Sentiment Analysis with LLMs for Predicting Trends in Bitcoin

Trinity College Dublin
Coláiste na Tríonóide, Baile Átha Cliath
The University of Dublin

Ziang Liu

Supervisor: Prof. Anil Kokaram
Dr. Claire Gallagher

Department of Electronic and Electrical Engineering
Trinity College Dublin

This dissertaton is submitted in partial fulfillment for the degree of
MSc in Electronic Information Engineering

Trinity College Dublin July 2025

To my wife, for your love, patience, and belief in me, even in the most challenging times.
To my parents, for your support and encouragement throughout my journey.
To my teacher, for your guidance, inspiration, and the knowledge you shared.
To my friends, thank you for cheering me up and reminding me to smile .

Declaration

I declare that this thesis has not been submitted as an exercise for a degree at this or any other university and it is entirely my own work. It contains nothing which is the outcome of work done in collaboration with others, except as specified in the text and Acknowledgements.

I have read and I understand the plagiarism provisions in the General Regulations of the University Calendar for the current year. I have also completed the Online Tutorial provided by the Library, Trinity College Dublin, on avoiding plagiarism 'Ready, Steady, Write', located at `http://tcd-ie.libguides.com/plagiarism/ready-steadywrite`

I agree to deposit this thesis in the University's open access institutional repository or allow the library to do so on my behalf, subject to Irish Copyright Legislation and Trinity College Library conditions of use and acknowledgement.

Ziang Liu
July 2025

Acknowledgements

It was never an easy choice to come here to pursue a second master's degree. The challenges and uncertainties turned out to be far greater than I had imagined. But I'm grateful for everything we've been through, it has shaped us into better people and strengthened our beliefs more than ever. There's still a long way to go in life, and reaching this point has not been easy. But with perseverance, the goals ahead will surely be achieved.

Abstract

This project uses LLMs to perform sentiment analysis on financial news headlines to predict Bitcoin price trends. First we replicated FinBERT's performance and retrained it on GDELT subset to improve its accuracy from 64.8% to 73.8%. Next, three sentiment scores were extracted from GDELT news dataset using retranied FinBERT model and the results were aggregated to develop multiple sentiment signals. Then we calculate Bitcoin returns from Bitcoin price dataset and construct multiple return signals. By calculating Pearson correlation coefficient, we find that the continuous sum sigmoid sentiment signal demonstrates the strongest correlation with Bitcoin returns. Based on this finding, we develop several trading strategies. Quantitative analysis shows that the second sentiment based strategy has an average of 20 percentage points higher return than the buy-and-hold strategy for most of the time. Moreover, this strategy still generates positive returns given the overall downward trend and the high volatility of the Bitcoin price. This work contributes to both academic research and practical applications by demonstrating the effectiveness of Large Language Models in enhancing financial market analysis through sentiment based methods.

Table of contents

List of figures

List of tables

CHAPTER 1

Introduction and Aim

1.1 Background

SENTIMENT analysis [28] has emerged as a key method in financial market forecasting since the emotional sentiment conveyed in financial news and social media greatly affects investor sentiment and the market trends. Due to the complexity and volatility of financial markets, identifying market trends is crucial for investors but also highly challenging. With the immense and increasing amount of such textual data produced every day, it is no longer possible to extract insights manually. Thus, automated sentiment analysis based on Natural Language Processing (NLP) methods has acquired tremendous popularity [4] over the last decade.

Recent advances in NLP models such as FinBERT [4] and GPT-4 have opened up new opportunities for sentiment analysis of unstructured financial text. These are models that processed textual data to identify sentiments such as positive, negative, or neutral, and they could be used to assist in forecasting the impact of sentiment on stock prices. As accurate stock market predictions can result in improved decision-making, minimize risks, and yield higher returns, sentiment-based analysis [28] is now a high-priority research field for researchers and investors alike.

Cryptocurrencies [2] have been a highly valued field of research in financial science as an alternative investment platform alongside traditional financial markets. Growing unrest and unpredictability in the global markets have forced many investors to seek the potential of digital assets, particularly cryptocurrencies, as a means of diversifying their portfolios.

Current events highlight that while cryptocurrencies remain relatively new and not fully understood, they are likely to still have an important role [35] to play in the future of finance. It enables borderless real-time transactions at lower charges, providing users with unparalleled access to international payment systems. These features are attractive to users and businesses, even as increased regulatory activity threatens to define how these benefits come to fruition.

Among the myriad of cryptocurrencies, Bitcoin is the most widely used and first to have been proposed. Initially proposed by Nakamoto [22], it started a decentralized, open-source payment system which serves as the foundation of extensive multidisciplinary research. Bitcoin's open-source nature, growing user population, and influence on the broader cryptocurrency market [30] give it a perfect environment to explore the financial results and predictive modeling of digital assets.

Overall, sentiment has considerable influence on market trends and Bitcoin is playing an increasingly crucial role in modern financial systems. It is strongly motivated to explore how sentiment analysis can support cryptocurrency predictions. With the development of Natural Language Processing techniques and the improvement of financial models, it is now applicable to extract insights from large volumes of unstructured financial text. This project aims to use these techniques to investigate the relationship between sentiments in financial news and the returns of Bitcoin, and how Large Language Models like FinBERT can be leveraged to make market predictions.

1.2 Aim and objectives

Project Aim

The aim of this project is to extract sentiment from news headlines and subsequently analyze the extracted sentiment to predict trends in Bitcoin price movements. It also aims to build a trading signal to quantify the prediction results and compare with other strategies.

Project Objectives

1. To conduct a comprehensive review of existing literature on the use of Large Language Models in financial market prediction, focusing on sentiment analysis and cryptocurrency markets.

2. To replicate the performance of the FinBERT model on the GDELT news database for sentiment prediction.

3. To retrain and fine-tune the FinBERT model in order to enhance the accuracy of sentiment extraction from financial related news headlines.

4. To evaluate the performance of the retrained FinBERT model using statistical metrics.

5. To build the sentiment and return signals and find the strongest correlations between any of these signals.

6. To develop a sentiment-based trading signal to quantify the results and compare its performance against a standard buy-and-hold investment strategy for Bitcoin.

7. To compile the findings and insights into a final project report and deliver a formal presentation of the results.

1.3 Significance

This work explores the use of financial news headlines for predicting bitcoin price trends through automated sentiment analysis. Academically, it helps address the limitations of traditional financial models in processing unstructured textual data. It contributes to research on sentiment analysis by evaluating and enhancing FinBERT's performance in predicting Bitcoin price trends based on news headlines and also integrates machine learning theory with real-world business data.

By utilizing FinBERT to forecast sentiment from news headlines and analyze correlation with Bitcoin price movement, the project addresses a critical need for more accurate, data-driven approaches to cryptocurrency market predictions. In a very volatile market with limited conventional valuation models, sentiment based analysis offers an applicable option for better market predictions. It also supports real world business need, including improving better asset management and enabling effective algorithmic trading.

CHAPTER 2

Literature Review

This section presents a broad survey of major papers, approaches, and technologies relevant to sentiment analysis, with a particular focus on the FinBERT model and Bitcoin. By reviewing recent relevant studies, we aim to highlight the feasibility, methodologies, and limitations of using Large Language Models to predict market trends.

Several articles support the feasibility of predicting market trends by carrying out textual sentiment analysis of news headlines. According to Melvin and Yin [21], readers typically focus more on the headlines of financial news stories as they happen. So news headlines are shown to have substantial effect on the market returns. Wuthrich [37] created an online computational linguistics system for stock price prediction by analyzing news items from five well-known financial websites. Additionally, Samuel's study [8] provides evidence that sentiments conveyed through text streams may be useful for discovering patterns in a stock market index.

So the question becomes how to perform sentiment analysis. There are already many existing methods, techniques, and articles that use Large Language Models for sentiment analysis. Manoel [3] applied FinBERT to analyze news headlines in Global Data on Events, Location and Tone (GDELT) 2.0 event database, enriching the news dataset with supervised machine learning scores for Relevance, Sentiment, and Strength. His work concludes that GDELT 2.0 event database offers a more reliable dataset than news chosen from cryptocurrency-focused websites as it contains balanced number of good and bad news. Olamilekan [28] conducted a comparative study that examines the performance of several advanced AI models, including FinBERT, GPT-4, Logistic Regression on sentiment analysis. Logistic Regression achieved the highest accuracy (over 80%) while FinBERT, though more resource-intensive, offered deeper semantic insights. This study highlights the trade-offs between computational cost and analytical depth of practical applications of AI approaches in stock market prediction. Another research by Lee [18] suggests a way to enhance financial forecasting with Large Language Models. His findings demonstrate that LLMs perform better than conventional time-series models in market prediction tasks, although there are still challenges like reproducibility and explainability to be addressed.

In this paper, we focus mainly on FinBERT [4] and Bitcoin [22] in particular, because FinBERT is commonly used to extract sentiment from financial texts, and Bitcoin is a volatile asset often used to evaluate the impact of sentiment on market behavior. A more detailed discussion will be provided in the following sections of this chapter.

In summary, the rapid growth of financial texts has increased the need for accurate, efficient, and scalable analytical tools. Textual inputs pose greater challenges in processing and interpretation compared to numerical data. This necessitates the use of domain-specific large language models, trained specifically for financial language. Among these, FinBERT has emerged as one of the most effective models in the financial domain. The sentiment insights extracted from these models can significantly influence our understanding of market trends and investor behavior. However, despite the growing interest and advancements in sentiment-based forecasting, there remains a lack of robust quantitative frameworks that translate sentiment signals into actionable trading strategies. In other words, while models like FinBERT can effectively extract and classify sentiment, there is still a noticeable gap in linking these results directly to market decision-making processes, such as when to buy or sell assets.

2.1 Models, Data, Metrics

2.1.1 Large Language Model

Recently, Large Language Models (LLMs) [26] [23] have emerged as cutting-edge artificial intelligence systems that can process and generate text with coherent communication and generalize to multiple tasks.

Large Language Models (LLMs) are based on the Transformer architecture [32], which uses tokenization, positional encoding, and self-attention to process and understand text. They are first pretrained on large-scale text data to learn general language patterns, and then finetuned on specific tasks or domains. Models like FinBERT are adapted this way for financial sentiment analysis, making them effective in understanding and interpreting financial texts. Next, we focus on the four core components of the Large Language Models, providing a concise overview of their key mechanisms and design principles.

- **Tokenization**

  Tokenization is a crucial text preparation step to prepare for the input tokens for Large Language Models. WordPiece and BPE are two tokenization method commonly used by modern models, such as BERT based models and GPT based models. Language models [7] could not function on unprocessed textual data without tokenization .

  Through tokenization [29] [16], each token in the LLM pre-training and inference procedures is given a distinct number. The vector representation of each integer is represented by a matching row in a lookup table. The language model will then use this vector as the input for a specific token. For instance, if the input string is $S =$ ‘‘`The market is growing steadily`’’. By using a predefined tokenizer, the input will be tokenized into a sequence of tokens $\{t_1, t_2, \ldots, t_n\}$.

  $$S \rightarrow \{t_1, t_2, \ldots, t_n\} \tag{2.1}$$

  As mentioned above, tokenizers may vary depending on the model architecture. Commonly used strategies includes word-level, subword-level, or character-level tokenization. For example, the above sentence may be tokenized like this using a subword-level tokenizer [6]:

  $$\text{``The market is growing steadily"} \rightarrow \{\text{``The"}, \text{``market"}, \text{``is"}, \text{``grow"}, \text{``ing"}, \text{``steadily"}\}$$

  Each token t_i is then mapped to a corresponding integer index based on a fixed vocabulary $\mathcal{V}$:

  $$t_i \in \mathcal{V}, \quad \forall i \in \{1, 2, \ldots, n\} \tag{2.2}$$

  This indexed sequence serves as the input to the embedding layer, which subsequently maps each token index to a continuous vector in $\mathbb{R}^d$.

