEN = 256). Padding shorter sequences with zeros and truncating longer sequences ensure that all inputs have the same length, which is necessary for batch processing in model training.

3.4 Sentiment Analysis

After selecting the models, choosing the dataset, and preprocessing the data, the sentiment analysis was done with the pre-trained models, and the results were recorded. The results provided a benchmark for fine-tuning to see how the models' performance would increase or decrease. The goal was to classify the sentiments of the Amazon reviews into positive or negative categories. The models were evaluated using standard metrics for classification tasks: precision, recall, F1-Score, and accuracy. Afterward, the models were fine-tuned and re-evaluated. Lastly, the performance of the four models before and after fine-tuning were compared.

3.5 Prompt Engineering

After performing sentiment analysis, more insights were extracted from the predicted reviews. LLaMA and Mistral were used for this task. Predicted reviews were

extracted at random and fed into the models. The below prompts were used to extract themes and the reasons behind the classification of a particular review as either positive or negative:

- List the two main topics discussed in the review.
- Explain briefly in one sentence why the review is positive/negative.

Since the models are not as robust and efficient as closed-source models such as ChatGPT, the prompts explicitly indicated whether the review being analyzed is positive or negative. This was because the concern was to get insights from the review because knowing the polarity of the reviews was already done by the models used for sentiment analysis. Hence, the task was for LLaMA and Mistral to provide insights on why the reviews are positive/negative.

3.6 Experimental Setup

The experimental setup outlines the environment and configuration used to conduct the experiments for sentiment analysis and prompt engineering.

3.6.1 Hardware and Software

For sentiment analysis, the experiments were conducted in Kaggle Notebooks using Kaggle GPUs to handle the computational load of training and inference. The implementation used Python and popular Machine Learning libraries such as TensorFlow, Pytorch, and the Hugging Face Transformers library.

3.6.2 Sentiment Analysis Setup

For sentiment analysis, the pre-trained model layers were frozen, and a classification layer with an output size of 2 (representing negative and positive classes) was introduced. The models were first tested before fine-tuning. Only the classification layer was trained during fine-tuning, while other layers remained frozen. The batch size for training and

evaluation was set to 16. After evaluating the models' prediction, five predicted reviews for each class were randomly selected and saved to CSV files for prompt engineering testing.

3.6.3 Prompt Engineering Setup

LLaMA and Mistral models were initialized using the Lamini library. The Lamini API key was then set to access model functionalities. Positive and negative reviews were loaded from CSV files. Using the models, custom functions were defined to generate themes and sentiment drivers from the reviews. The models processed several reviews from the CSV files, extracting and displaying themes and sentiment drivers for each review.

Chapter 4: Results

4.1 Sentiment Analysis

In this section, the results of the sentiment analysis using the pre-trained and fine-tuned models are presented. The evaluation metrics used include precision, recall, F1-Score, and accuracy.

4.1.1 DistilBERT Results

Figure 4.1.1.1 below shows the classification report for the pre-trained DistilBERT.

	precision	recall	f1-score	support
negative	0.69	0.00	0.00	14948
positive	0.50	1.00	0.67	15052
accuracy			0.50	30000
macro avg	0.59	0.50	0.33	30000
weighted avg	0.59	0.50	0.34	30000

Figure 4.1.1.1: Classification report for pre-trained DistilBERT model

The classification report for the pre-trained DistilBERT model demonstrates significant imbalances in its performance between the negative and positive classes. The model exhibits a precision of 0.69 for the negative class, suggesting that it is correct 69% of the time when it predicts a negative sentiment. However, the recall for the negative class is 0.00, indicating a complete failure to correctly identify any actual negative instances. Consequently, the F1-score, the harmonic mean of precision and recall, is also 0.00 for the negative class.

In contrast, the positive class shows a precision of 0.50 and a perfect recall of 1.00, meaning the model identifies all positive instances correctly but at the cost of many false positives. The F1-score for the positive class is 0.67, highlighting the disparity in class performance. Overall, the model achieves an accuracy of 0.50, reflecting its tendency to

predict the majority class (positive) regardless of the actual sentiment. Both the macro and weighted averages for precision, recall, and F1-score are below satisfactory levels (0.59 for precision, 0.50 for recall, and around 0.33-0.34 for F1-score), indicating an overall imbalance in prediction capabilities and highlighting the need for further refinement or fine-tuning.

Figure 4.1.1.2 below shows the confusion matrix for the pre-trained DistilBERT.

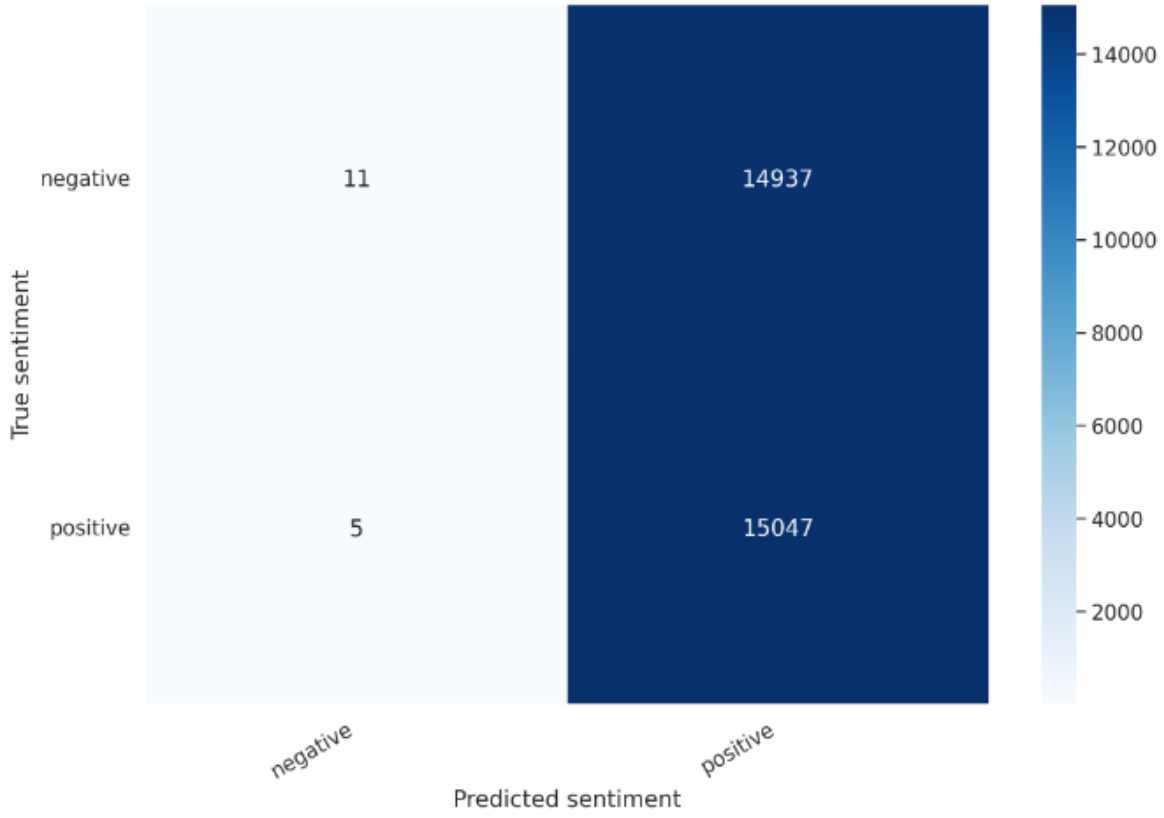

Figure 4.1.1.2: Confusion matrix of pre-trained DistilBERT

The confusion matrix for the pre-trained DistilBERT model further illustrates the issues highlighted in the classification report. It shows that out of 14,948 actual negative instances, the model correctly identified only 11, resulting in a substantial number of false negatives (14,937). The model correctly identified 15,047 out of 15,052 positive instances for the positive class, with only 5 false negatives. The distribution emphasizes the model's bias toward predicting positive sentiment, leading to many misclassifications in the negative

class. The imbalance is visually apparent, as most negative instances are incorrectly classified as positive, emphasizing the need for model adjustments.

```
Review text: Your services are good. Well done!
Sentiment  : positive
Confidence : 56.84%
```

