TITLE: Instructions for the Authors of Papers

Urs A. Hurni

University of Lausanne

Lausanne, Switzerland

urs.hurni@unil.ch

# Abstract

# Introduction

Advances in machine learning and natural language processing (NLP) have transformed how data is utilized, offering new perspectives on market trends and potentially enhancing predictive models for crypto price movements. As we delve deeper into the realm of modern finance, the role of sentiment analysis becomes increasingly crucial. By harnessing sophisticated algorithms and machine learning techniques, analysts and investors are now able to decipher vast amounts of unstructured data from news articles, social media, and financial reports to gauge public sentiment toward various financial instruments, including cryptocurrencies.

Cryptocurrencies, by their very nature, are highly volatile and influenced by a wide array of factors ranging from global economic indicators to regulatory news and even social media trends. Traditional financial models, while still relevant, often fall short when tasked with capturing the swift shifts in investor sentiment that can drastically affect crypto markets. This is where AI and NLP stand out, providing the tools necessary to analyze and interpret the mood and opinions of the market at large, translating this data into actionable insights that can precede market movements

The integration of sentiment analysis into financial decision-making processes marks a significant shift towards data-driven strategies. This approach not only enhances the understanding of market dynamics but also aids in the development of more robust trading systems that can better withstand the unpredictability of the crypto markets. By applying sentiment analysis, traders can identify potential buying or selling signals based on the collective emotions of market participants, thereby gaining a strategic edge.

Moreover, the ability of NLP to process and analyze real-time data allows for a more agile response to market changes. In the fast-paced world of cryptocurrency trading, where prices can fluctuate wildly within minutes, the speed at which data is processed and interpreted is crucial.

However, the application forecasting, machine learning and NLP in sentiment analysis is not without its challenges. One of the primary concerns is the accuracy of the sentiment gauged from various sources. The contextual nuances of language, sarcasm, and misleading information can sometimes skew the analysis, leading to potentially erroneous conclusions. Furthermore, the sentiment itself is highly subjective and can be influenced by temporary external factors that do not necessarily reflect the long-term market trends.

Despite these challenges, the potential benefits of integrating sentiment analysis into cryptocurrency trading and broader financial strategies are significant. It opens up new avenues for research and development within financial technology, encouraging further innovation and refinement of analytical tools

# research Question

*How effectively can market sentiment derived from various sources predict cryptocurrency price movements in the short-term and long-term, and what is the optimal lag time between sentiment shifts and price changes for accurate predictions?*

## Problem

The main issue being addressed is the effect of market sentiment on cryptocurrency prices and how to enhance the prediction of these price movements through sentiment analysis. Cryptocurrencies are notably volatile and highly sensitive to public sentiment, which is quickly spread via social media, news outlets, and other digital channels. The challenge lies in quantitatively evaluating how changes in public mood and opinion influence cryptocurrency values over both short and long terms.

## Objective

quantify the impact of market sentiment on cryptocurrency prices and determine if sentiment alone can predict short-term and long-term price movements. The primary objectives of this project are:

* To quantify the impact of market sentiment on cryptocurrency prices and identify if sentiment only can predict short-term and long-term price movements.
* To develop a predictive model that utilizes sentiment scores from various sources to forecast cryptocurrency price changes
* To examine the lag effect between sentiment shifts and price changes, determining optimal time frames for predicting price movements based on sentiment analysis.

However, due to the lack of free api to get reliable news, the focus of the project will be on an already loaded api to avoid surpassing the free limit and on free apis that may not be as reliable and not as dense as paying for getting out of the free tier. It’s can be easily enhance by just implementing new classes for new apis and just transform the data into the same format as the others.

## Literature Review

Fazlija and Harder [3] explored the use of sentiment from financial news articles to forecast the direction of the S&P 500 index. They applied natural language processing (NLP) to determine sentiment scores from articles and incorporated these scores into predictive models like the random forest classifier. Their findings suggest that news-based sentiment scores are effective in predicting stock movements, underscoring the value of sentiment analysis in financial forecasting.

Souma, Vodenska, and Aoyama [5] investigated the potential of deep learning, particularly transformer models such as BERT, for sentiment analysis in financial texts. Their results showed that these advanced models could accurately gauge sentiment and significantly improve stock price prediction accuracy, suggesting their possible application to cryptocurrencies.