- **Transformer Architecture**

The Transformer model [36] follows a encoder-decoder structure. It is based on the multi-head attention mechanism, in which text is converted to numerical representations called tokens, and each token is converted into a vector via lookup from a word embedding table. Here shows the architecture of Transformer model in Figure 2.1.

The encoder is made up of encoding layers that process all of the input tokens together one layer after another. The decoder is made up of decoding layers that repeatedly process the encoder's output and the decoder's output tokens thus far.

Each encoder layer aims to provide contextualized representations of the tokens using a self-attention mechanism. By focusing on the encoder's output and the previously created tokens , each decoder layer creates the output sequence, enabling it to construct context-aware predictions one token at a time.

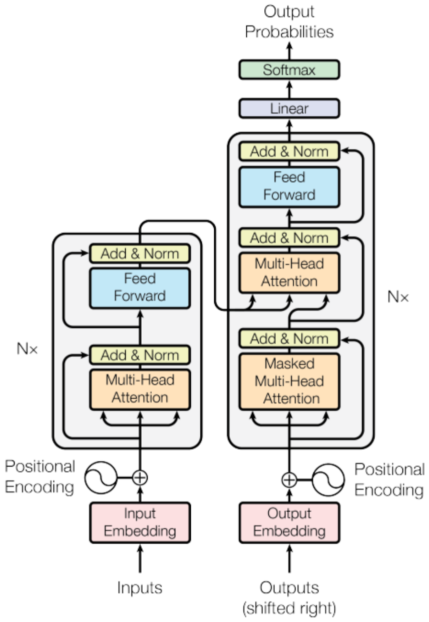

Fig. 2.1 The Transformer Architecture follows a encoder-decoder structure [32]

- **Positional Encoding**

  The meaning of words is greatly influenced by their sequence. The positions of the elements in a sequence are represented by positional vectors. Models that process sequences step-by-step can implicitly capture the order of the tokens. In contrast, the Transformer architecture processes all of the tokens in a sequence simultaneously, allowing it to obtain a complete picture of all the tokens. The self-attention mechanism allows for simultaneous processing, but it also implies that the model does not naturally understand where each token is in the sequence. Positional encodings [33] address this issue by giving each token a positional context. Transformer models utilize the position information by using a feature-level positional encoding. For example, convolutional seq2seq [15] proposed learnable position embeddings to represent the positions in a sequence.

Each token in the input sequence is initially mapped to a dense vector representation via an embedding layer [32]. Let the input sequence be denoted as $\{t_1, t_2, \ldots, t_n\}$, where each t_i is a token. The embedding layer will transform each token t_i into a vector $e_i \in \mathbb{R}^d$, where d is the dimension of the embedding space.

To integrate the positional information into the model, each dimension of the positional encoding corresponds to a sinusoid. The positional encoding $PE_{\text{pos},i}$ is defined for each position pos and each dimension index i:

$$PE_{\text{pos},2i} = \sin\left(\frac{\text{pos}}{10000^{\frac{2i}{d_{\text{model}}}}}\right) \tag{2.3}$$

$$PE_{\text{pos},2i+1} = \cos\left(\frac{\text{pos}}{10000^{\frac{2i}{d_{\text{model}}}}}\right) \tag{2.4}$$

Here, d_{model} denotes the dimension of the model embeddings. These functions ensure that each position in the sequence has a unique encoding, and their relative positions could be determined by the model as well. So the resulting positional encoding matrix has the same dimension d as the token embeddings, enabling direct element wise addition.

The input to the Transformer encoder is then calculated by adding the positional encodings to the token embeddings:

$$z_i = e_i + PE_i, \quad \text{for } i = 1, 2, \ldots, n \tag{2.5}$$

The combined representations $\{z_1, z_2, \ldots, z_n\}$ now contains information from both the semantic content of the tokens and their position within the sequence. Next, $\{z_1, z_2, \ldots, z_n\}$ are passed to the Transformer layers for further processing.

- **Multi-Head Attention**

A mechanism of attention [32] [24] takes in a query and an ordered list of key-value pairs, and produces an output vector. Everything in the query, keys, values, and the output are vectors. The query is weighted over the value vectors in the output, where the weights are calculated by a compatibility function deciding how well each key is represented by the query.

In practice, the queries, the values and the keys are integrated into matrix Q, K and V. The attention function is calculated as:

$$\text{Attention}(Q,K,V) = \text{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)V \tag{2.6}$$

where the dot products between the query and key vectors determine the attention weights, which are then applied to the value vectors to produce the final output.

In order to allow the model to attend to information from different representation subspaces, it's optimal to use multiple attention heads, each with its own set of learned linear projections. It enables the model to capture a richer set of relationships by focusing on different parts of the input in parallel.

$$\text{MultiHead}(Q,K,V) = \text{Concat}(\text{head}_1,\ldots,\text{head}_h)W^O \tag{2.7}$$

$$\text{where} \quad \text{head}_i = \text{Attention}(QW_i^Q, KW_i^K, VW_i^V) \tag{2.8}$$

W_i^Q, W_i^K, W_i^V are learnable projection matrices for the i-th attention head. W^O is the final output projection matrix applied after concatenating all heads.

2.1.2 FinBERT

FinBERT [4] is a language model designed especially to perform natural language processing (NLP) tasks in the finance industry. It could process textual data and generate scores that are related to the sentiment of that text. It has been demonstrated to perform better on financial sentiment analysis datasets than the most advanced machine learning techniques.

Based on BERT, FinBERT was initially trained on a financial corpus and refined for sentiment analysis in financial domain. The dataset includes TRC2-financial, Financial PhraseBank, FiQA Sentiment. FinBERT is one of the earliest BERT-based models specifically adapted for finance and among the few that incorporate additional pretraining on domain-specific corpora. According to the original FinBERT paper [4], FinBERT outperforms previous models by up to 15% in classification accuracy on standard financial sentiment datasets.

This strong performance makes FinBERT an ideal baseline for our project, which aims to perform sentiment analysis on financial news headlines. By leveraging FinBERT's ability to comprehend domain-specific language, we can better classify sentiment as positive, negative, or neutral—a highly important step in gauging market reaction and investor sentiment. Furthermore, pre-training on financial corpus ensures it learns subtle linguistic cues general-purpose models overlook. Thus, FinBERT not only enhances sentiment classification accuracy but also results in more knowledgeable downstream tasks such as risk analysis, trend forecasting, and strategic financial decision-making.

2.1.3 Bitcoin

The cryptocurrency market [31] has been a perfect test field for sentiment analysis due to its unique character. It is a decentralized and highly volatile financial system, and it offers both high-frequency trade opportunities and intense public debate, which are perfect for studying the dynamics of market sentiment and prices [12]. Compared to traditional financial markets, cryptocurrencies have lower barriers to entry and greater data availability so that they can provide abundant real-time data. These characteristics make cryptocurrencies an ideal subject for sentiment analysis and machine learning applications which aimed at understanding market behavior.

Most of the past studies about cryptocurrencies have been focusing on Bitcoin, and it has becoming the most popular an actively traded digital asset. The Bitcoin price data shown in Fig. 2.2 is obtained using the yfinance Python library, which retrieves historical financial data from Yahoo Finance [38]. The dataset reflects the daily closing prices of Bitcoin from June 1, 2015, to June 1, 2025. While Bitcoin has shown a significant upward trend overall since 2017, but its volatility remains high. Many people viewed it as the potential digital alternative to fiat currency, because of its security and decentralization. Over the past few decades, Bitcoin has consistently dominated the crypto market, making it a making it a logical focal point for research [17].

2.1.4 News Dataset

Many research and studies have been focusing on sentiment analysis based on news headlines. News based sentiment analysis could capture real world events and how do the public react to them. Therefore, they are crucial to provide essential context for financial market trend predictions. These news datasets typically contain metadata including timestamps, source of information, sentiment labels and entities mentioned. In this work,

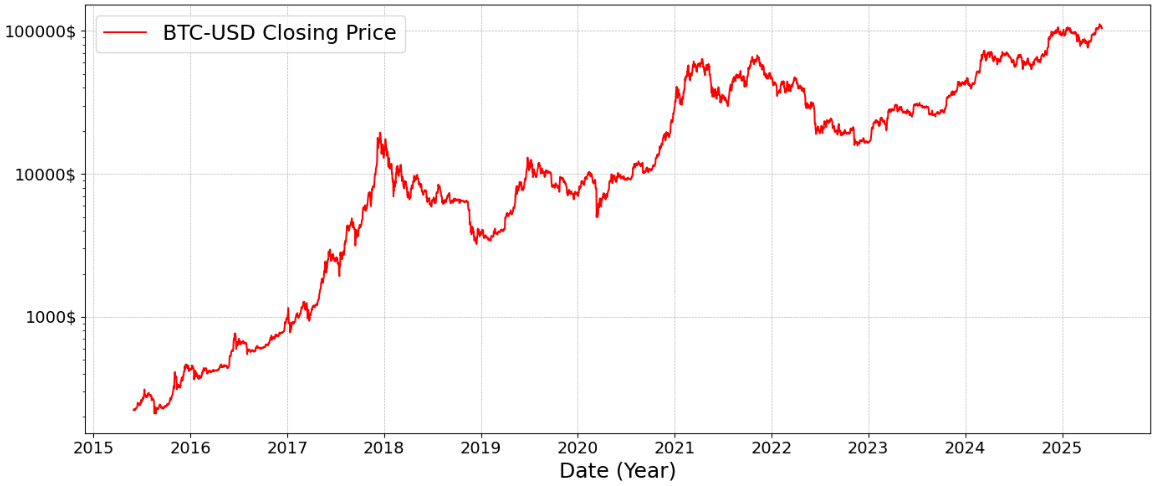

Fig. 2.2 Logarithm of Bitcoin price from 2015-06-01 to 2025-06-01. It shows a significant upward trend with high volatility throughout the period.

we mainly use GDELT 2.0 event database and CryptoLin as our news datasets. Here's a detailed overview of them.

- **GDELT 2.0 event database**

  In this project, the FinBERT model is retrained and evaluated using news headlines from Global Data on Events, Location and Tone database (GDELT) 2.0 event database [19] [34]. It contains more than 200 million geolocated events, covering over 300 categories of physical activities around the world.

  The database belongs to the GDELT project [14], which is one of the most ambitious platforms ever created for global news monitoring and real-time analysis of human society. It continuously monitors news media from over 100 countries in 65 languages and translates and processes them within 15 minutes of publication. GDELT 2.0 event database translates the massive volumes of global news into a structured, computable format. And it also extracts the events, entities, themes, multimedia content and emotions to create a rich and live metadata stream. This provides researchers with resources to perform large scale analysis of global events, sentiments and trends.

  The news data downloaded from GDELT 2.0 event database includes fields such as url, url mobile, title, seen date, social image, domain, language, and source country. However, it does not contain sentiment labels. Therefore, we apply external sentiment analysis tools to assess the emotional tone of each article title.

- **CryptoLin**

Using the cryptocurrency CoinMarketCal website [11], the CryptoLin data set [13] contained 2683 news pieces on all cryptocurrencies that were web crawled in English from many sources during a 42-month period (July 2018–January 2022).

Students from IE Business School manually annotated the CryptoLin dataset with discrete values that stood for negative, neutral, and positive news, respectively. They were all master's students at IE University with an average of eight years of work experience. Three randomly assigned annotators label each news article, and then simple voting was used to reach a consensus. If one of the annotators completely disagreed with the other two (1 negative vs. 2 positive or 1 positive vs. 2 negative), the label was set to neutral by default.

A text span is also included in the annotations, giving the three manual labels further context for the logic behind the choice. It also incorporates other data, such as the Fama French Three Factor Model and the results of applying many pre-trained Sentiment Analysis models, such as FinBERT, to further confirm the accuracy of the labeling and the use of CryptoLin.