Figure 4.1.1.3: Raw sentiment prediction using pre-trained DistilBERT

Figure 4.1.1.3 above shows the DistilBERT predicting a positive sentiment. The model correctly makes the prediction.

	precision	recall	f1-score	support
negative	0.83	0.88	0.85	14948
positive	0.87	0.83	0.85	15052
accuracy			0.85	30000
macro avg	0.85	0.85	0.85	30000
weighted avg	0.85	0.85	0.85	30000

Figure 4.1.1.4: Classification report for fine-tuned DistilBERT

The figure above, Figure 4.1.1.4, shows the classification report of the fine-tuned DistilBERT model. The classification report for the fine-tuned DistilBERT model shows marked improvements in performance across both sentiment classes compared to the pre-trained version. The negative class now has a precision of 0.83, indicating that when the model predicts negative sentiment, it is correct 83% of the time. Recall for the negative class has improved dramatically to 0.88, meaning the model successfully identifies 88% of actual negative instances. This leads to an F1-score of 0.85, reflecting a balanced performance between precision and recall.

Similarly, the positive class demonstrates strong performance with a precision of 0.87 and a recall of 0.83, resulting in an F1-score of 0.85. The improvements in both classes indicate a more reliable and balanced prediction capability. Overall accuracy has

significantly increased to 0.85, showing that fine-tuning has successfully enhanced the model's general predictive power. The macro and weighted averages for precision, recall, and F1-score are consistently at 0.85, demonstrating a well-rounded and effective sentiment classification.

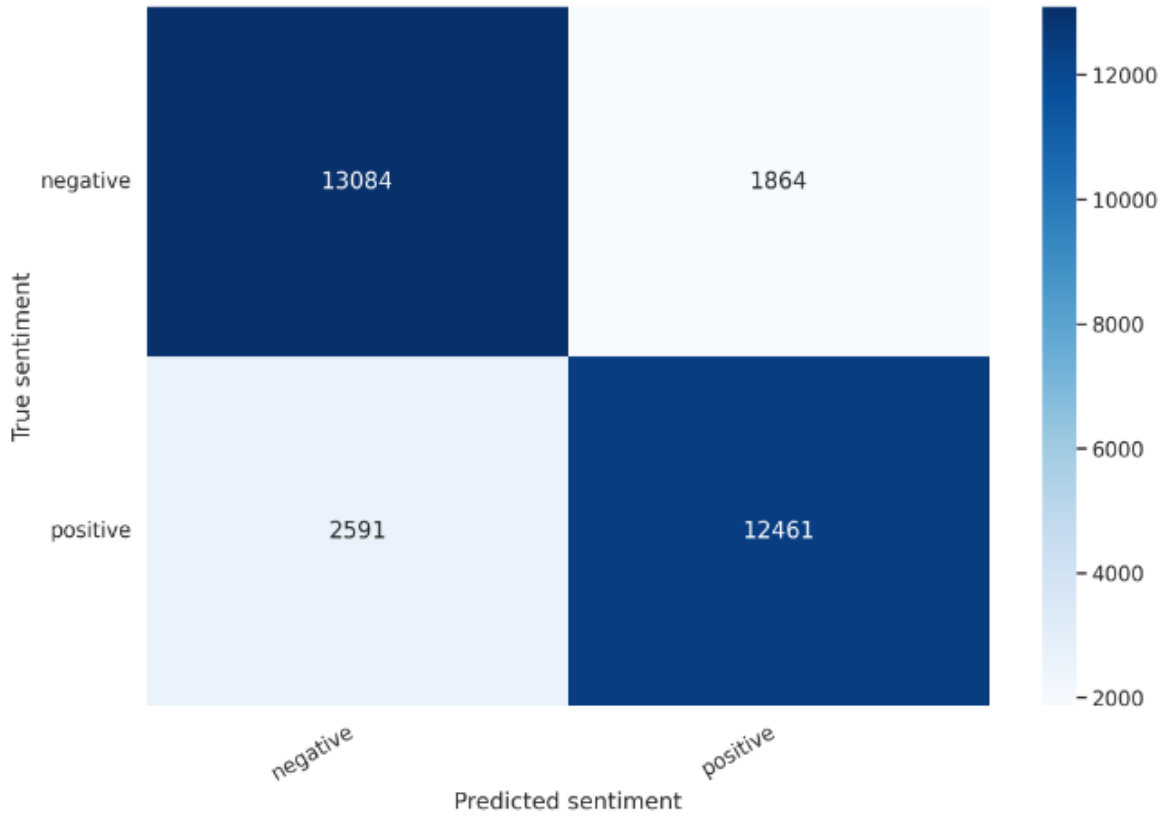

Figure 4.1.1.5: Confusion matrix for fine-tuned DistilBERT

The confusion matrix in Figure 4.1.1.5 for the fine-tuned DistlBERT model shows significant improvements in sentiment classification compared to the pre-trained version. The model correctly identifies 13,084 out of 14,948 negative instances, indicating a strong ability to recognize negative sentiments with few false positives (1,864). Similarly, it accurately classifies 12,461 out of 15,052 positive instances, reflecting the effective detection of positive sentiments despite some false negatives (2,591). This balanced performance demonstrates that fine-tuning has enhanced the model's accuracy and reliability in sentiment analysis.

```
Review text: I got my fries very late from your shop. I am very sad. You should improve your services.
Sentiment  : negative
Confidence : 55.91%
```

Figure 4.1.1.6: Raw sentiment prediction using fine-tuned DistilBERT

Figure 4.1.1.6 above shows the fine-tuned DistilBERT correctly predicting a negative sentiment.

4.1.2 GPT-2 Results

Figure 4.1.2.1 below shows the classification report for the pre-trained GPT-2 model.

	precision	recall	f1-score	support
negative	0.50	1.00	0.67	14948
positive	0.00	0.00	0.00	15052
accuracy			0.50	30000
macro avg	0.25	0.50	0.33	30000
weighted avg	0.25	0.50	0.33	30000

Figure 4.1.2.1: Classification report for pre-trained GPT-2 model

The pre-trained GPT-2 model exhibits significant imbalances in its sentiment classification performance. The model achieves a precision of 0.50 for the negative class, indicating that half of its negative predictions are correct. The recall is 1.00, showing that the model identifies all negative instances correctly. Consequently, the F1-score for the negative class is 0.67, reflecting a reasonable balance between precision and recall for negative sentiments.

However, the model performs poorly on the positive class, with precision, recall, and F1-score all at 0.00. This indicates that the model fails to correctly identify any positive instances, misclassifying all of them as negative. As a result, the model's overall accuracy is 0.50, suggesting a bias toward predicting negative sentiments and highlighting a

significant need for further training or fine-tuning to achieve balanced performance across both sentiment classes.