T. Adams and all [4] conducted a study on using Twitter sentiment analysis to predict stock market trends. They concluded that sentiments expressed on social media could be powerful indicators for market movements. Given the active discussion about cryptocurrencies on platforms like Twitter, these methods might also predict trends in cryptocurrency prices effectively.

M. P. Cristescu [6] focused on how news sentiment impacts stock prices, combining sentiment analysis with traditional financial metrics. They discovered that adding sentiment scores to past price data considerably enhances the accuracy of predictions. This method could be particularly useful in the cryptocurrency sector, where market sentiment frequently influences price changes.

Research [7] has also highlighted the effectiveness of random forest classifiers in forecasting stock prices using sentiment scores. By combining several decision trees, this method improves prediction accuracy and could be adapted to analyze cryptocurrency market trends, utilizing sentiment data for better forecasting accuracy.

Despite the potential benefits, sentiment analysis in financial markets faces several challenges. The accuracy of sentiment detection can be affected by the contextual nuances of language, sarcasm, and misleading information. Additionally, sentiment is inherently subjective and can be influenced by transient external factors that do not reflect long-term market trends. These challenges must be addressed to improve the reliability of sentiment-based predictive models [8]

## Scope

For the further analysis we are going to use a set of a pre-loaded dataset, only for testing purposes. That looks like that : For the 24h period : 2024-05-18\_24h\_news\_with\_sentiment.csv and 2024-05-18\_24h\_50meme\_history.csv and for the 30 day period : 2024-04-05\_30d\_news\_with\_sentiment.csv and 2024-04-05\_30d\_50meme\_history.csv.

These datasets are representative of the type of data typically accessible through APIs. However, to adhere to the constraints of the free API tier and minimize data fetching, we are using these static files. While the results presented are based on this specific data, the methodology developed is designed to be adaptable and replicable across different sectors.

# Methodology

The methodology section of this research outlines the systematic steps and processes used to explore the influence of market sentiment on cryptocurrency prices.

## Data Collection

Kaggle

[Crypto Data Hourly Price since 2017 to 2023-10 (kaggle.com)](https://www.kaggle.com/datasets/franoisgeorgesjulien/crypto)

[Crypto News + (kaggle.com)](https://www.kaggle.com/datasets/oliviervha/crypto-news)

## Data Processing

TBD

## Sentiment

A pretrained model specifically FinBERT is used to define the sentiment score. It is a language model based on BERT (Bidirectional Encoder Representations from Transformers), further pre-trained on a financial corpus to better handle the unique vocabulary and expressions used in financial texts [9].

This model, provided by ProsusAI, is finely tuned to interpret the nuances of financial language, making it ideal for analyzing the sentiment embedded within cryptocurrency-related news articles.

## Model Application and Description

### ARIMA

The ARIMA (AutoRegressive Integrated Moving Average) model is widely used for predicting future price changes in cryptocurrencies, leveraging historical price data. This model is particularly effective in scenarios where the data shows trends or seasonal patterns. In cases where such patterns are absent, ARIMA can switch to a simpler, naive approach, typically using the most recent observed value as the forecast. The 'auto' aspect of ARIMA is designed to adapt to the incoming data by automatically selecting the best parameters (p, d, q) to optimize the model. This optimization aims to minimize forecasting errors, guided by criteria such as the Akaike Information Criterion (AIC) and the Bayesian Information Criterion (BIC). ARIMA then predicts the average percentage change in future cryptocurrency prices over a predetermined number of periods.

### Correlation

This analysis measures the strength and direction of the relationship between sentiment scores and subsequent price changes. Pearson correlation coefficients is used to pinpoint the strongest predictive relationships. This method helps in quantifying how sentiment, as expressed through various metrics, aligns with and potentially influences future price changes.

### Linear Regression

Linear regression in this context utilizes the Ordinary Least Squares (OLS) method to analyse the predictive power of sentiment on cryptocurrency price. This model assumes a linear relationship between the independent variable (sentiment scores) and the dependent variable (price changes). The strength of this approach lies in its straightforwardness and the interpretability of its results, which indicate how sentiment scores quantitatively impact price changes in the market.

### Random Forest

The Random Forest model employs the RandomForestRegressor for predicting cryptocurrency price changes using sentiment data. This approach is non-linear, allowing it to model more complex relationships than linear models. Random Forest uses an ensemble learning technique, which involves the construction of multiple decision trees during training and outputs the mean prediction of the individual trees. This method enhances the robustness and accuracy of predictions by reducing the risk of overfitting to the training data and improving performance on unseen data.