2.1.5 Evaluation Metrics

- **Recall, Precision, Accuracy, F1 Score**

  In a balanced dataset, three commonly used evaluation metrics in this work are defined as:

$$\text{Recall} = \frac{TP}{TP + FN} \tag{2.9}$$

$$\text{Precision} = \frac{TP}{TP + FP} \tag{2.10}$$

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN} \tag{2.11}$$

$$\text{F1 Score} = \frac{2 \cdot TP}{2 \cdot TP + FP + FN} \tag{2.12}$$

  Where TP, TN, FP, FN represent true positives, true negatives, false positives, and false negatives, respectively.

- **Return on Investment**

Return on Investment (ROI) [1] is a measure of the gain or loss of an investment over a specific period, expressed as the gain or loss divided by initial value of investment . Here we define 1 week and 1 day return as follow, P_{start} is the price at the starting date, P_{week} is the price after 1 week from the starting date:

$$R_w = \left(\frac{P_w - P_0}{P_0}\right) \times 100\% \tag{2.13}$$

Similarly, the daily return is defined as:

$$R_d = \left(\frac{P_d - P_0}{P_0}\right) \times 100\% \tag{2.14}$$

where R_w and R_d denote the weekly and daily returns, respectively; P_0 is the initial price, P_d is the price after one day, and P_w is the price after one week.

- **Pearson Correlation**

  The Pearson correlation [5] coefficient is a measure of the linear correlation between two variables x and y. The correlation coefficient r is defined as:

$$r = \frac{\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^{n}(x_i - \bar{x})^2} \cdot \sqrt{\sum_{i=1}^{n}(y_i - \bar{y})^2}} \tag{2.15}$$

  where $x = (x_1, x_2, \ldots, x_n)$, $y = (y_1, y_2, \ldots, y_n)$, and $\bar{x}$ and $\bar{y}$ are the means of x and y, respectively.

- **Trading Signal**

  A trade signal [9] serves as a prompt to take action—either buying or selling a security or asset—based on analytical methods. Generally, trading signals are built on the technical indicators that help investors make trading strategies. These indicators are either buy, sell or hold, which are derived from historical stock data or other data sources. Its main goal is to identify optimal turning points in stock prices. However, predicting these signals is challenging due to the high dimensionality and non-stationary nature of price fluctuations. They are influenced by the economic environment and political events which keep complicating the predicting process.

  To address these challenges, researchers have proposed various methods for constructing trading signals. For example, Chen [10] integrate artificial neural network with an intelligent piecewise linear representation (PLR). First, turning points are chosen from the historical stock price database using piecewise linear representation. Next, the PLR result is converted into trading signals. Then, neural networks learned the relationship between technical indexes and trade signals. At last, a dynamic threshold technique is used to identify the buy-sell signals during the test time. Similarly, Luo [20] employed PLR in conjunction with a weighted support vector machine. This approach aims to avoid the disadvantages in over-fitting, underfitting and the difficulties in choosing the threshold of the trading signal so that it could improve the robustness of trading signal construction.

In more recent research, Saud and Shakya [27] used LSTM and GRU networks, which can handle long-term dependencies and preserve context. The stock trading signal prediction strategies are created using several technical indicators like Moving Average Convergence Divergence (MACD), Directional Movement Index (DMI), and Know Sure Thing Indicator (KST). It comes to the conclusion that the MACD-based strategy is the most secure and successful.

In summary, trading signal construction is a crucial step in algorithmic trading which requires methods of capturing the market trends and reducing the noise. Based on these research, our project aims to build the trading signal using indicators derived from sentiment analysis. After identifying the sentiment signal most closely related to returns, we apply a simple threshold-based strategy to set buy or sell indicators. This approach could effectively turn sentiment trends into actionable trading decisions.

2.2 Project Contribution and Positioning

In this project, we replicate the performance of the FinBERT model and then fine-tune it to improve its accuracy in the context of cryptocurrency related news.headline-focused sentiment labeling We apply the retrained model to the GDELT news dataset to generate sentiment labels for financial news headlines. The Pearson correlation between the extracted sentiment scores and Bitcoin returns signals is evaluated to identify the relationship between sentiment and market trends. Based on this, we develop different sentiment driven trading signals and compare them with a baseline buy and hold strategy. This approach can effectively determine the impact of news sentiment on Bitcoin price trends.

This work explores the use of financial news headlines for predicting bitcoin price trends through automated sentiment analysis. It offers a pipeline from news headline collection to sentiment labeling using FinBERT and then to mapping to Bitcoin price trends or trade signals, and eventually closing the loop between textual data and market behavior. Instead of focusing on stock markets like most prior work, this project mainly focuses on Bitcoin, which has distinct volatility, investor sentiment, high public and institutional interest. We also assess FinBERT's limitations on crypto news and discuss issues like retraining needs, bias, or domain mismatch. This offers insight into how well a finance-tuned LLM works outside its original context. Last but not least, many papers stop at sentiment classification, yet we aim to generate interpretable trading signals, addressing the missing link between sentiment and returns, and evaluate the results quantitatively.

2.3 Limitation

- This study explore the impact of news sentiment on price or return, using a one-year time frame for analysis. Given the market volatility and trading signal strategies, the resulting returns may be sensitive to the choice of the start and end dates. Using a longer time frame or averaging results across multiple distinct periods could improve its robustness and reliability.
- The collection and cleaning of news headlines are relatively complex and time-consuming, and the dataset may not be well-balanced. Therefore, there may be more suitable or cleaner datasets available.
- This study does not take into account intraday price fluctuations or transaction fees; only the daily closing price is used as the price signal.
- Due to factors such as the size of the training set and model parameter tuning, there is still room for improvement in the performance of FinBERT.
- In constructing the trading signal, future work could adopt methods from other studies that involve designing new deep learning architectures to identify more optimal trading strategies.

CHAPTER 3

Methodology

3.1 Methodology Overview

As mentioned above, the goal of this work is to fine-tune FinBERT and use it to extract sentiment from news headlines. Based on the sentiment analysis results, trading signals are designed to quantitatively evaluate the performance of the model. The process architecture is shown below in Figure 3.1:

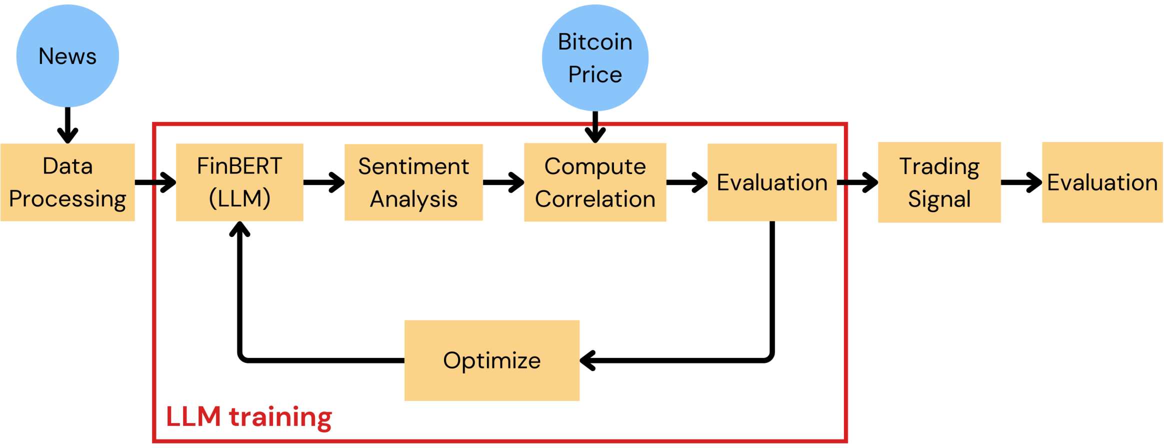

Fig. 3.1 Data Flow and Process Architecture

1. Before initiating our main methodology, we first replicated the performance of the original FinBERT model on CryptoLin dataset to verify its reported effectiveness.

2. Raw dataset of news headlines from GDELT were collected, cleaned and filtered.
3. The dataset was fed into the fine-tuned FinBERT model to conduct sentiment analysis, from which multiple sentiment signals were generated.
4. Bitcoin price data was downloaded and processed to prepare for different return signals.
5. The correlations between each of the sentiment signals and the return signals were calculated to identify the most strongly correlated pair.
6. After parameter tuning and performance evaluation, the LLM model has been trained to have the best performance.
7. Multiple trading signals were constructed based on the sentiment signals generated by the model.
8. The performance of trading signals were assessed quantitatively.
9. Finally, the trading strategies were optimized based on the analysis results to achieve higher returns.

3.2 Replicate FinBERT's performance

As mentioned by Manoel in his paper [3], FinBERT achieved only about 50% accuracy on the CryptoLin dataset. Therefore, he retrained and fine-tuned the FinBERT model using the same dataset. In order to evaluate the performance of FinBERT, we conducted a replication experiment using the following steps:

- **Data Preparation**

  First, we loaded the dataset `CryptoLin_IE_v2.csv` which contains 2683 manually labeled news in total. Next, We cleaned the dataset by dropping the unnecessary columns, and checked its label distribution:

- **Model Loading**

  We used the pre-trained original FinBERT model provided by ProsusAI [25] and its associated tokenizer from Hugging Face's `transformers` library.

Label	Sentiment	Count
1	Positive	1366
0	Neutral	921
-1	Negative	396
Total		2683

Table 3.1 Distribution of sentiment labels in the CryptoLin dataset.

- **Sentiment Prediction**

  Each news headline was tokenized and fed into FinBERT model. The model output went through a softmax function which will then generate the three sentiments possibilities s_{pos}, s_{neu}, and s_{neg}. These probabilities were added as new columns to the dataset.

- **Label Mapping**

  To assess the performance of FinBERT, the model's three sentiment scores must be mapped to match the manually labeled sentiment in the dataset. For each prediction, the sentiment class with the highest probability was selected and assigned to an integer value between -1, 0, 1. The sentiment score S is calculated using the following function:

$$\text{S} = f(s_{\text{pos}}, s_{\text{neu}}, s_{\text{neg}}) = \begin{cases} 1 & \text{, if } \max(s_{\text{pos}}, s_{\text{neu}}, s_{\text{neg}}) = s_{\text{pos}} \\ 0 & \text{, if } \max(s_{\text{pos}}, s_{\text{neu}}, s_{\text{neg}}) = s_{\text{neu}} \\ -1 & \text{, if } \max(s_{\text{pos}}, s_{\text{neu}}, s_{\text{neg}}) = s_{\text{neg}} \end{cases}$$

- **Evaluation**

  Eventually, the performance of the predicted sentiment labels were evaluated using accuracy, precision, recall, F1 score, and the confusion matrix. These metrics could provide a comprehensive evaluation of the model's performance across all sentiment categories.

3.3 Retraining FinBERT

To improve FinBERT's performance on financial news headlines from the GDELT dataset, we retrained the model using a manually labeled subset of GDELT data. The retraining process is shown below:

The performance of the pre-trained FinBERT model was evaluated on these manually labeled samples as baseline. Then we processed the news titles and fine tuned the model with various parameters. The loss and accuracy of training and testing dataset was monitored and plotted.

- **Data Preparation**

  Eventually, the FinBERT model was applied to news headlines from GDELT 2.0 event dataset, so the training data for fine tuning needs to be directly drawn from this source. However, the GDELT dataset does not contain sentiment annotations for headlines. We randomly selected 600 news headlines from GDELT and manually labeled them as ground truth.

Label	Sentiment	Count
1	Positive	291
0	Neutral	214
-1	Negative	95
Total		600

Table 3.2 Distribution of sentiment labels in the Manually labeled GDELT subset.

- **Data Preprocessing**

  We processed the news titles to reduce the input noise and enhance the model's performance. First, we managed to delete the stop words inside each news titles, which are meaningless common words like "and", "is", "the". They usually does not contain useful information for sentiment anaylsis. The list of English stopwords from the NLTK (Natural Language Toolkit) library were downloaded. The stop words were deleted and all character in the news titles were set to lowercase.