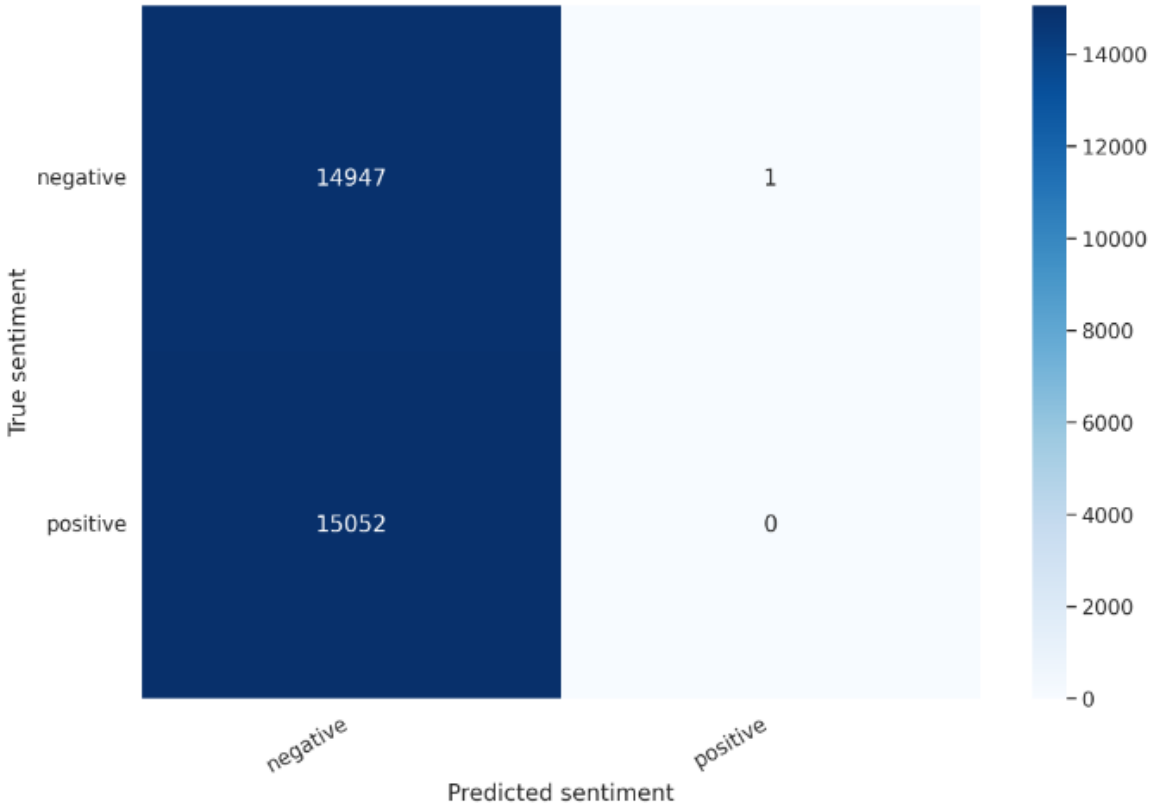

Figure 4.1.2.2: Confusion matrix for pre-trained GPT-2

The confusion matrix in Figure 4.1.2.2 reveals a class imbalance in GPT-2's predictions. Out of 14,948 actual negative instances, the model accurately predicted 14,947 as negative, resulting in a very high true negative rate. However, the model failed to identify any positive instances, with all 15,052 actual positive cases predicted as negative. This highlights a significant performance issue for the positive class, suggesting that the model may be biased towards the negative class.

```
Review text: Your services are good. Well done!
Sentiment  : negative
Confidence : 99.23%
```

Figure 4.1.2.3: Raw sentiment prediction using pre-trained GPT-2 model

Figure 4.1.2.3 above shows the prediction of a positive sentiment by the pre-trained GPT-2. As can be seen, the model predicts the sentiment as negative. This further shows how the model is biased towards predicting negative sentiments.

	precision	recall	f1-score	support
negative	0.84	0.80	0.82	14948
positive	0.81	0.85	0.83	15052
accuracy			0.83	30000
macro avg	0.83	0.83	0.83	30000
weighted avg	0.83	0.83	0.83	30000

Figure 4.1.2.4: Classification report for fine-tuned GPT-2

Figure 4.1.2.4 above shows the classification report for the fine-tuned GPT-2 model. Following fine-tuning, the GPT-2 model demonstrated substantial improvements in performance across classes. For the negative class, the model achieved a precision of 0.84, recall of 0.80, and f1-sore 0.82. These metrics indicate a strong capability to identify negative instances while balancing precision and recall. For the positive class, the model showed even more impressive results with a precision of 0.81, recall of 0.85, and an f1-score of 0.83. This suggests that the fine-tuning process significantly enhanced the model's ability to identify positive instances correctly. The model's accuracy improved to 0.83, reflecting its effectiveness in distinguishing between classes.

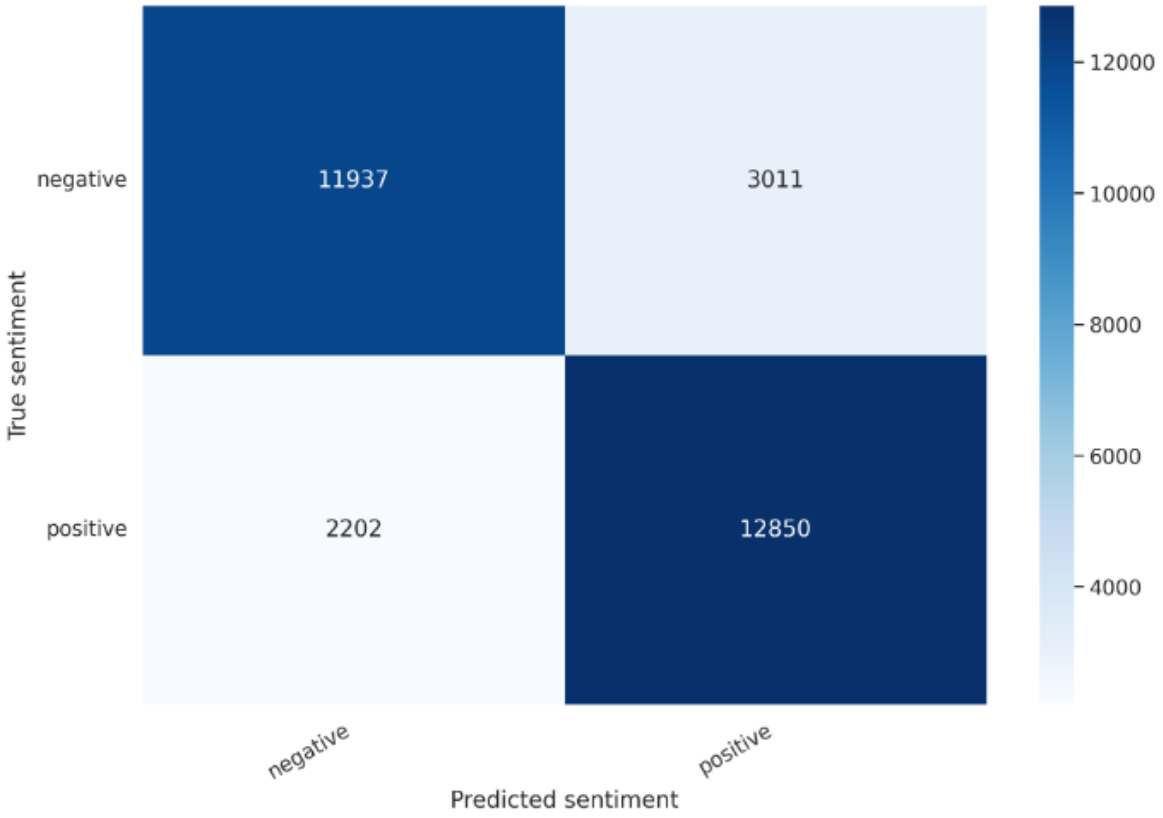

Figure 4.1.2.5: Confusion matrix for fine-tuned GPT-2

The confusion matrix for the fine-tuned GPT-2 shown in Figure 4.1.2.5 shows the model's strong performance across both classes. Of the 14948 negative instances, the model correctly predicted 11,937. Of the 15,052 positive instances, the model correctly predicted 12,850. This shows the model's ability to distinguish between the two classes.

```
Review text: I got my fries very late from your shop. I am very sad. You should improve your services.
Sentiment  : positive
Confidence : 55.57%
```

Figure 4.1.2.6: Raw sentiment prediction using fine-tuned GPT-2

The results in Figure 4.1.2.6 show the fine-tuned model predicting a negative sentiment. The model predicted it as positive with a 55.57% confidence. This indicates that the model is unsure of the sentiment's polarity.