## Splitting

Linear regression and random forest models are split into a training set, comprising 75% of the data, and a test set, containing the remaining 25%. This division is designed to ensure a more realistic simulation of model performance in real-world scenarios. By training the models on the larger portion of the data, they can learn and adapt to the underlying patterns and relationships between sentiment scores and cryptocurrency price changes. The smaller test set then serves as a new, unseen dataset for evaluating how well these models can generalize their predictions to new data. This approach helps to mitigate overfitting and provides a clear measure of model efficacy and predictive accuracy outside of the training sample.

## Model Evaluation

Using the previously described data split, various performance metrics such as R-squared, mean squared error (MSE), and accuracy are calculated to determine the effectiveness of the models in predicting cryptocurrency price changes. These metrics are crucial for assessing the precision and reliability of the predictions made by the models.

To provide a thorough evaluation of model performance, two key metrics are computed: R-squared and Root Mean Squared Error (RMSE). R-squared is a statistical measure that represents the proportion of the variance in the dependent variable that is predictable from the independent variables. This metric is especially useful in comparing the fit of different predictive models, as it provides a sense of how well the unseen data points are replicated by the model. A higher R-squared value indicates a better fit to the data.

Root Mean Squared Error (RMSE), on the other hand, measures the average magnitude of the errors between predicted and observed values, providing a clear indication of prediction accuracy. Unlike R-squared, RMSE offers more direct insight into the actual differences in predicted vs. actual values, with a lower RMSE reflecting higher accuracy.

Together, R-squared and RMSE provide a comprehensive view of model performance, highlighting both the accuracy and efficiency of the models in terms of their predictive capabilities. By analyzing these metrics, we can fine-tune the models to improve their predictive accuracy, adjust the complexity of the model if necessary, and better understand the dynamics of the factors influencing cryptocurrency prices.

## Lag Features

As often market reactions to news or events are not immediate a lag analysis is going to be computed for the correlation, linear model, random forest .

Using various time lags to determine the optimal predictive lag period. The core of the analysis involves shifting the sentiment data backward and forward by different time periods to explore how previous and subsequent sentiments correlate with price changes. This shifting helps to test whether the market's reaction to sentiment is immediate or delayed.

* Sentiment scores are shifted backward to align with future price changes, hypothesizing that current sentiment affects future prices. Conversely, sentiment data is also shifted forward to assess if previous price changes can predict future sentiment shifts, capturing the market's reactionary nature.
* For each lag configuration, the correlation between lagged sentiment and subsequent price changes is calculated. Linear regression models are then used to quantify this relationship, providing metrics such as the R-squared value (which measures the proportion of variance in the dependent variable predictable from the independent variable) and root mean squared error (RMSE), indicating the model’s accuracy.

## Predictions

Predictions are made based on the three model described before (ARIMA, Linear Regression and Random Forest) to assess the average price movement.

by using the derived models to forecast future price changes based on recent sentiment data, shifted by the analyzed lags. These predictions are then aggregated to provide a comprehensive outlook on expected price movements.

# Dataset

The dataset for this research is derived from a comprehensive collection of cryptocurrency-related news articles and financial data, spanning multiple sources. The data is obtained through various APIs that provide both historical and real-time data concerning cryptocurrency market dynamics and news sentiment. This information forms the backbone of the analysis, offering insights into the impact of market sentiment on cryptocurrency prices.

converts all date entries in the dataset to a uniform format (*YYYY-MM-DD HH:MM:SS*) for consistent chronological comparisons and data aggregation.

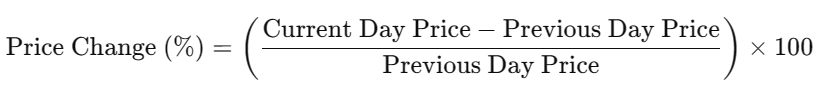
Duplicates are also processed This involves normalizing text fields to remove case sensitivity and trailing spaces and then dropping entries that share the same initial segment of the headline, thus ensuring that each news article is unique.

Cryptocurrency price data is aligned with news data by matching each price entry to the nearest date in the news dataset. This precise alignment is essential for correlating market price movements with specific news events and sentiment analysis.

price data between 0 and 1. This normalization allows the price data to be directly comparable with sentiment scores, which are derived from the content of news articles.

Using the pd.resample method, daily closing prices are computed by taking the last recorded price for each day. This method ensures that the closing price reflects the last market sentiment of the day, which is crucial for daily trend analysis.