  Next, we removed the punctuation and special characters, leaving the news titles with only words and white spaces. At last, another natural language preprocessing step, lemmatization was implemented to further clean and normalize the news headlines. The titles were tokenized, and each word is reduced to its root form, then joined back to a single string.

- **Model Training and Evaluation**

  The dataset was split into 70% for training and 30% for testing. We initialized the pre-trained FinBERT model from Hugging Face. Training was conducted using the AdamW optimizer, with cross-entropy loss as the objective function. Batch size was set to 8, learning rate and weight decay was set to 5e-5 and 0.05 to prevent overfitting. The model was trained for 40 epochs while monitoring the loss and accuracy of training and validation set to ensure convergence and avoid overfitting. The final fine-tuned FinBERT model with the highest validation accuracy was saved for sentiment analysis.

3.4 Sentiment Analysis

Overview of the Bitcoin-related news dataset, Sentiment extraction using retrained FinBERT. How the sentiment signal and return signal are constructed. Correlation between sentiment and return.

- **Data Preparation**

  The cryptocurrency-related news dataset was downloaded from GDELT 2.0 event database [3]. Table 3.3 shows the first row of the dataset. Since the "url_mobile" column typically contains either invalid values or duplicates of the "url" column, it was excluded from the table. Additionally, columns such as "social image", "domain", "language" and "source country" were irrelevant from the sentiment analysis as well. So we only kept the "title" and "seen date" columns. The table was then reformatted and sorted chronologically by "seen date".

Key	**Value**
id	0
url	https://www.digitaljournal.com/pr/ longhash-ventures-and-terraform-labs-join-forces...
url_mobile	NaN
title	LongHash Ventures and Terraform Labs Join Forces to Advance Web3 Projects on the Terra Blockchain
seendate	20220406T163000Z
socialimage	NaN
domain	digitaljournal.com
language	English
sourcecountry	United States

Table 3.3 First row of GDELT news dataset

At last, we verified the integrity of the final dataset by checking the format and range of the dates, and also the number of news titles per day. Then the same preprocessing steps were applied to the news titles as mentioned above in the retraining section 3.3. We then obtained Bitcoin price data using the yfinance Python library [38].

- **Building Sentiment Signal**

  Next, we applied the retrained FinBERT model to the news titles to obtain sentiment scores. For each news headline, we obtained three sentiment scores: s_{pos}, s_{neu}, and s_{neg}. However, the Bitcoin price data was recorded on a daily basis, we needed to aggregate the sentiment scores of all news articles on the same day to calculate a daily sentiment score. Since we needed to identify the strongest correlation between any of the sentiment signals and return signals, it was necessary to experiment with multiple sentiment aggregation strategies. The aggregation methods are as follows:

$$s_d = \begin{cases} 1, & \text{if } \max(s_{\text{pos}}, s_{\text{neu}}, s_{\text{neg}}) = s_{\text{pos}} \\ 0, & \text{if } \max(s_{\text{pos}}, s_{\text{neu}}, s_{\text{neg}}) = s_{\text{neu}} \\ -1, & \text{if } \max(s_{\text{pos}}, s_{\text{neu}}, s_{\text{neg}}) = s_{\text{neg}} \end{cases} \tag{3.1}$$

$$s_c = -1 \cdot s_{\text{neg}} + 0 \cdot s_{\text{neu}} + 1 \cdot s_{\text{pos}} = s_{\text{pos}} - s_{\text{neg}} \tag{3.2}$$

  We use s_d and s_c to describe the discrete and continuous sentiment score of each news titles. And the series of sentiment scores are referred as $S_d = \{s_{d,1}, s_{d,2}, \dots, s_{d,N}\}$, $S_c = \{s_{c,1}, s_{c,2}, \dots, s_{c,N}\}$.

  But the number of news titles on each day varies significantly. According to the news dataset, there are over 1000 news on some day, but only 1 news on another day. Although the number of news articles can add noise to the daily sentiment signal, it also reflects the strength or intensity of the market sentiment. For example, if there are 1000 positive news articles on a given day, there must have been more impact on investors than 1 single positive news article. So we adopted the following approach to further refine the sentiment aggregation:

$$\bar{s}_d = \frac{1}{N}\sum_{i=1}^{N} s_{d,i} \qquad\qquad T_d = \sum_{i=1}^{N} s_{d,i} \tag{3.3}$$

$$\bar{s}_c = \frac{1}{N}\sum_{i=1}^{N} s_{c,i} \qquad\qquad T_c = \sum_{i=1}^{N} s_{c,i} \tag{3.4}$$

Where $\bar{s}_d$ and T_d is the mean and sum of all discrete sentiment scores, and $\bar{s}_c$ and T_c is the mean and sum of all continuous sentiment scores.

The return signals bounded within the range [-1, 1]. Since the value of T_d and T_c do not fall between -1 and 1, this may introduce more error when calculating correlation. Therefore, we applied 2 different functions to normalize them into the range of -1 to 1. The functions are defined as follow:

$$\mathrm{sgn}(T_d) = \begin{cases} 1 & \text{if } T_d > 0 \\ 0 & \text{if } T_d = 0 \\ -1 & \text{if } T_d < 0 \end{cases} \tag{3.5}$$

$$f(T_c) = 2\sigma(T_c) - 1 = \frac{2}{1 + e^{-T_c}} - 1 = \tanh\left(\frac{T_c}{2}\right) \tag{3.6}$$

In this step, we constructed a total of six sentiment signals, laying the groundwork for calculating correlation with the return signals in the next step.

- **Building Return Signal**

  Bitcoin price dataset contains daily market data including Date, closing price, highest and lowest prices, opening price and trading volume.

  Closing and opening price is the price at the start and end of the trading day. Trading volume is the total number of Bitcoin traded during the day.

  We selected the closing price as the indicator of Bitcoin's price because it reflects the final consensus value of the asset on that day.

  Since we want to identify the connection between sentiments and returns, we need to construct the return signals instead of directly using the price signals. The daily and weekly returns were calculated using Eq. 2.14 and Eq. 2.13, and were added to the price dataset.

  To maximize the correlation coefficient, we applied the same normalization function to both signals, ensuring they were within the same range as the sentiment signals:

$$\operatorname{sgn}(R_d) = \begin{cases} 1 & \text{if } R_d > 0 \\ 0 & \text{if } R_d = 0 \\ -1 & \text{if } R_d < 0 \end{cases} \tag{3.7}$$

$$\operatorname{sgn}(R_w) = \begin{cases} 1 & \text{if } R_w > 0 \\ 0 & \text{if } R_w = 0 \\ -1 & \text{if } R_w < 0 \end{cases} \tag{3.8}$$

$$f(R_d) = 2 * \sigma(R_d) - 1 = \tanh(\frac{R_d}{2}) \tag{3.9}$$

$$f(R_w) = 2 * \sigma(R_w) - 1 = \tanh(\frac{R_w}{2}) \tag{3.10}$$

At this stage, we have constructed six return signals, enabling us to compute the Pearson correlations (2.15) between these six return signals and the six sentiment signals above.

3.5 Trading Signal

In this section, two sentiment based trading strategies were tested and compared to a buy-and-hold baseline strategy. Let P_t denote the price of Bitcoin at time t, and B_t represent the amount of Bitcoin held at time t. Based on an initial capital of $C_0 = 1,000,000\$$, we aim to construct a trading signal over the course of one year that achieves the highest possible return R.

- **Buy-and-hold strategy**
  - As shown in Fig. 4.7, the buy-and-hold strategy invests all initial capital C_0 in Bitcoin at $t = 0$, and sells all holdings on the last day $t = T$.
  - At time $t = 0$, the entire capital is used to buy Bitcoin:

    $$B_0 = \frac{C_0}{P_0}, \quad C_1 = 0$$

  - From time $t = 1$ to $t = T - 1$, no trading is performed:

    $$B_t = B_{t-1}, \quad C_t = 0$$

- At time $t = T$, all Bitcoin is sold at price P_T:

$$C_T = B_{T-1} \cdot P_T, \quad B_T = 0$$

- This strategy does not consider the impact of news sentiment on the market at all, so it is highly exposed to market price fluctuations, and its return directly follows the overall price trend:

$$R_T = \frac{P_T}{P_0} - 1$$

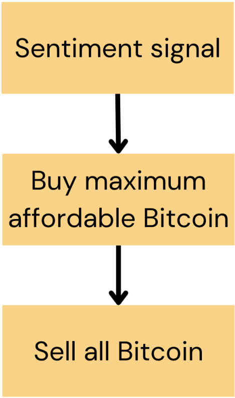

Fig. 3.2 Buy-and-hold strategy flow chart

- **Sentiment strategy 1**

  In the previous section, we have identified the pair of sentiment and return signals with the strongest correlation. This sentiment signal would then be used to mark the indicator and construct the trading signal. In fact, there are plenty of ways to define an indicator. For instance, we can build by averaging the sentiment scores over the past few days, or by using machine learning techniques. However, due to time constraints, we simplified the process here. Since the sentiment signals range from -1 to 1, we set a threshold to construct a trading indicator that guides investors on when to buy, hold, or sell Bitcoin. The sentiment strategy is shown in Fig. 3.3:

  - Let S_t denote the sentiment score at time t, and θ denote the sentiment threshold.
  - Let C_t be the capital at time t, P_t be the Bitcoin price at time t, and B_t be the number of Bitcoins held at time t.

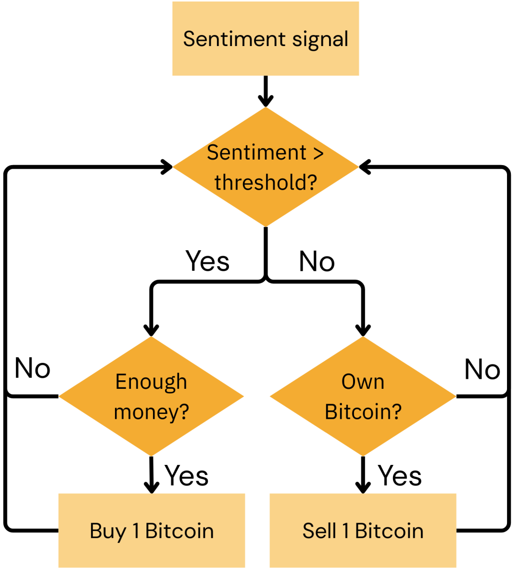

Fig. 3.3 Sentiment strategy 1

- If $S_t > \theta$ and $C_t \geq P_t$, then:

$$C_{t+1} = C_t - P_t, \quad B_{t+1} = B_t + 1$$

- If $S_t \leq \theta$ and $B_t \geq 1$, then:

$$C_{t+1} = C_t + P_t, \quad B_{t+1} = B_t - 1$$

- Otherwise:

$$C_{t+1} = C_t, \quad B_{t+1} = B_t$$

The final return is defined as:

$$R_T = \frac{C_T + B_T \cdot P_T}{C_0} - 1$$

- **Sentiment strategy 2**

  The sentiment strategy 2 is similar to strategy 1, the difference is that it will buy the maximum affordable Bitcoin when the sentiment is positive. The sentiment strategy 2 is shown in Fig. 3.4:

  - Let S_t denote the sentiment score at time t, and θ denote the sentiment threshold.