4.1.3 XLNet Results

	precision	recall	f1-score	support
negative	0.49	0.29	0.36	14948
positive	0.50	0.71	0.58	15052
accuracy			0.50	30000
macro avg	0.50	0.50	0.47	30000
weighted avg	0.50	0.50	0.47	30000

Figure 4.1.3.1: Classification report for pre-trained XLNet model

Figure 4.1.3.1 above shows the classification report for the pre-trained XLNet model. The evaluation of the pre-trained model reveals varying performance between the negative and positive classes, with an overall accuracy of 0.50. For the negative class, the model achieved a precision of 0.48, a recall of 0.29, and an f1-score of 0.36. These metrics indicate a significant challenge in correctly identifying negative instances, as the model struggles with low recall, suggesting it misses many negative cases.

In contrast, the positive class showed better performance, with a precision of 0.50, a recall of 0.71, and an F1 score of 0.58. This indicates that while the model is more adept at identifying positive instances, it still exhibits limitations, particularly in maintaining a balanced trade-off between precision and recall. The overall accuracy of 0.50 suggests that the model's predictions are not significantly better than random guessing.

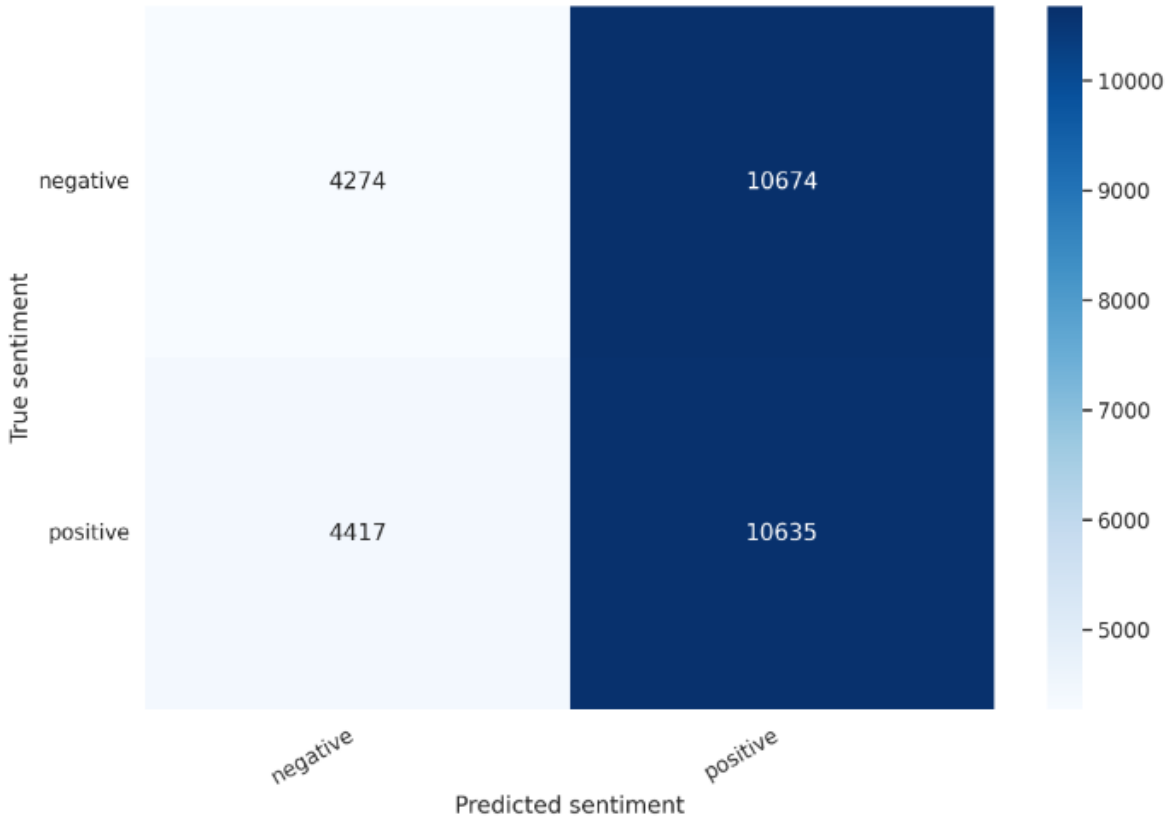

Figure 4.1.3.2: Confusion matrix for pre-trained XLNet model

The evaluation of the pre-trained XLNet model through its confusion matrix shown in Figure 4.1.3.2 reveals a significant disparity in performance between the negative and positive classes. For the negative class, the model correctly predicted 4,274 out of 14,948 instances as negative. This results in a low accuracy for the negative class, as the model misclassified 10,674 negative instances as positive. This highlights the model's struggle to correctly identify negative cases, which aligns with the previously reported low recall of 0.29. The low recall indicates that the model fails to detect many actual negative instances, leading to a high rate of false positives.

On the other hand, the positive class performs better, with the model correctly predicting 10,635 out of 15,052 positive instances. This suggests that the model is more effective at identifying positive instances, which is supported by a higher recall of 0.71.

```
Review text: Your services are good. Well done!
Sentiment  : negative
Confidence : 69.99%
```

Figure 4.1.3.3: Raw sentiment prediction using pre-trained XLNet model

Figure 4.1.3.3 above shows the pre-trained model predicting a positive sentiment. The model predicts it as negative. This shows an instance of an inaccurate prediction by the model.

	precision	recall	f1-score	support
negative	0.69	0.89	0.78	14948
positive	0.84	0.61	0.71	15052
accuracy			0.75	30000
macro avg	0.77	0.75	0.74	30000
weighted avg	0.77	0.75	0.74	30000

Figure 4.1.3.4: Classification report for fine-tuned XLNet model

The classification report shown in Figure 4.1.3.4 above demonstrates a notable performance improvement after fine-tuning the XLNet model. For the negative class, the model achieved a precision of 0.69, a recall of 0.89, and an f1 score of 0.78. The high recall indicates the model is quite effective at identifying negative instances and correctly capturing most negative cases. In the positive class, the model achieved a precision of 0.84, a recall of 0.61, and an f1 score of 0.71. The high precision indicates that when the model predicts a positive instance, it is correct 84% of the time, showing its strength in minimizing false positives. However, the recall of 0.61 reveals that the model misses some actual positive instances. The overall accuracy of 0.75 signifies a significant improvement over the pre-trained model, reflecting a much more reliable performance distinguishing between negative and positive instances.