Daily price changes are calculated as percentages, which are essential metrics for understanding the day-to-day volatility and the impact of external factors such as news sentiment on the cryptocurrency market. This formula is used:



To investigate the influence of daily news sentiment on subsequent price movements, the price change data is shifted backward by one day using the shift(-1) method. This adjustment aligns the price change data with the corresponding day’s sentiment data, facilitating the analysis of sentiment’s predictive power on price movements.

And finally, The processed price data and sentiment scores are merged based on date indices. This merged dataset includes crucial variables like normalized prices, next-day price changes, and average daily sentiment. The integration of these datasets is vital for exploring the interplay between market sentiment derived from news articles and actual market performance.

The final dataset is reviewed for any missing values that might arise from non-overlapping dates between the news and price data or the data shifting process. Such missing values are addressed to ensure the dataset’s completeness, either by imputation of missing values or by excluding incomplete records, preparing the dataset for robust and reliable analysis.

# Implementation

## Fetching and Processing Data

Multiple API classes fetch cryptocurrency prices and news from various sources. For instance, CryptoDataFetcher retrieves historical price data using the Coinranking API, which is pivotal for analyzing the impact of news sentiment on price movements. [here is a flowchart of the feching and precessing of data through apis]

## Sentiment Analysis

The *SentimentAnalyzer* class uses the ProsusAI/FinBERT model.

The class initializes a tokenizer and model from the pretrained FinBERT model. If a GPU is available, the model is loaded onto it to speed up the computations.

The *predict\_sentiment\_batch* method processes the text data in batches. This is crucial for handling large datasets efficiently. Text data is then tokenized using the Bert Tokenizer, converted to tensors, and processed in batches. Each batch of text is then forwarded through the FinBERT model to obtain sentiment predictions. These predictions are represented as logits, which are converted into probabilities using the softmax function to provide a clearer interpretation of the sentiment values.

The *add\_sentiments\_to\_df* method takes the calculated sentiment scores and integrates them into the original DataFrame by appending a new column. This allows the enhanced DataFrame to contain both the original data and the sentiment analysis results. This integration facilitates in-depth analytical reviews and straightforward visualization within the DataFrame structure.

The resulting DataFrame includes a new column of sentiment scores ranging from 0 to 1, where higher values indicate more positive sentiment. These scores provide a quantitative measure of the sentiment prevalent in each textual entry in the dataset, which can include financial reports, news articles, or social media blurbs [3].

## Predictive Analysis

### Sentiment and price

The Visualizations class offers graphical representations to get an intuitive look on how the data behaves. The packages *Matplotlib* and *Seaborn* are used for this data visualization due to their flexibility. The function *plot\_normalized\_price\_and\_sentiment* is designed to plot both normalized prices and sentiment over time on a dual-axis chart. The dual-axis setup allows viewers to directly compare how changes in sentiment might correlate with shifts in cryptocurrency prices. This visual comparison is crucial for identifying potential predictive relationships.

To make the sentiment trends clearer and less noisy, the sentiment data is smoothed using a rolling mean with a window defined by the window\_size parameter. This smoothing helps in highlighting broader sentiment trends rather than reacting to abrupt fluctuations which may not be significant.

### ARIMA forecast

The *forecast\_prices\_with\_arima* function in the code is designed to utilize the Auto ARIMA model for forecasting future cryptocurrency prices based on historical data. The *pm.auto\_arima* function from the *pmdarima* package, which automatically determines the best parameters (p, d, q) and the type of seasonality. This process involves iteratively exploring different combinations of parameters to find the model that best fits the historical price data.

The function is configured to handle non-seasonal data (*seasonal=False*), which is typical in monthly or daily price data where seasonal patterns are not evident.

The model forecasts prices for a specified number of future periods (*forecast\_periods),* which is essential for short-term trading and investment decisions.

It also calculates confidence intervals for these forecasts, providing an estimate of the potential variability in future prices, which is crucial for risk assessment.

Finally, the function plots these forecasts along with historical prices to visually compare past performance with future expectations. This plot includes the forecasted values and their confidence intervals, which are highlighted to show potential price ranges.

### Feature Engineering – Lag

The analysis function begins by preparing the dataset based on specified time intervals and lag periods. Different lags of sentiment data are analyzed to determine their potential influence on future price changes, allowing us to identify the most predictive time frames.