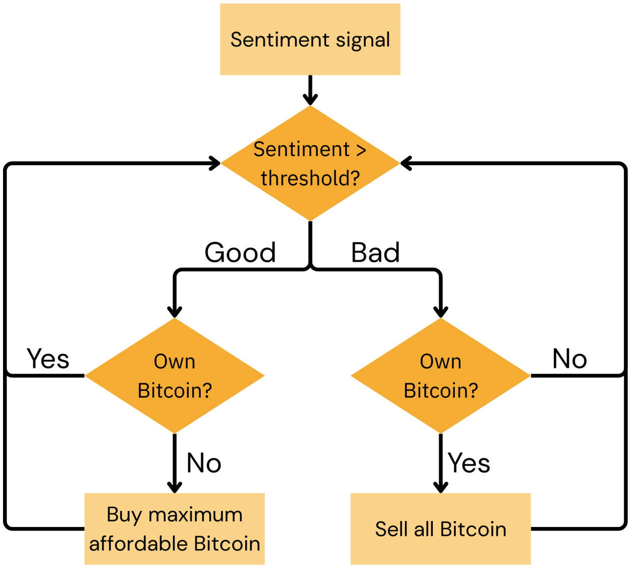

Fig. 3.4 Sentiment strategy 2

- Let C_t be the capital at time t, P_t be the Bitcoin price at time t, and B_t be the number of Bitcoins held at time t.
- If $S_t > \theta$ and $B_t = 0$, then:

$$B_{t+1} = \left\lfloor \frac{C_t}{P_t} \right\rfloor, \quad C_{t+1} = C_t - \left\lfloor \frac{C_t}{P_t} \right\rfloor \cdot P_t$$

- If $S_t \leq \theta$ and $B_t > 0$, then:

$$C_{t+1} = C_t + B_t \cdot P_t, \quad B_{t+1} = 0$$

- Otherwise:

$$C_{t+1} = C_t, \quad B_{t+1} = B_t$$

- The final return is defined as:

$$R_T = \frac{C_T + B_T \cdot P_T}{C_0} - 1$$

CHAPTER 4

Results and Analysis

4.1 Performance of FinBERT

FinBERT's performance was evaluated using two datasets. The first is the CryptoLin dataset, which includes 2,683 manually labeled news articles [3]. The second is a subset of the GDELT 2.0 event database, consisting of 600 news articles that I manually labeled.

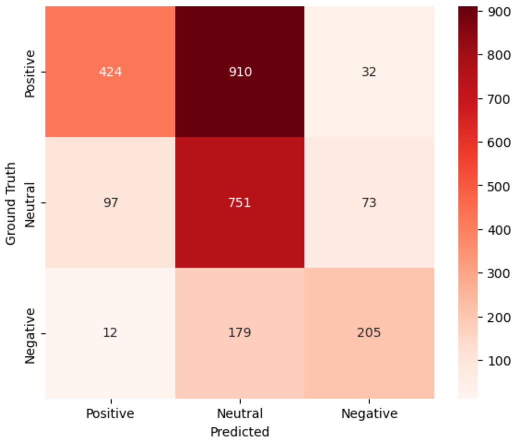

Fig. 4.1 Confusion matrix on CryptoLin dataset. The model has a strong bias toward predicting "neutral"

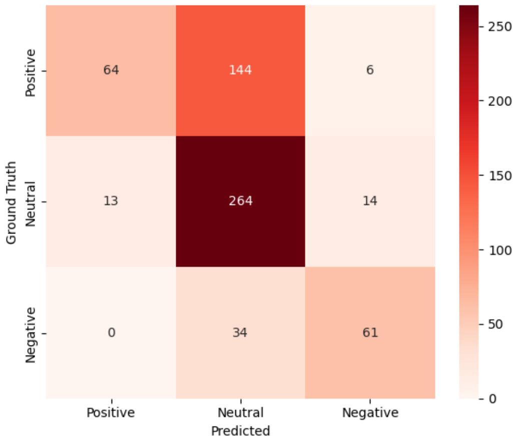

Fig. 4.2 Confusion matrix on GDELT subset. The model still has a neutral bias, but the overall performance is better

According to the confusion matrix for the CryptoLin dataset, 910 positive samples and 179 negative samples were misclassified as neutral. This indicates that the model has a strong bias toward predicting "neutral". FinBERT was originally trained on financial corpora, such as earnings reports and analyst commentaries, which may differ from the CryptoLin dataset. Additionally, the number of positive and negative samples is imbalanced in the dataset, which may lead to the model's poor performance as well.

Although there is less sample in the GDELT subset, the confusion is less extreme. The overall classification improves, particularly for negative and neutral.

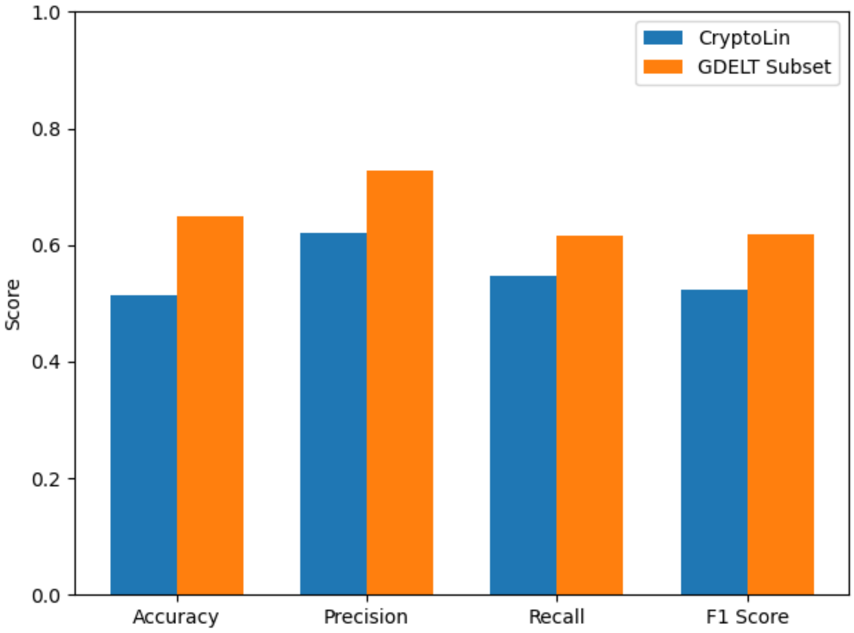

Fig. 4.3 FinBERT performance on CryptoLin dataset and GDELT subset. This shows that FinBERT performs better on the GDELT subset.

Fig. 4.3 clearly shows that FinBERT performs better on the GDELT subset than on the Cryptolin dataset in four evaluation metrics. More importantly, FinBERT was to apply to the GDELT news dataset for sentiment analysis, so we need to use the GDELT subset for retraining process.

4.2 Results of Retraining FinBERT

Since the pre-trained FinBERT achieved an accuracy of 64.8% on the GDELT news subset, it is necessary to retrain the model to enhance its performance.

After preprocessing the news titles, the pre-trained model, training arguments and trainer are defined properly. The results are evaluated using loss and accuracy. The trainer class in Transformers library only computes evaluation metrics on the validation set, so we can't track of log training accuracy during training. The results are shown in Fig. 4.4,

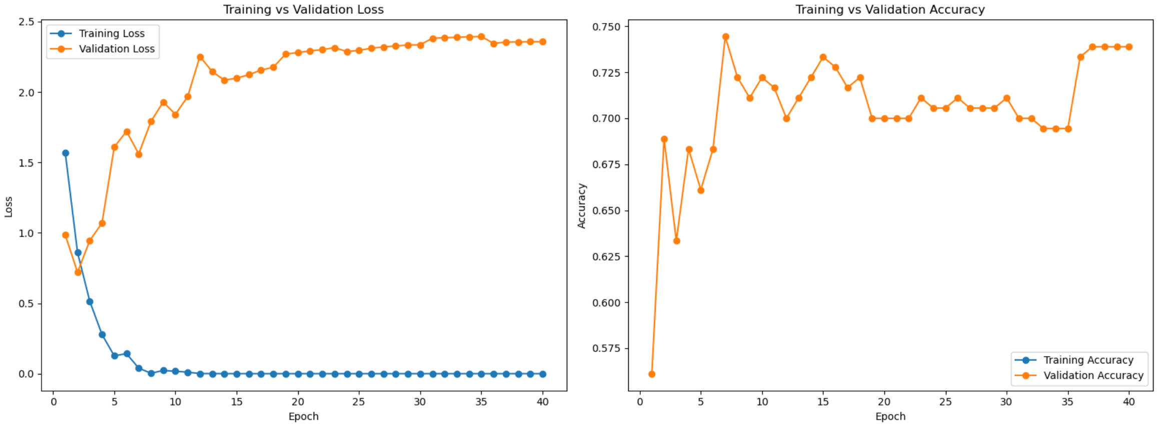

Fig. 4.4 Training, Validation Accuracy and Loss

Training loss drops sharply and almost reaches zero very early, indicating that the model fits the training data extremely well. Validation loss decreases at first but then keep increasing which is a sign of overfitting. Training accuracy is not shown in the plot, but we assume that it reaches a very high level due to the low loss. Validation accuracy increases initially and fluctuates between 70% to 75% and finally stabilizes around 73.8%. These two plots show that the model isn't generalize better on new data, probably because of small and imbalanced dataset and weak regularization. Without time constraints, we can try to address these by increase the dataset size or use dropout layer, increase weight decay etc. However, the overall accuracy improved, indicating that the model still learned meaningful patterns from the training data.

4.3 Results of Sentiment Analysis

4.3.1 Final news dataset

The GDELT news dataset was processed using the steps mentioned in 3.4, the final cleaned dataset was shown in Table 4.1.

Next, we prepared the Bitcoin price dataset, Table 4.2 shows the first three and last three rows of the dataset. The raw Bitcoin dataset contains "Date", four different types of Bitcoin price, and the "Volume" traded on each day.

	Title	**Seen Date**
0	Happy International Trans Day of Visibility | . . .	2021-03-31
1	Chain of Events Update from Blockchain . com Re . . .	2021-03-31
2	CME Develops Micro Bitcoin Futures , Set to Launch . . .	2021-03-31
3	Daily Briefing : Archegos Leveraged Blow - Up . . .	2021-03-31
4	Trekkies Rejoice , Real World Shatner NFTs Now . . .	2021-03-31
⋮	⋮	⋮
243499	The Impact on Brands when Trademarks are Used . . .	2022-04-30
243500	Marvel Avengers Celebrates Ramadan With Free M . . .	2022-04-30
243501	Can Crypto Stay Neutral ?	2022-04-30
243502	Battery life of the Pixel Watch could be fanta . . .	2022-04-30
243503	Data Doctors : Crypto 101 Digital Wallets	2022-04-30

Table 4.1 Overview of GDELT News Headlines with Dates

Date	**Close**	**High**	**Low**	**Open**	**Volume**
2021-03-19	58346.65	59498.38	56643.70	57850.44	49063873786
2021-03-20	58313.64	60031.29	58213.30	58332.26	50361731222
2021-03-21	57523.42	58767.90	56005.62	58309.91	51943414539
...	...	...	...	...	...
2022-05-05	36575.14	39789.28	35856.52	39695.75	43106256317
2022-05-06	36040.92	36624.36	35482.13	36573.18	37795577489
2022-05-07	35501.95	36129.93	34940.82	36042.50	24375896406

Table 4.2 Overview of Bitcoin price dataset

We kept the "Close" column to represent the daily price of Bitcoin and "Date" column to match with the news dataset. By using Eq. 2.14 and Eq. 2.13, and filtering the date to match the specific date period, we enriched the dataset with daily return R_d and Weekly return R_w as shown in Table 4.3.