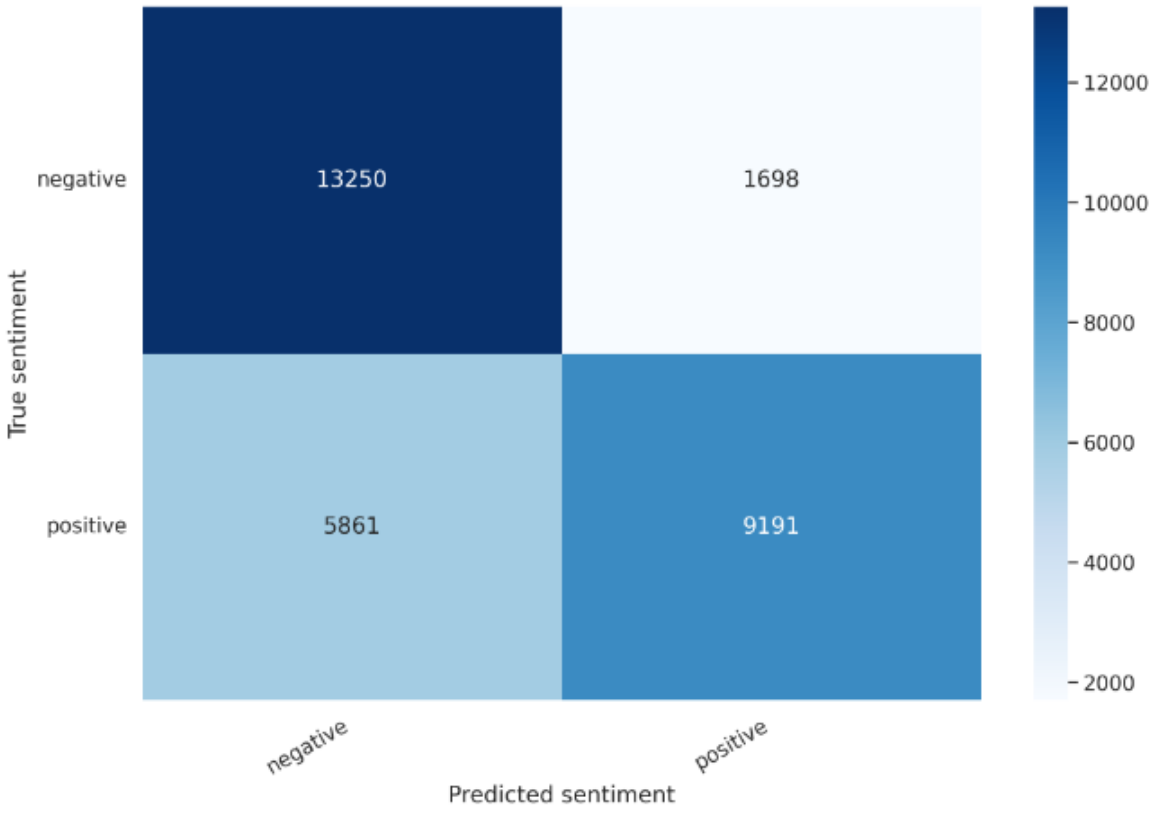

Figure 4.1.3.5: Confusion matrix for fine-tuned XLNet model

The confusion matrix for the fine-tuned XLNet model shown in Figure 4.1.3.5 reveals significant improvements in correctly classifying negative and positive instances. For the negative class, the model correctly predicted 13,250 out of 14,948 instances as negative. This high number of true negatives indicates a strong ability to identify negative instances, corresponding with the high recall of 0.89 reported in the classification metrics. The relatively low number of false positives suggests the model can accurately differentiate negative instances, reflecting a precision of 0.69.

In the positive class, the model correctly identified 9,191 out of 15,052 instances as positive. This shows a notable improvement compared to the pre-trained model, highlighting its ability to recognize positive cases with a precision of 0.84. However, the model still misclassifies some positive instances as negative, evident in the recall of 0.61.

```
Review text: I got my fries very late from your shop. I am very sad. You should improve your services.
Sentiment  : positive
Confidence : 68.40%
```

Figure 4.1.3.6: Raw sentiment prediction using fine-tuned XLNet model

Figure 4.1.3.6 above shows the prediction of negative sentiment by the fine-tuned XLNet model. The model predicted it as positive. This shows an instance where the model incorrectly predicts the polarity of a sentiment.

4.1.4 ELECTRA Results

negative	0.44	0.07	0.12	14948
positive	0.50	0.91	0.64	15052
accuracy			0.49	30000
macro avg	0.47	0.49	0.38	30000
weighted avg	0.47	0.49	0.38	30000

Figure 4.1.4.1: Classification report for pre-trained ELECTRA model

The evaluation of the pre-trained ELECTRA model reveals significant discrepancies in its performance across the negative and positive classes, resulting in an overall accuracy of 0.49, as shown in Figure 4.1.4.1 above. For the negative class, the model achieved a precision of 0.44, a recall of 0.07, and an f1 score of 0.12. These metrics indicate a severe challenge in correctly identifying negative instances. The very low recall suggests that the model is missing most actual negative cases, leading to many false positives. The f1 score of 0.12 highlights the imbalance between precision and recall, with the model struggling to classify negative instances effectively.

In contrast, the positive class shows a much stronger performance, with a precision of 0.50, a recall of 0.91, and an f1 score of 0.64. The high recall indicates that the model is highly effective at identifying positive instances and capturing the most true positive cases.

However, the precision of 0.50 suggests many false positives, affecting the model's ability to distinguish positive instances confidently.

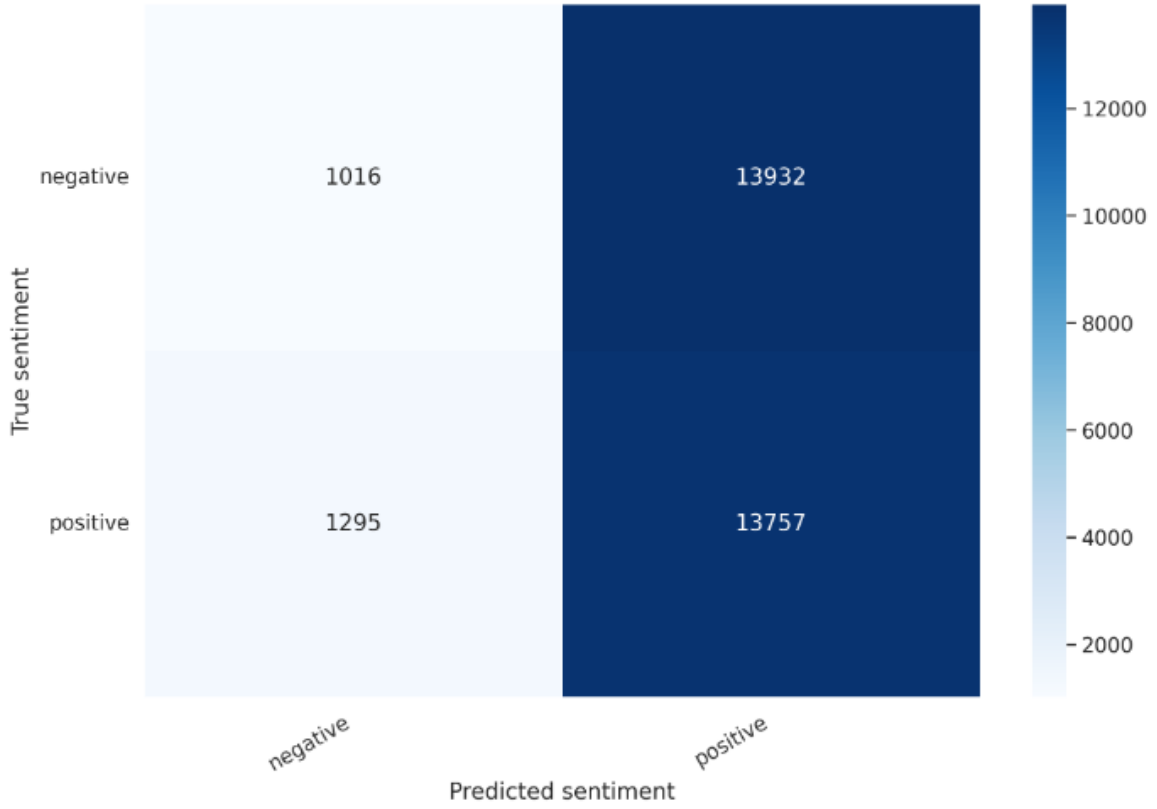

Figure 4.1.4.2: Confusion matrix for pre-trained ELECTRA

The confusion matrix for the pre-trained EECTRA model, shown in Figure 4.1.4.2 above, indicates a significant disparity in accurately classifying negative and positive instances. For the negative class, the model correctly predicted only 1,016 out of 14,948 instances as negative. This results in many false positives and demonstrates the model's difficulty in identifying negative cases, aligning with the low recall of 0.07 reported in the classification metrics. The model shows strong performance for the positive class, correctly predicting 13,757 out of 15,052 instances as positive. This reflects the model's high recall of 0.91 for the positive class, indicating its effectiveness in identifying positive cases. The model's performance is biased towards predicting positive instances, as evidenced by the overall accuracy of 0.49, which is close to random guessing.

```
Review text: Your services are good. Well done!
Sentiment  : positive
Confidence : 59.04%
```