The code defines a range of lags, both positive and negative, to shift the sentiment data backward and forward in time. For instance, lags are set up as [-7, -5, -2, -1, 0, 1, 2, 5, 7, 10], where negative numbers represent a backward shift (future prediction scenario), and positive numbers represent a forward shift (past analysis scenario).

These lags correspond to the unit of time defined by from\_date and the time variable, which could be in minutes or days depending on the granularity of the data being analyzed.

For each specified lag, the sentiment data is shifted using the *.shift()* method in pandas. This method is applied to the average sentiment column of the combined data DataFrame.

Negative lag shifts data backward in time to see if sentiment on a particular day influences price changes of the previous day, while positive lag shifts data forward to explore if sentiment on a given day reacts to the price change from the day before.

This shifted data (Lagged Sentiment) is then used in subsequent analyses to correlate and predict price changes.

### Correlations

For each lag value, the function calculates the correlation between lagged sentiment and subsequent price changes. This is achieved using pandas' *corr* function to compute the correlation coefficient, which helps determine the strength and direction of the relationship.

The iloc[0, 1] is used to access the correlation value of the off-diagonal element in the resulting correlation matrix, which represents the correlation between the two variables.

The calculated correlation coefficient for each lag is appended to a list correlations for further analysis or reporting.

### Model Implementation and Execution

The dataset is split into training and testing sets using Scikit-learn's train\_test\_split, ensuring that both sets are representative of the overall data distribution. This step is crucial for evaluating the model's performance on unseen data.

The implementation code supports a choice between a linear regression model and a random forest regression model

For the linear regression analysis, we utilize the Ordinary Least Squares (OLS) function from the statsmodels Python library, favored for its comprehensive statistical models and tests. This involves adding a constant to the features for OLS regression, fitting the model on the training set, and then making predictions on the test set. The sentiment scores, adjusted by the specified lag, serve as the independent variable. To accommodate the requirements of the OLS model from statsmodels, a constant is added to the array of independent variables using sm.add\_constant(). This step is crucial as it allows the model to include an intercept in the regression equation.

The price changes, corresponding to the same dates as the lagged sentiment scores, are used as the dependent variable. This setup directly assesses how changes in sentiment at a given time might influence subsequent price movements.

The OLS regression model is executed using the fit() method. This method computes the least squares fit to the data, providing a model that minimizes the sum of the squared differences between observed and predicted values, which is the standard objective in regression analysis.

For the random forest, a more complex approach where a RandomForestRegressor is trained. This model can capture non-linear dependencies better than linear models. The choice of n\_estimators=1000 was made to ensure a robust and stable model by averaging the results of a large number of decision trees, which helps to reduce overfitting and improve predictive accuracy. The specific value of 1000 is often used and it strikes a balance between achieving these benefits and managing computational efficiency. No other hyperparameters were tuned at this stage to maintain simplicity and focus on the primary adjustments.

The lagged sentiment scores serve as the input features. And the corresponding price changes are used as the target variable.

These features are prepared similarly to those in the linear regression analysis but tailored to the requirements of the Random Forest model.

The RandomForestRegressor is initialized with the previously specified parameters. The model is then trained using the .fit() method on the prepared feature and target variables. This training process involves constructing multiple decision trees on various subsets of the data and averaging their individual predictions to form the final output.

### Performance Evaluation

After model training, key performance metrics such as R-squared and RMSE are calculated to assess the fit and predictive accuracy of the model. These metrics are needed for validating the model's effectiveness.

R-squared, Calculated using the r2\_score function from sklearn.metrics, represents the proportion of the variance in the dependent variable that is predictable from the independent variable. A higher R-squared value indicates a better fit of the model to the data.

Root Mean Squared Error (RMSE), computed using the mean\_squared\_error function also from sklearn.metrics, is a measure of the accuracy of the model in predicting the dependent variable.

### Future predictions

For the ARIMA model, percentage Change Calculation are computed from the last known price, alongside the upper and lower bounds of the confidence intervals. These statistics are aggregated to provide a summary of expected price movements and their potential range.

For both linear regression and random forest models, future price changes are predicted using the most up-to-date sentiment data, which is adjusted according to the optimal lags identified during the model training phase. Once the sentiment data is prepared, it is fed into the fitted models to forecast future price movements. This process uses the model's .predict() method, where the input features are the lag-adjusted sentiment scores.

# Results

The result of the sentiment analysis project illustrates several key insights into the relationship between news sentiment and cryptocurrency price movements.