Date	Closing Price ($)	R_d	R_w
2021-03-31	58918.83	0.30	-4.87
2021-04-01	59095.81	0.49	-1.31
2021-04-02	59384.31	-3.00	-1.92
2021-04-03	57603.89	2.00	3.80
2021-04-04	58758.55	0.51	2.46
⋮	⋮	⋮	⋮
2022-04-26	38117.46	2.95	-0.96
2022-04-27	39241.12	1.36	1.17
2022-04-28	39773.83	-2.93	-8.04
2022-04-29	38609.82	-2.32	-6.65
2022-04-30	37714.88	2.00	-5.87

Table 4.3 Bitcoin closing prices, daily and weekly returns from 2021.3.31 to 2022.4.25

The Bitcoin closing price from 2021.3.31 to 2022.4.25 were plotted in Fig. 4.5. The returns were shown in Fig. 4.6. Overall, Bitcoin exhibited significant price volatility. In the first four months, it showed a fluctuating downward trend, even dropping to around $30000 at some point. During the next four months, it demonstrated a sharp increase, peaking at over $65000. In the last four months of this period, the price declined again with fluctuations, eventually settling at approximately $38000. In general, the price trend was highly unstable and showed a long term downward pattern. The return curve also shows that at certain moments, the weekly return exceeded 30% or dropped below -20%, which are likely outliers. Moreover, the return chart reflects a similar level of volatility as observed in the price trend.

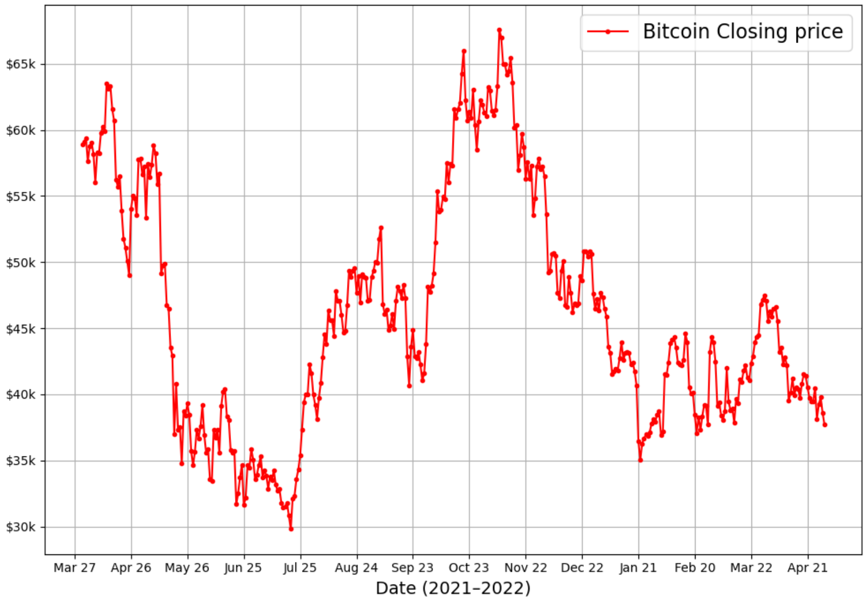

Fig. 4.5 Bitcoin closing price from 2021.3.31 to 2022.4.25, exhibited significant volatility and overall downward trend.

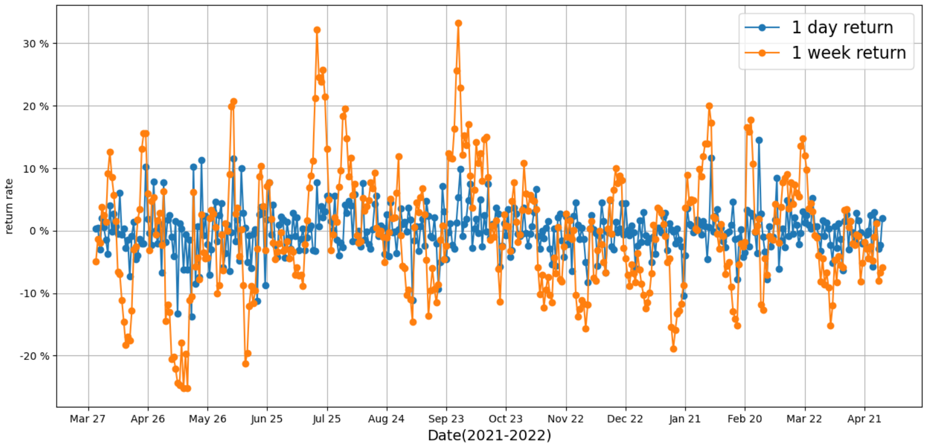

Fig. 4.6 Bitcoin weekly and daily returns from 2021.3.31 to 2022.4.25. Some dates show unusually large fluctuations in return values. The overall volatility is high and lacks any clear pattern.

4.3.2 Building Sentiment Signals

In the previous section, we retrained the FinBERT model using the GDELT 2.0 subset which contains 600 manually labeled news. Here, we apply the model to analyze the sentiment of news headlines from the GDELT news dataset, recording the results as S_{pos}, S_{neu}, and S_{neg} as shown in Table 4.4.

	Title	S_{pos}	S_{neu}	S_{neg}
0	happy international trans day visibility | kfa...	0.09	99.78	0.13
1	chain event update blockchain . com reveals su...	0.08	0.05	99.87
2	cme develops micro bitcoin future , set launch...	0.05	0.09	99.86
3	daily briefing : archegos leveraged blow - : d...	99.20	0.47	0.33
4	trekkies rejoice , real world shatner nfts ava...	1.22	1.99	96.79
⋮	⋮	⋮	⋮	⋮
243499	impact brand trademark used military strategy ...	0.09	99.82	0.09
243500	marvel avenger celebrates ramadan free m . mar...	0.08	99.82	0.11
243501	crypto stay neutral ?	0.13	99.78	0.09
243502	battery life pixel watch could fantastic	0.10	0.28	99.62
243503	data doctor : crypto 101 digital wallet	0.51	80.91	18.57

Table 4.4 Sentiment prediction of news headlines using retrained FinBERT

Next, we need to construct multiple sentiment signals. By using Equations 3.1 and 3.2, the discrete sentiment scores and continuous sentiment score of each news headlines were calculated. Then we applied 3.3, 3.4, 3.5 and 3.6 to the scores to obtain our six sentiment signals. Table 4.5 shows the final daily sentiment signals.

Date	T_d	$\bar{s}_d$	T_c	$\bar{s}_c$	$\text{sgn}(T_d)$	$f(T_c)$
2021-03-31	-297	-0.45	-297.87	-0.45	-1	-1.00
2021-04-01	-315	-0.48	-313.87	-0.48	-1	-1.00
2021-04-02	-269	-0.49	-271.14	-0.49	-1	-1.00
2021-04-03	-236	-0.59	-237.11	-0.59	-1	-1.00
2021-04-04	-172	-0.62	-171.42	-0.62	-1	-1.00
⋮	⋮	⋮	⋮	⋮	⋮	⋮
2022-04-26	-297	-0.46	-298.61	-0.47	-1	-1.00
2022-04-27	-362	-0.47	-359.93	-0.47	-1	-1.00
2022-04-28	-400	-0.47	-400.93	-0.47	-1	-1.00
2022-04-29	-392	-0.46	-394.96	-0.47	-1	-1.00
2022-04-30	-5	-0.42	-5.28	-0.44	-1	-0.99

Table 4.5 Daily Aggregated Sentiment Signals over the date period

4.3.3 Building Return Signals

Based on the Table 4.3, we applied the Equations 3.10 to the 1 day return and 1 week return signals. The results were shown in Table 4.6.

Date	**Price($)**	R_d	R_w	$f(R_d)$	$f(R_w)$	**sgn**(R_d)	**sgn**(R_w)
2021-03-31	58918.83	0.30	-4.87	0.15	-0.98	1	-1
2021-04-01	59095.81	0.49	-1.31	0.24	-0.57	1	-1
2021-04-02	59384.31	-3.00	-1.92	-0.90	-0.74	-1	-1
2021-04-03	57603.89	2.00	3.80	0.76	0.96	1	1
2021-04-04	58758.55	0.51	2.46	0.25	0.84	1	1
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
2022-04-26	38117.46	2.95	-0.96	0.90	-0.45	1	-1
2022-04-27	39241.12	1.36	1.17	0.59	0.52	1	1
2022-04-28	39773.83	-2.93	-8.04	-0.90	-1.00	-1	-1
2022-04-29	38609.82	-2.32	-6.65	-0.82	-1.00	-1	-1
2022-04-30	37714.88	2.00	-5.87	0.76	-0.99	1	-1

Table 4.6 Closing Price and Return Signals over the date period

4.3.4 Calculating Correlation

Now that we have the sentiment signals summarized in Table 4.5 and the return signals summarized in Table 4.6, we can compute the Pearson correlation coefficients between them to identify which sentiment signals have the strongest influence on Bitcoin price movements. The Pearson correlation coefficients were shown in Table 4.7.

	R_d	R_w	$f(R_d)$	$f(R_w)$	**sgn**(R_d)	**sgn**(R_w)
T_d	-0.035	-0.048	-0.020	$\underline{0.018}$	0.005	0.012
$\bar{s}_d$	-0.062	-0.051	-0.037	-0.042	-0.007	-0.041
T_c	-0.034	-0.051	-0.019	0.016	0.006	0.011
$\bar{s}_c$	-0.061	-0.055	-0.036	-0.043	-0.006	-0.040
sgn(T_d)	-0.095	-0.058	-0.091	-0.074	-0.071	-0.066
$f(T_c)$	**-0.096**	-0.059	-0.091	-0.076	-0.072	-0.068

Table 4.7 Pearson Correlation Coefficient Between Sentiment Signals and Price Return Metrics. The largest absolute value is -0.096, between R_d and $f(T_c)$

From the table, we observed that the highest value is 0.018, which appears between T_d and the $f(R_w)$. Meanwhile, the value with the largest absolute magnitude is the $f(T_c)$ correlated with R_d (highlighted in bold as -0.096). This indicates the strongest negative correlation between these two variables. Although their Pearson correlation coefficient is negative, it still suggests that the lower the sentiment score, the more negative the news is classified, the more likely the price of Bitcoin is going to increase. This insight provides an important guideline for constructing our trading signals in the next steps.

In fact, during the course of this study, we initially performed the sentiment analysis on a randomly selected set of 40,000 news headlines from GDELT. Interestingly, using a smaller number of news items tended to produce more positive correlation coefficients. We speculate that this may be due to the high level of noise affecting Bitcoin's price. When the input consists of a small subset of news headlines, the results are more susceptible to random fluctuations or biased samples. However, when processing with a larger volume of news headlines, these noise effects tend to average out. The Bitcoin price is definitely highly volatile and hard to predict. Although this result shows a negative correlation coefficient, it still provides a more reliable and robust signal that reflects the overall market sentiment rather than transient or isolated events.

4.4 Performance of Trading signal

The trading signals were designed based on the sentiment signal $f(T_c)$. A threshold is established to evaluate the optimal trading signals across different strategies. Since the Pearson correlation coefficient is negative, we assume that a buy signal is triggered when the sentiment score is below the threshold, and a sell signal is triggered when the sentiment score is above the threshold.

4.4.1 Buy-and-hold Strategy (B&H)

As the Buy-and-Hold strategy is unaffected by the sentiment of news headlines, its returns align closely with the overall trend of Bitcoin's price. It exhibits high volatility and a generally downward trend over the long term, as shown in Fig. 4.7.