Figure 4.1.4.3: Raw sentiment prediction using pre-trained ELECTRA model

Figure 4.1.4.3 shows the pre-trained ELECTRA predicting a positive sentiment. The model correctly predicts it.

	precision	recall	f1-score	support
negative	0.77	0.59	0.67	14948
positive	0.67	0.83	0.74	15052
accuracy			0.71	30000
macro avg	0.72	0.71	0.70	30000
weighted avg	0.72	0.71	0.70	30000

Figure 4.1.4.4: Classification report for fine-tuned ELECTRA model

After fine-tuning the ELECTRA model, it exhibits significant improvements in classification performance for both the negative and positive classes, achieving an overall accuracy of 0.71. For the negative class, the model attained a precision of 0.77, a recall of 0.59, and an f1 score of 0.67. This represents a substantial improvement over the pre-trained model, indicating that the fine-tuned model is now more capable of correctly identifying negative instances. The increase in recall shows that the model is better at capturing true negative cases. For the positive class, the model achieved a precision of 0.67, a recall of 0.83, and an f1 score of 0.74. The high recall demonstrates the model's strength in accurately identifying positive cases, and improved precision suggests a decrease in false positives compared to the pre-trained model.

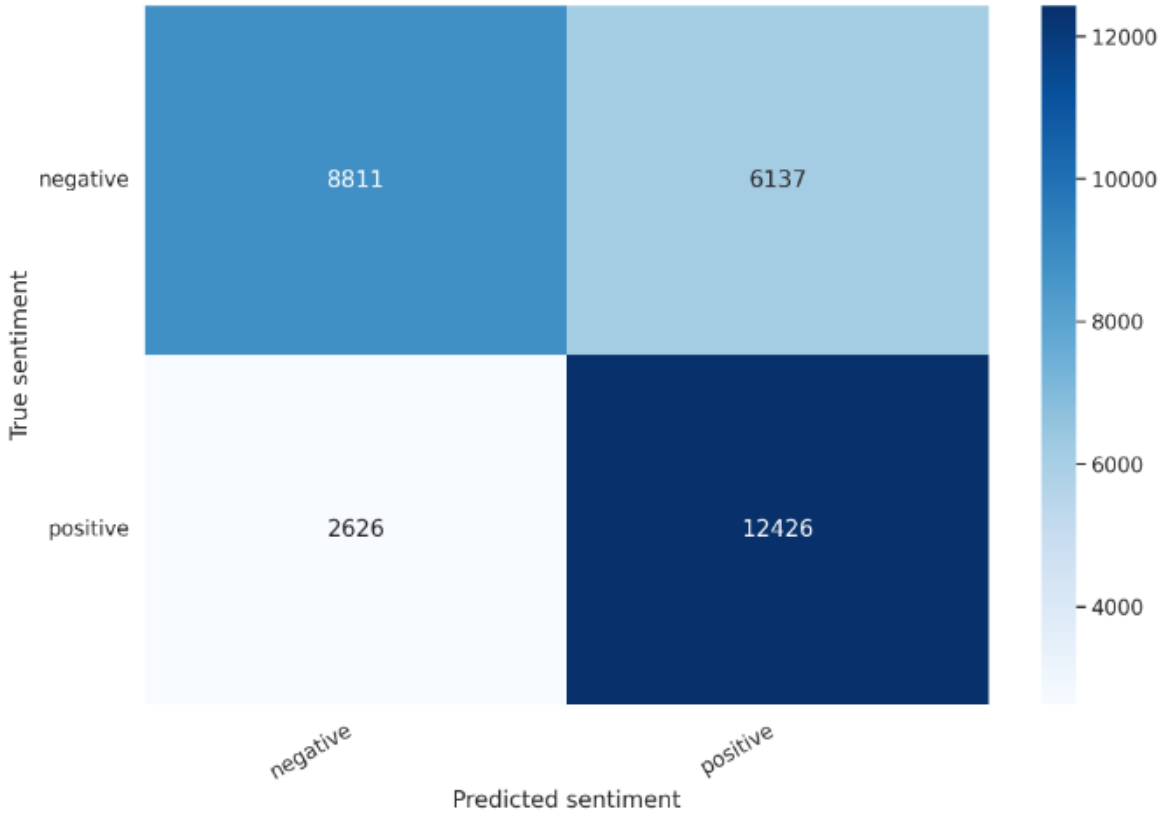

Figure 4.1.4.5: Confusion matrix for fine-tuned ELECTRA model

The confusion matrix for the fine-tuned ELECTRA model demonstrates marked improvement in its ability to accurately classify negative and positive instances, as shown in Figure 4.1.4.5. For the negative class, the model correctly predicted 8,811 out of 14,948 instances as negative. This indicates a significant improvement in identifying negative cases compared to the pre-trained model. For the positive class, the model correctly identified 12,426 out of 15,052 instances as positive. This high number of true positives emphasizes the model's strong performance in recognizing positive cases, consistent with the high recall reported for the positive class. Overall, the fine-tuned ELECTRA model achieves an accuracy of 0.71.

```
Review text: I got my fries very late from your shop. I am very sad. You should improve your services.
Sentiment  : negative
Confidence : 65.10%
```

Figure 4.1.4.6: Raw sentiment prediction using fine-tuned ELECTRA model

Figure 4.1.4.6 shows the prediction of negative sentiment by the fine-tuned ELECTRA model. The model correctly predicts the sentiment as negative with 65.10% confidence.

4.2 Comparative Analysis of Sentiment Analysis Models

4.2.1 Pre-trained models

The performance of the pre-trained models on the sentiment analysis task is shown in the table below in Figure 4.2.1.1:

Model	Precision	Recall	F1 score	Accuracy
Distilbert	0.59	0.50	0.34	0.50
GPT2	0.25	0.50	0.33	0.50
Xlnet	0.50	0.50	0.47	0.50
Electra	0.47	0.49	0.38	0.49

Fig 4.2.1.1: Table of the results of sentiment analysis using the pre-trained models

The results show that XLNet achieved the highest F1 Score (0.47) among the pre-trained models and Accuracy (0.50). DistilBERT also performed reasonably well, with a Precision of 0.59 and an Accuracy of 0.50. GPT-2 had the lowest Precision (0.25) and F1 Score (0.33), indicating that it struggled with the sentiment classification task without fine-tuning.

4.2.2 Fine-tuned Models

After fine-tuning the models, significant improvements in their performance were observed, as shown in the table below:

Model	Precision	Recall	F1 score	Accuracy
Distilbert	0.85	0.85	0.85	0.85
GPT2	0.83	0.83	0.83	0.83
XLnet	0.77	0.75	0.74	0.75
Electra	0.72	0.71	0.70	0.71

Fig 4.2.2.1: Table of the results of sentiment analysis using the fine-tuned models

The fine-tuned models exhibited marked improvements across all metrics. DistilBERT achieved the highest performance with an F1 Score and Accuracy of 0.85, indicating that it was highly effective after fine-tuning. GPT-2 also showed substantial improvement, achieving an F1 Score and Accuracy of 0.83. XLNet and ELECTRA improved significantly, with XLNet achieving an F1 Score of 0.74 and Accuracy of 0.75, while ELECTRA achieved an F1 Score of 0.70 and Accuracy of 0.71.

4.2.3 Summary of Comparative Analysis

All models showed considerable improvement in their evaluation metrics after fine-tuning. This highlights the importance of fine-tuning pre-trained models on the specific dataset to enhance performance. DistilBERT emerged as the best-performing model post-finetuning, with the highest scores across all metrics. This indicates its robustness and effectiveness in the sentiment analysis task. GPT-2, which initially performed poorly as a pre-trained model, showed significant gains after fine-tuning, suggesting that it requires more task-specific training to perform well. XLNet and ELECTRA also benefited from fine-tuning, although their gains were less pronounced than those of DistilBERT and GPT-2.