A graph with lines and numbers

Description automatically generatedFirstly, the visualizations generated from the Visualizations class clearly highlighted the fluctuations in cryptocurrency prices in response to shifts in news sentiment. These plots showed that significant news events often coincided with movements in cryptocurrency prices, suggesting a potential correlation between news sentiment and market behavior.

A graph with a red line and a line

Description automatically generatedThrough quantitatively assessing the correlation a moderate relationship was found in certain instances

A graph with colored dots and a red line

Description automatically generatedcorrelation of 0.3 for Monthly and correlation of 0.5 for daily

However, note that the correlations are generally weak and mixed in direction, indicating no strong or consistent predictive relationship between sentiment and price changes.

For the linear regression low R-squared values and high RMSE across all models was observed which suggest that a linear approach may not be the best fit for predicting price changes based on sentiment.

For the random forest model strong Model Performance was observed. The high R-squared values across different lags highlight the Random Forest model's effectiveness in capturing the complexities of the relationship between sentiment and price changes, which was not apparent with the linear models.

The lower RMSE values across all lags compared to those from the linear model suggest that the Random Forest model provides more accurate and reliable predictions.

Additionally, the resampling and lag analysis techniques helped to uncover that the effect of sentiment on price changes is not immediate but rather delayed, with the strongest correlations observed at specific lags depending on the cryptocurrency and the nature of the news.

Furthermore, the data suggests an inverse relationship in most cases where higher sentiment correlates with lower prices immediately or shortly after, which is contrary to typical expectations. This might imply that high sentiment could be a reaction to peak prices which then correct downwards or this could imply overreaction in prices to positive sentiment, leading to \*corrections\*. This is a often reccurring pattern in the meme coin in crypto

# Conclusion

In conclusion, the results from this project demonstrate the potential of using automated sentiment analysis combined with historical price data to predict future price movements in the cryptocurrency market. However, it also highlights the complexity and challenges in accurately modeling and predicting such volatile and sentiment-driven markets. The findings suggest that while sentiment is a significant factor, it must be integrated with other market indicators for more reliable predictions.

# References

[1] L. Chappex, "Interview with Richard Peterson, CEO of MarketPsych," Swissquote, [Online]. Available: https://en.swissquote.lu/international-investing/investing-ideas/interview-richard-peterson-ceo-marketpsych. [Accessed: May 21, 2024].

[2] L. Chappex, "Market mood dissected by AI," Swissquote, [Online]. Available: https://www.swissquote.com/en-ch/market-mood-dissected-ai. [Accessed: May 21, 2024].

[3] B. Fazlija and P. Harder, “Using financial news sentiment for stock price direction prediction,” \*Mathematics\*, vol. 10, no. 13, p. 2156, 2022. [Online]. Available: https://doi.org/10.3390/math10132156

[4] T. Adams, A. Ajello, D. Silva, and F. Vazquez-Grande, “More than words: Twitter chatter and financial market sentiment,” \*Finance and Economics Discussion Series\*, vol. 2023-034, Board of Governors of the Federal Reserve System, Washington, 2023. [Online]. Available: https://doi.org/10.17016/FEDS.2023.034

[5] W. Souma, I. Vodenska, and H. Aoyama, “Enhanced news sentiment analysis using deep learning methods,” \*Journal of Computational Social Science\*, vol. 2, pp. 33-46, 2019.

[6] M. P. Cristescu, D. A. Mara, R. A. Nerișanu, L. C. Culda, and I. Maniu, “Analyzing the impact of financial news sentiments on stock prices—a wavelet correlation,” \*Mathematics\*, vol. 11, no. 23, p. 4830, 2023. [Online]. Available: https://doi.org/10.3390/math11234830

[7] K. Mishev, A. Gjorgjevikj, I. Vodenska, L. Chitkushev, and D. Trajanov, "Evaluation of Sentiment Analysis in Finance: From Lexicons to Transformers," IEEE Access, vol. 8, pp. 131662-131682, 2020.

[8] A. K. Nassirtoussi, S. Aghabozorgi, T. Y. Wah, and D. C. L. Ngo, "Text mining for market prediction: A systematic review," Expert Systems with Applications, vol. 41, no. 7653-7670, 2014.

[9] D. T. Araci, “FinBERT: Financial Sentiment Analysis with Pre-trained Language Models,” \*arXiv preprint arXiv:1908.10063\*, 2019. [Online]. Available: https://doi.org/10.48550/arXiv.1908.1006