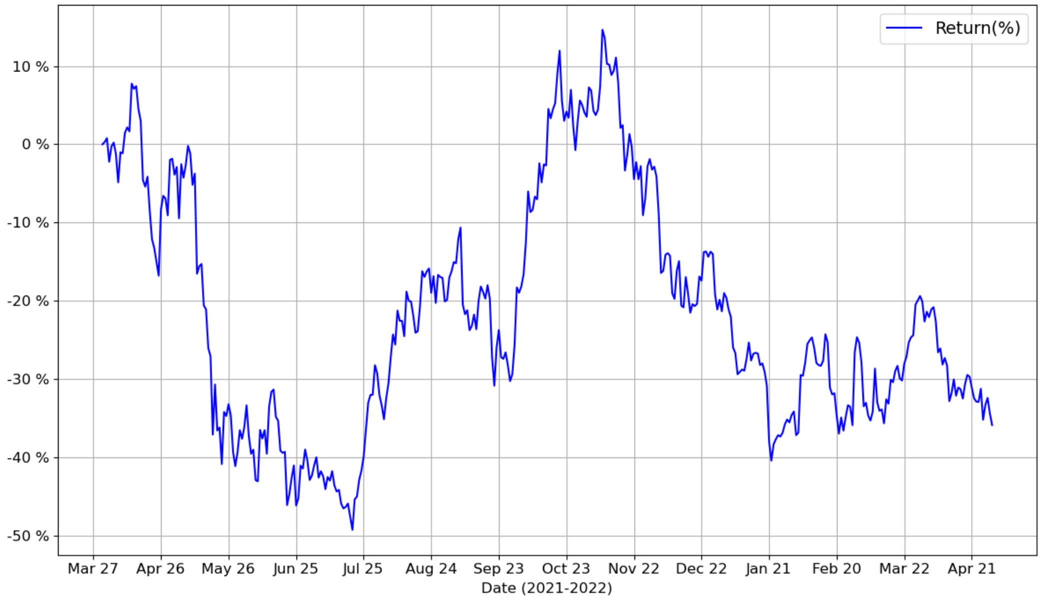

Fig. 4.7 Cumulative Return Over Time For Buy-and-hold Strategy. Similar to Bitcoin price, it shows high volatility and a overall downward trend

4.4.2 Sentiment strategy 1 (S1)

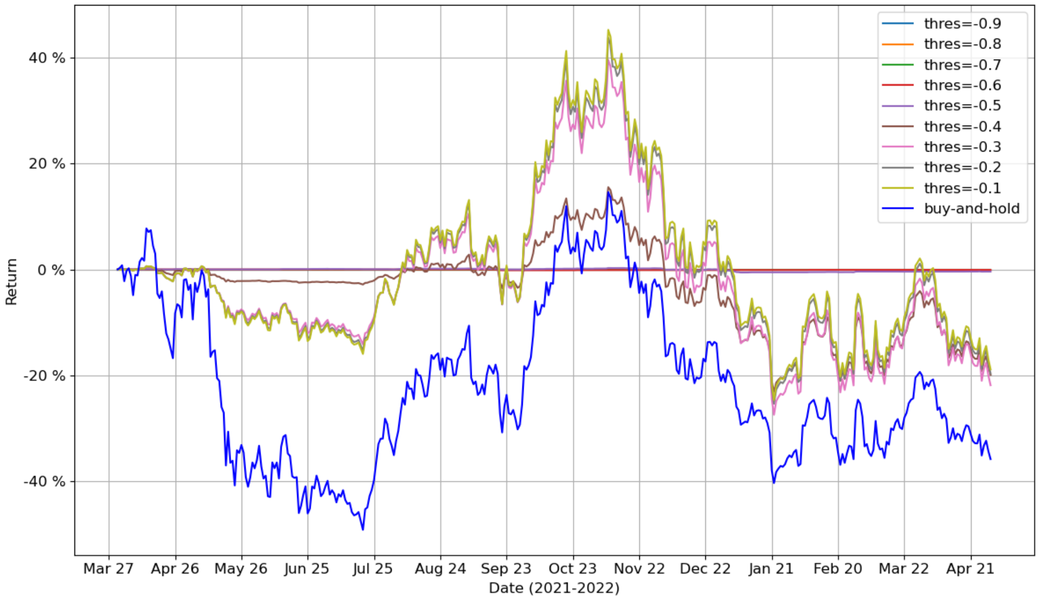

Fig. 4.8 Cumulative Return Over Time For Sentiment Strategy 1. The return signal's best performance is observed at a threshold of -0.1.

For Sentiment Strategy 1, the threshold were set from -0.1 to -0.9, and the cumulative returns were plotted in Fig. 4.8. We use the blue line, which represents B&H, as the baseline.

The figure shows that when the threshold is set too low, the return is around zero all the time. This indicates that there are only few sentiment scores low enough to trigger a buy signal, so there are almost no trading activity. As the threshold increases, the return curve goes above the baseline generally. When the threshold reaches -0.1, the return curve performs the best. Although the return remains below zeros for nearly half of the time, the overall return for the entire time period stays mostly above the baseline, and it even exceeds 40% at one point.

4.4.3 Sentiment strategy 2 (S2)

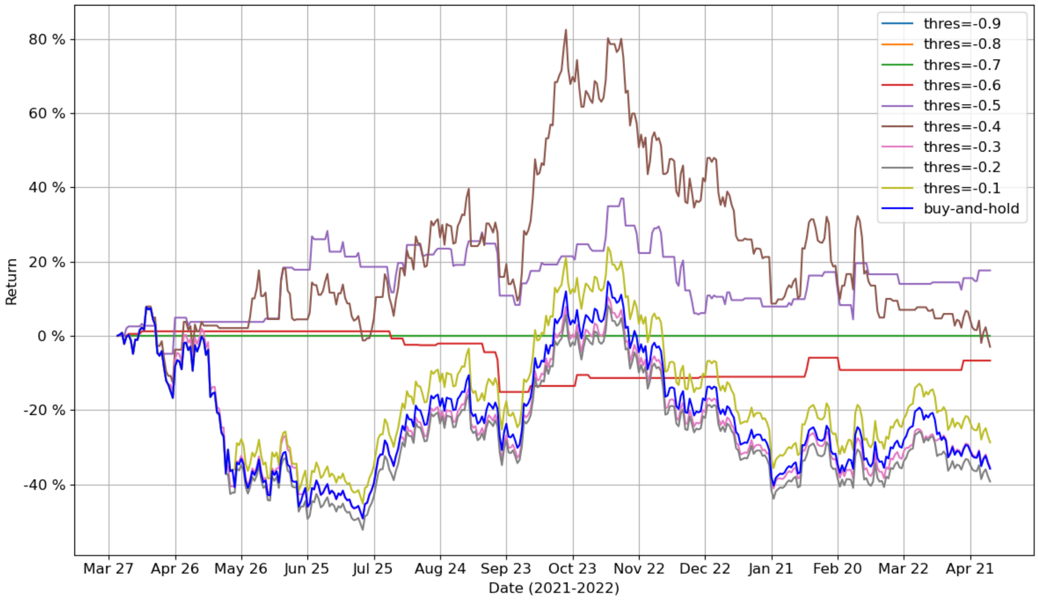

Fig. 4.9 Cumulative Return Over Time For Sentiment Strategy 2. The return signal's best performance is observed at a threshold of -0.4.

In Sentiment Strategy 2 shown in Fig. 4.9, since the entire available capital is used to purchase Bitcoin upon each buy signal, the returns are consequently more sensitive to the sentiment scores than S1. Similarly, the thresholds are set from -0.1 to -0.9, but the results in this case do not show a clear overall pattern.

When the threshold is low (from -0.9 to -0.7), the return remains consistently at zero, which indicates that there are not many buy signals. When the threshold is high (from -0.3

to -0.1), the return curve slightly outperforms the B&H baseline. However, for threshold between -0.4 to -0.6, the return curve fluctuate significantly.

The best performance is observed at a threshold of -0.4, where the returns go consistently above both the baseline and zero throughout most of the period. At some point, it even reached a 80% return.

When the threshold reached -0.5, the return signal's performance is also better than the baseline. Although its maximum return is lower than 80%, it is more stable, with less volatility and staying around just below 20% for a large portion of the time. This demonstrated that given the highly volatile Bitcoin price trends, trading signal based on S2 could not only perform better than the simple B&H, but also it could be more stable and it could yield positive returns.

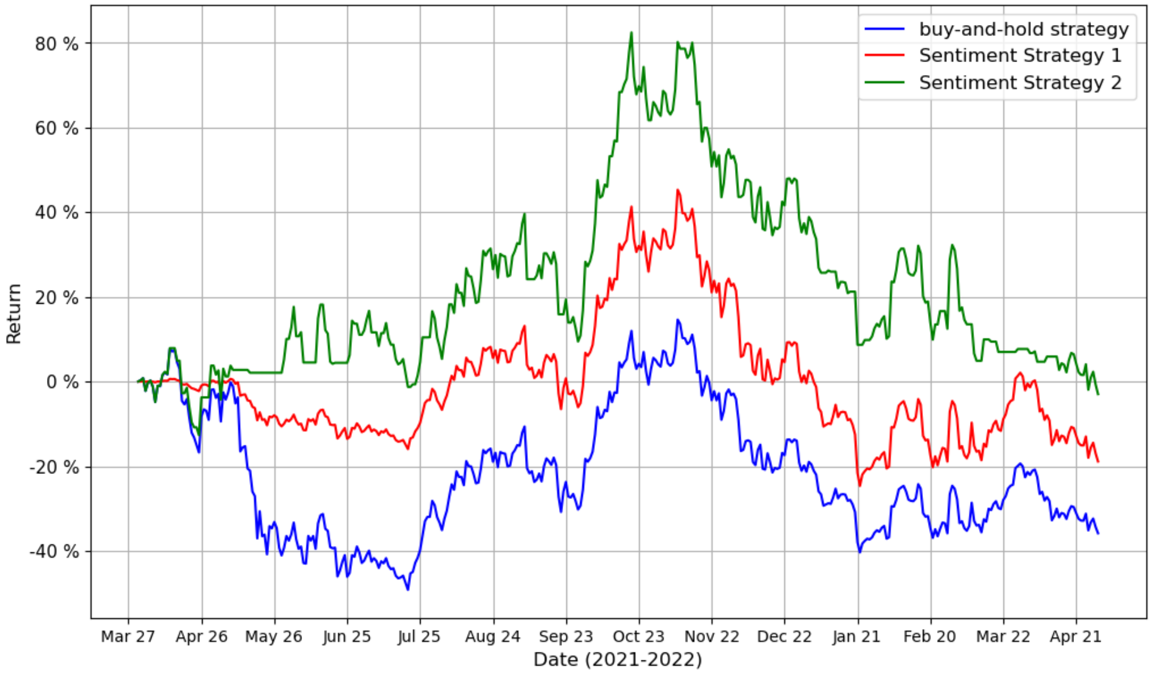

Fig. 4.10 Cumulative Returns Based on different strategies

The best performance of both sentiment strategy and the baseline is shown in Fig. 4.10. The trading signal based on S2 clearly outperforms the others. S2 peaks at around 80% whereas S1 peaks at around 45% and B&H peaks at around 15%, so S2 is approximately 35 percentage points higher than S1, and 65 percentage points higher than B&H at their peaks. S2 fluctuates between 10 to 40%, averaging about 25%. S1 fluctuates from -10% to 30%, averaging about 10%. B&H fluctuates between -50% and 10%, averaging about -25%. Therefore, S2 outperforms B&H by 50 percentage points, and outperforms S1 by 15 percentage points. Overall, S2 exhibits approximately 20 percentage points higher returns than S1 for most of the time. At some point, the difference reaches up to 40 percentage points. Meanwhile, S1 also averages about 20 percentage points higher returns than the B&H baseline.

4.5 Summary of Key Results

Overall, this study offers valuable insights into how FinBERT can be applied and enhanced to analyze sentiment in financial news. It also proposed method about how those insights can help in understanding and predicting trends in the Bitcoin market.

Due to the adaptability of Large Language models, we chose to retrain FinBERT using the GDELT subset rather than the CryptoLin dataset. After retraining, FinBERT's accuracy on the GDELT dataset improved by approximately 9%. Subsequently, we applied FinBERT to analyze news headlines and prepared multiple sentiment signals using different aggregation methods. Meanwhile, we also constructed multiple return signals based on Bitcoin price dataset.

By calculating the Pearson correlation coefficients, the results indicate that the sentiment signal $f(T_c)$ best guides the construction of trading signals. Based on this sentiment signal, we developed two different sentiment-based trading strategies (S1,S2) and compared them against the baseline buy-and-hold strategy (B&H). The results showed that S2 outperforms the others with a consistently higher and positive return curve. S2 has approximately at least 20 percentage points higher returns than S1 and S1 has about 20 percentage points higher returns than the B&H.