4.3 Comparative Analysis of Prompt Engineering Models

4.3.1 Positive Review Analysis

LlaMA (Meta-Llama-3-8B-Instruct):

A randomly selected positive sentiment was tested using LlaMA (Meta-Llama-3-8B-Instruct) and Mistral (mistralai/Mistral-7B-Instruct-v0.1). The sentiment was:

- Huppert does a wonderful portrayal of an inarticulate beauty who falls in love with a law student. Inevitably their affair ends unhappily. Good supporting performances by her mother and best friend. I like very few romantic movies, but I saw this one 4 times and still remember it after 20 years.

It was tested using the below prompts:

- List the two main topics discussed in this positive review.
- Explain briefly in one sentence why this review is positive.

The results shown in the table below were obtained.

Model	Themes
Meta-Llama-3-8B-Instruct	1. The complexity of human emotions and relationships. 2. Superb cinematography, music, and acting
mistralai/Mistral-7B-Instruct-v0.1	1. Huppert's portrayal and supporting performances. 2. Reviewer's appreciation despite not liking romantic movies.

Model	Drivers
Meta-Llama-3-8B-Instruct	It's a beautiful, poignant, and memorable film. Highly recommended, especially for French cinema and Isabelle Huppert fans.
mistralai/Mistral-7B-Instruct-v0.1	No response

Fig 4.3.1.1: Table showing prompt results

Themes:

- Complexity of human emotions and fragility of relationships: The model identifies the nuanced portrayal of emotions and relationships as a central theme.
- Stunning cinematography, beautiful music, and superb acting contribute significantly to the positive sentiment.

- Appreciation for French Cinema and Storytelling: The review admires French cinema's style and narrative techniques.

Sentiment Drivers:

- Poignant and memorable film: The film's emotional impact and lasting impression are key drivers of the positive sentiment.
- High recommendation for fans of French cinema and Isabelle Huppert: The reviewer's endorsement is particularly strong for specific audiences, adding to the positive sentiment.

Mistral (mistral/Mistral-7B-Instruct-v0.1):

The results for Mistral were as follows.

Themes:

- Portrayal of the character by Huppert: The performance of the lead actress is identified as a significant theme.
- Supporting performances by her mother and best friend: These performances are also noted as contributing to the review.
- Reviewer's enjoyment despite usually not liking romantic movies: The model captures the reviewer's unexpected enjoyment as a theme.

Sentiment Drivers:

- Not explicitly provided: The model does not delve into the specific aspects that drive the positive sentiment, limiting the depth of analysis.

4.3.2 Negative Sentiment Analysis

A negative sentiment was also tested:

- For sure, this tool did not work for me! Each time I have tried to use it, I ended up soaked to the skin. Did it move leaves down the gutter? No! Perhaps, someone will show me how to operate it before I pitch it into the trash.

The prompts used were similar. The table below shows the results obtained.

Model	Themes
Meta-Llama-3-8B-Instruct	1. The reviewer's personal frustration and disappointment. 2. Criticism of the product's design and value
mistralai/Mistral-7B-Instruct-v0.1	No response

Model	Drivers
Meta-Llama-3-8B-Instruct	Extremely disappointed, would not recommend, struggling to use, and highly frustrated.
mistralai/Mistral-7B-Instruct-v0.1	No response

Fig 4.3.2.1: Table showing the prompt results.

LlaMA (Meta-Llama-3-8B-Instruct):

Themes:

- Personal experience with the product leading to frustration and disappointment: The model identifies the reviewer's negative personal experience as a central theme.
- Criticism of the product's design and value: Specific issues with the product's design and perceived value are highlighted.

Sentiment Drivers:

- Extreme disappointment and frustration with the product: The reviewer's strong negative feelings are identified as primary drivers.
- Not recommending the product to others: The recommendation against the product contributes to the overall negative sentiment.

Mistral (mistralai/Mistral-7B-Instruct-v0.1):

Themes:

- The model does not provide themes for the negative review, resulting in a lack of detailed insights.

Sentiment Drivers:

- Not explicitly provided: Similar to the themes, the model does not extract specific sentiment drivers, limiting the understanding of the negative sentiment.

The comparative analysis reveals that LlaMA provides a more comprehensive and detailed analysis of positive and negative reviews. It successfully extracts themes and sentiment drivers, offering deep insights into the aspects contributing to the sentiments expressed in the reviews. In contrast, while identifying the main topics in positive reviews, Mistral lacks explicit sentiment drivers and fails to provide themes or drivers for negative reviews. This limitation results in a less detailed analysis and restricts the depth of insights drawn from the reviews.

Chapter 5: Conclusion and Recommendation

5.1 Summary

In this project, I conducted a comparative analysis of open-source large language models (LLMs) for sentiment analysis and prompt engineering. I utilized models such as DistilBERT, GPT-2, XLNet, and ELECTRA for sentiment analysis and LLaMA and Mistral for prompt engineering. The methodology included data preprocessing steps like checking for null values, adjusting sentiment values, tokenization, encoding, and padding and truncation. For sentiment analysis, I followed an approach where I froze the layers of the LLMs, introduced a classification layer with an output size of 2 (representing negative and positive classes), and used the softmax function to convert logits to probabilities. I tested the models before and after fine-tuning, with the fine-tuning process involving freezing other layers and only training the classification layer. For prompt engineering, I focused on understanding the elements contributing to negative sentiment in the text, employing LLaMA and Mistral to extract insights from the predicted reviews.

5.2 Limitations

The project faced several limitations that impacted the overall findings and conclusions. One major constraint was the inherent architecture and scope of the models used. For instance, DistilBERT, while being computationally efficient, is smaller than BERT. This architectural limitation may have affected its performance in sentiment analysis tasks. Additionally, the quality and diversity of the dataset used for training and evaluation posed significant challenges. The dataset might not fully represent the diverse language used in various real-world scenarios, which could lead to biased or less accurate predictions.

Another notable limitation was the resource intensity required for fine-tuning large language models. The process is computationally expensive and time-consuming, making

it less feasible for all research settings or practical applications. This high resource demand also restricts the ability to experiment with a broader range of models and configurations. Furthermore, the generalizability of the findings is limited by the specific models and datasets used in this study. Different models or datasets might yield varying results, thereby constraining the applicability of the conclusions drawn from this analysis to other contexts.

5.3 Future Work

Looking ahead, several avenues for future research could build on and enhance the findings of this project. One key area for future work is the expansion of model selection. Including a broader range of models, particularly newer and more advanced ones, could provide a more comprehensive comparison and deeper insights into the capabilities of different large language models for sentiment analysis and prompt engineering.

Enhancing the diversity and extent of datasets used for training and evaluation is another critical area. By incorporating more diverse datasets, future research could improve the robustness and generalizability of the findings, ensuring that the models perform well across a broader range of real-world scenarios. Efficiency improvements in fine-tuning also present a significant opportunity for future research. Investigating methods to reduce the computational cost and time required for fine-tuning, such as utilizing more efficient training algorithms or advanced hardware, could make these techniques more accessible to a broader audience. Additionally, applying the models and methodologies to real-world applications and evaluating their performance in practical scenarios could provide valuable insights and help fine-tune the approaches for specific use cases.

References

[1] Amazon reviews, "Kaggle," May 15, 2021. [Online]. Available: https://www.kaggle.com/datasets/kritanjalijain/amazon-reviews.

[2] M. E. Basiri and A. Kabiri, "Sentence-level sentiment analysis in Persian," *2017 3rd International Conference on Pattern Recognition and Image Analysis (IPRIA)*, Shahrekord, Iran, 2017, pp. 84-89, doi: 10.1109/PRIA.2017.7983023.

[3] R. Clarisó and J. Cabot, "Model-Driven Prompt Engineering," *2023 ACM/IEEE 26th International Conference on Model Driven Engineering Languages and Systems (MODELS)*, Västerås, Sweden, 2023, pp. 47-54, doi: 10.1109/MODELS58315.2023.00020

[4] S. Daulatkar and A. Deore, "Post Covid-19 Sentiment Analysis of Success of Online Learning: A Case Study of India," *2022 9th International Conference on Computing for Sustainable Global Development (INDIACom)*, New Delhi, India, 2022, pp. 460-465, doi: 10.23919/INDIACom54597.2022.9763272.

[5] Z. Fachrina and D. H. Widyantoro, "Aspect-sentiment classification in opinion mining using the combination of rule-based and machine learning," *2017 International Conference on Data and Software Engineering (ICoDSE)*, Palembang, Indonesia, 2017, pp. 1-6, doi: 10.1109/ICODSE.2017.8285850

[6] A. Fuggetta, "Open source software—an evaluation," *Journal of Systems and Software*, vol. 66, no. 1, pp. 77-90, 2003

[7] A. Ilmania, Abdurrahman, S. Cahyawijaya, and A. Purwarianti, 'Aspect Detection and Sentiment Classification Using Deep Neural Network for Indonesian Aspect-Based Sentiment Analysis,' in *2018 International Conference on Asian Language Processing*

(IALP), Bandung, Indonesia: IEEE, Nov. 2018, pp. 62–67. doi: 10.1109/IALP.2018.8629181.

[8] S. Jha, S. K. Jha, P. Lincoln, N. D. Bastian, A. Velasquez, and S. Neema, 'Dehallucinating Large Language Models Using Formal Methods Guided Iterative Prompting,' in *2023 IEEE International Conference on Assured Autonomy (ICAA)*, Laurel, MD, USA: IEEE, Jun. 2023, pp. 149–152. doi: 10.1109/ICAA58325.2023.00029.

[9] A. Joshy and S. Sundar, 'Analyzing the Performance of Sentiment Analysis using BERT, DistilBERT, and RoBERTa,' in *2022 IEEE International Power and Renewable Energy Conference (IPRECON)*, Kollam, India: IEEE, Dec. 2022, pp. 1–6. doi: 10.1109/IPRECON55716.2022.10059542.

[10] G. Li, Q. Zheng, L. Zhang, S. Guo, and L. Niu, 'Sentiment Infomation based Model For Chinese text Sentiment Analysis,' in *2020 IEEE 3rd International Conference on Automation, Electronics and Electrical Engineering (AUTEEE)*, Shenyang, China: IEEE, Nov. 2020, pp. 366–371. doi: 10.1109/AUTEEE50969.2020.9315668.

[11] L. Mathew and V. R. Bindu, 'A Review of Natural Language Processing Techniques for Sentiment Analysis using Pre-trained Models,' in *2020 Fourth International Conference on Computing Methodologies and Communication (ICCMC)*, Erode, India: IEEE, Mar. 2020, pp. 340–345. doi: 10.1109/ICCMC48092.2020.ICCMC-00064.

[12] W. Medhat, A. Hassan, and H. Korashy, "Sentiment analysis algorithms and applications: A survey," *Ain Shams Engineering Journal*, vol. 5, no. 4, pp. 1093-1113, 2014

[13] B. Meskó, "Prompt engineering as an important emerging skill for medical professionals: tutorial," *Journal of Medical Internet Research*, vol. 25, p. e50638, 2023

[14] R. Rahmania, H. I. Pohan, A. I. Arrahmah, and S. A. Wibowo, 'Performance Analysis of VADER and RoBERTa Methods for Smart Retail Customer Sentiment on Amazon Go Store,' in *2023 International Conference on Modeling & E-Information Research, Artificial Learning and Digital Applications (ICMERALDA)*, Karawang, Indonesia: IEEE, Nov. 2023, pp. 277–282. doi: 10.1109/ICMERALDA60125.2023.10458153.

[15] V. Ramanathan and T. Meyyappan, 'Twitter Text Mining for Sentiment Analysis on People's Feedback about Oman Tourism,' in *2019 4th MEC International Conference on Big Data and Smart City (ICBDSC)*, Muscat, Oman: IEEE, Jan. 2019, pp. 1–5. doi: 10.1109/ICBDSC.2019.8645596.

[16] V. A. Rohani and S. Shayaa, 'Utilizing machine learning in Sentiment Analysis: SentiRobo approach,' in *2015 International Symposium on Technology Management and Emerging Technologies (ISTMET)*, Langkawai Island, Kedah, Malaysia: IEEE, Aug. 2015, pp. 263–267. doi: 10.1109/ISTMET.2015.7359041.

[17] V. Sanh, L. Debut, J. Chaumond, and T. Wolf, "DistilBERT, a distilled version of BERT: smaller, faster, cheaper and lighter," *arXiv preprint arXiv:1910.01108*, 2019.

[18] S. Shaikh, S. M. Daudpota, S. Y. Yayilgan, and S. Sindhu, 'Exploring the potential of large-language models (LLMs) for student feedback sentiment analysis,' in *2023 International Conference on Frontiers of Information Technology (FIT)*, Islamabad, Pakistan: IEEE, Dec. 2023, pp. 214–219. doi: 10.1109/FIT60620.2023.00047.

[19] Y. Shen, L. Heacock, J. Elias, K. D. Hentel, B. Reig, G. Shih, and L. Moy, "ChatGPT and other large language models are double-edged swords," *Radiology*, vol. 307, no. 2, p. e230163, 2023.

[20] V. Tran and T. Matsui, 'Public Opinion Mining Using Large Language Models on COVID-19 Related Tweets', in *2023 15th International Conference on Knowledge and*

Systems Engineering (KSE), Hanoi, Vietnam: IEEE, Oct. 2023, pp. 1–6. doi: 10.1109/KSE59128.2023.10299499.

[21] S. R. Varghese, S. Juliet and A. N. S, "Social Media Text Analysis for Disaster Management Using DistilBERT Model," *2024 International Conference on Science Technology Engineering and Management (ICSTEM)*, Coimbatore, India, 2024, pp. 1-7, doi: 10.1109/ICSTEM61137.2024.10560620.

[22] X. Zhang, 'The Application of Natural Language Processing Technology Based on Deep Learning in Japanese Sentiment Analysis,' in *2023 International Conference on Ambient Intelligence, Knowledge Informatics and Industrial Electronics (AIKIIE)*, Ballari, India: IEEE, Nov. 2023, pp. 1–5. doi: 10.1109/AIKIIE60097.2023.10390437.

[23] P. Singh, B. Jain and K. Sinha, "Evaluating Bert and GPT-2 Models for Personalised Linkedin Post Recommendation," *2023 14th International Conference on Computing Communication and Networking Technologies (ICCCNT)*, Delhi, India, 2023, pp. 1-7, doi: 10.1109/ICCCNT56998.2023.10307957.

[24] S. Athithan, S. Sachi, A. K. Singh, A. Jain, Divya and Y. K. Sharma, "Twitter Fake News Detection by Using Xlnet Model," *2023 3rd International Conference on Technological Advancements in Computational Sciences (ICTACS)*, Tashkent, Uzbekistan, 2023, pp. 868-872, doi: 10.1109/ICTACS59847.2023.10389975.

[25] M. Taeb, Y. Torres, H. Chi, and S. Bernadin, "Investigating Gender and Racial Bias in ELECTRA," *2022 International Conference on Computational Science and Computational Intelligence (CSCI)*, Las Vegas, NV, USA, 2022, pp. 127-133, doi: 10.1109/CSCI58124.2022.00027.

[26] Meta-Llama, "Meta-Llama-3-8B-Instruct," Hugging Face, 2024. [Online]. Available: https://huggingface.co/meta-llama/Meta-Llama-3-8B-Instruct. [Accessed: 31-Jul-2024].

[27] MistralAI, "Mistral-7B-Instruct-v0.1," Hugging Face, 2024. [Online]. Available: https://huggingface.co/mistralai/Mistral-7B-Instruct-v0.1. [Accessed: 31-Jul-2024].