CHAPTER 5

Conclusion

5.1 Summary of Work and Key Contributions

This project aimed to analyze the sentiment of the financial news headlines and to use the insights to predict Bitcoin price trends. FinBERT model was evaluated and improved for sentiment analysis. Based on the best sentiment signal, we designed different sentiment-driven trading strategies and constructed trading signals. Finally, we quantitatively analyzed and compared the performance of different trading signals by calculating their returns.

Initial Evaluation of FinBERT

First, the workflow began with the evaluation of the CryptoLin and GDELT subsets, which showed that FinBERT had a strong bias toward predicting neutral sentiment. Mainly because FinBERT was pretrained on financial documents such as earnings reports, which are often conservative in tone. So, FinBERT still needs to be retrained to improve its performance on specific corpus.

Preprocessing Data and Retraining

The next step involved preprocessing of news data extracted from the GDELT dataset. 600 manually labeled GDELT news titles were preprocessed and used to retrain FinBERT.

Preprocessing steps such as stopword removal, punctuation cleaning, and lemmatization were applied to standardize the data before training. FinBERT resulted in a modest but meaningful improvement in validation accuracy, from 64.8% to approximately 73.8%. However, due to the small dataset size, class imbalance etc, there is still a sign of overfitting. Although full generalization was not achieved, the retrained model captured more nuanced sentiment patterns relevant to real world news headlines.

Sentiment Signal Construction

After fine-tuning, the retrained FinBERT model was applied to extract three sentiment scores (S_{pos}, S_{neu}, S_{neg}) from over 240,000 GDELT news headlines. Subsequently, sentiment signals were constructed from the FinBERT outputs using multiple aggregation strategies. These sentiment scores were then aggregated on a daily basis and normalized to construct multiple sentiment signals.

Return Signal Construction and Correlation Analysis

Bitcoin market data was retrieved from yfinance library over the same date period. We computed the daily and weekly return signals and then processed them using normalization functions, resulting in multiple return signals. Pearson correlation coefficient was calculated to determine the strongest relationship between sentiment and returns signals. The $f(T_c)$ signal exhibited the highest absolute Pearson correlation coefficient with return signals, and was therefore selected as the sentiment signal for constructing the trading strategy.

Trading signal Design and Performance Evaluation

Finally, two sentiment-based trading strategies were developed and compared against a buy-and-hold strategy, which served as the baseline. These strategies used the strongest-correlated sentiment signal to generate trading signals based on a sentiment threshold. We tested different threshold values and selected the optimal one for each strategy based on the return curves. The final result showed that, despite the high volatility and overall downward trend of Bitcoin prices during the selected time period, the second sentiment-based strategy consistently outperformed the baseline in terms of return at most time, and yielded a positive overall return. S2 has approximately at least 20 percentage points higher returns than S1 and percentage points higher returns than B&H baseline.

5.2 Real-World Implications and Future Directions

This project shows the important role of sentiment analysis in improving cryptocurrency market prediction accuracy. Especially in the highly volatile and less-regulated markets such as Bitcoin. By calculating the correlation between sentiment signals and return signals, it fills the gap between natural language processing techniques and real-world business decision-making. With further improvements, it has the potential to make greater contributions in both academic research and practical applications.

Research Contributions

This work provides a foundation for further studies in the field of NLP research on financial tasks. It enhanced the performance of FinBERT and applied the retrained model to predict Bitcoin price trends. In addition, it provides empirical evidence about fine-tuning domain-specific language models on unstructured, noisy financial news titles. Besides, the project examines multiple sentiment signal construction techniques and measures their correlation with market returns. This could offer methodological insights for future research on sentiment based financial forecasting.

Practical Applications

This study could benefit the applications in the financial domain. The sentiment analysis method in this work could be applied to generate trading signals and detect early indications of market changes. This would provide the investors with more effective investment strategies and therefore mitigate the potential risks. Additionally, it could be integrated into algorithmic trading systems to improve trading efficiency.

Future Work

We have encountered several challenges during this project, including limited and imbalanced training data, highly noisy and volatile price movements. Future research could incorporate larger and more diverse labeled datasets, such as multilingual news sources. The performance of the FinBERT model could still be further improved to make better sentiment analysis. Other machine learning methods could be implemented as well to construct better sentiment signals and trading signals. Additionally, we could also consider other price-affecting factors such as macroeconomic indicators, social media sentiment, regulatory news, and market liquidity to further enhance the model's predictive power.

Furthermore, due to time limitations, no robustness testing was performed by changing the starting dates for the sentiment analysis and trading strategies. We only selected a starting date to begin our analysis, but sentiment analysis and trading signals are sensitive to dates. In other words, this project needs to perform multiple testings across different time windows to prove its consistency. Future work could include walk-forward analysis to evaluate how stable and reliable the strategy remains under different market conditions.

References

[1] (2019). Module 4: Data-driven innovation. https://innovation.lv/wp-content/uploads/2019/02/4_Module_Eng_pdf.pdf. Accessed: 2025-06-24.

[2] AL-MANSOUR, B. Y. (2020). Cryptocurrency market: Behavioral finance perspective. *The Journal of Asian Finance, Economics and Business*, 7(12):159–168.

[3] Alonso, F. and Ángel Sicilia, M. (2022). Cryptocurrency curated news event database from gdelt. *Research Square (Research Square)*.

[4] Araci, D. (2019). Finbert: Financial sentiment analysis with pre-trained language models. *arXiv:1908.10063 [cs]*.

[5] Asuero, A. G., Sayago, A., and González, A. G. (2006). The correlation coefficient: An overview. *Critical Reviews in Analytical Chemistry*, 36(1):41–59.

[6] Chai, Y., Fang, Y., Peng, Q., and Li, X. (2024). Tokenization falling short: On subword robustness in large language models. page 1582–1599.

[7] Chair, D. (2019). Language models are unsupervised multitask learners | bibsonomy.

[8] Chan, S. W. and Chong, M. W. (2017). Sentiment analysis in financial texts. *Decision Support Systems*, 94:53–64.

[9] Chang, P.-C., Liao, T. W., Lin, J.-J., and Fan, C.-Y. (2011). A dynamic threshold decision system for stock trading signal detection. *Applied Soft Computing*, 11(5):3998–4010.

[10] Chen, Y. and Hao, Y. (2020). A novel framework for stock trading signals forecasting. *Soft Computing*, 24(16):12111–12130.

[11] CoinMarketCal (2024). Coinmarketcal – cryptocurrency calendar. Accessed: 2025-07-03.

[12] Colianni, S., Rosales, S., and Signorotti, M. (2015). *Algorithmic Trading of Cryptocurrency Based on Twitter Sentiment Analysis.*

[13] Gadi, M. F. A. and Sicilia, M. (2024). A sentiment corpus for the cryptocurrency financial domain: the cryptolin corpus. *Language Resources and Evaluation*, 59(2):871–889.

[14] GDELT Project (2015). Gdelt 2.0: Our global world in realtime. https://blog.gdeltproject.org/gdelt-2-0-our-global-world-in-realtime/. Accessed: 2025-06-24.

[15] Gehring, J., Auli, M., Grangier, D., Yarats, D., and Dauphin, Y. N. (2017). Convolutional sequence to sequence learning. In Precup, D. and Teh, Y. W., editors, *Proceedings of the 34th International Conference on Machine Learning*, volume 70 of *Proceedings of Machine Learning Research*, pages 1243–1252. PMLR.

[16] Grefenstette, G. (1999). Tokenization. *Text, Speech and Language Technology*, page 117–133.

[17] Huang, X., Zhang, W., Tang, X., Zhang, M., Surbiryala, J., Iosifidis, V., Liu, Z., and Zhang, J. (2021). Lstm based sentiment analysis for cryptocurrency prediction. *Database Systems for Advanced Applications*, 12683:617–621.

[18] Lee, H., Choi, Y., and Kwon, Y. (2024). Quantifying qualitative insights: Leveraging llms to market predict.

[19] Leetaru, K. and Schrodt, P. (2013). *GDELT: Global Data on Events, Location and Tone, 1979-2012.*

[20] Luo, L. and Chen, X. (2013). Integrating piecewise linear representation and weighted support vector machine for stock trading signal prediction. *Applied Soft Computing*, 13(2):806–816.

[21] Melvin, M. and Yin, X. (2000). Public information arrival, exchange rate volatility, and quote frequency. *The Economic Journal*, 110(465):644–661.

[22] Nakamoto, S. (2008). *Bitcoin: a Peer-to-Peer Electronic Cash System.*

[23] Naveed, H., Khan, A. U., Qiu, S., Saqib, M., Anwar, S., Usman, M., Akhtar, N., Barnes, N., and Mian, A. (2023). A comprehensive overview of large language models.

[24] Niu, Z., Zhong, G., and Yu, H. (2021). A review on the attention mechanism of deep learning. *Neurocomputing*, 452:48–62.

[25] ProsusAI (2021). Finbert: Financial sentiment analysis with bert. https://github.com/ProsusAI/finBERT. Accessed: 2025-06-24.

[26] Raiaan, M. A. K., Mukta, M. S. H., Fatema, K., Fahad, N. M., Sakib, S., Mim, M. M. J., Ahmad, J., Ali, M. E., and Azam, S. (2024). A review on large language models: Architectures, applications, taxonomies, open issues and challenges. *IEEE Access*, 12:26839–26874.

[27] Saud, A. S. and Shakya, S. (2024). Technical indicator empowered intelligent strategies to predict stock trading signals. *Journal of Open Innovation: Technology, Market, and Complexity*, 10(4):100398.

[28] Shobayo, O., Adeyemi-longe, S., Popoola, O., and Ogunleye, B. (2024). Innovative sentiment analysis and prediction of stock price using finbert, gpt-4, and logistic regression: A data-driven approach.

[29] Toraman, C., Yilmaz, E. H., ŞahinuçFurkan, and Ozcelik, O. (2023). Impact of tokenization on language models: An analysis for turkish. *ACM Transactions on Asian and Low-Resource Language Information Processing*, 22(4):1–21.

[30] Urquhart, A. (2016). *The Inefficiency of Bitcoin.*

[31] Valencia, F., Gómez-Espinosa, A., and Valdés-Aguirre, B. (2019). Price movement prediction of cryptocurrencies using sentiment analysis and machine learning. *Entropy*, 21(6).

[32] Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A., Kaiser, , and Polosukhin, I. (2017). *Attention Is All You Need.*

[33] Wang, Y.-A. and Chen, Y.-N. (2020). What do position embeddings learn? an empirical study of pre-trained language model positional encoding. arXiv preprint arXiv:2010.04903.

[34] Ward, M., Beger, A., Cutler, J., Dickenson, M., Dorff, C., and Radford, B. (2013). Comparing gdelt and icews event data. *Analysis*, 21:267–297.

[35] Wójcik-Czerniawska, A. (2019). Cryptocurrency and its influence on global financial markets. *Zeszyty Naukowe Wyższej Szkoły Bankowej w Poznaniu*, 84(1):109–120.

[36] Wolf, T., Debut, L., Sanh, V., Chaumond, J., Delangue, C., Moi, A., Cistac, P., Rault, T., Louf, R., Funtowicz, M., Davison, J., Shleifer, S., Von Platen, P., Ma, C., Jernite, Y., Plu, J., Xu, C., Le Scao, T., Gugger, S., and Drame, M. (2020). *Transformers: State-of-the-Art Natural Language Processing.*

[37] Wüthrich, B., Permunetilleke, D., Leung, S., Lam, W., Cho, V., and Zhang, J. (1998). Daily prediction of major stock indices from textual www data. *HKIE Transactions*, 5(3):151–156.

[38] Yahoo Finance (2025). Bitcoin usd (btc-usd) historical data. Accessed: 2025-07-04.