Predicting stock market prices leveraging investor sentiments

Authors:
Ondrej Gajdos, Mahtab Mirhaj, Venugopal Srinivas

May 27, 2024

UPPSALA
UNIVERSITET

1 Introduction

Predicting stock market trends has always been a challenging yet rewarding task for researchers and investors. Conventional prediction approaches have primarily depended on statistical models and technical indicators based on the past stock prices. Unfortunately, sometimes these models fail to capture the complexity and the dynamics of stock markets. The rise of social media platforms, particularly Twitter, has marked a new era in stock market predictions. The sentiments of the investors are based on public perceptions and news feeds. Using Natural Language processing (NLP) to extract investor sentiments on social media platforms such as Twitter gives an additional dimension in predicting the stock market trends.

In this project we aim to predict the stock market trends analyzing Twitter sentiments and using Long Short Term Memory (LSTM) networks. Pre-trained Transformer model RoBERTa [1] was used to analyze the Twitter dataset and assign a score (positive, neutral, negative) to every tweet. This was concatenated with the stock prices of the company tickers in predicting their movements.

2 Background

2.1 Introduction

Long Short-Term Memory (LSTM) networks are a type of Recurrent Neural Network designed to capture long-term dependencies in sequential data.

2.2 LSTM

Long Short-Term Memory (LSTM) is an architecture for artificial recurrent neural networks (RNNs) employed in deep learning. Unlike standard feedforward neural networks, LSTMs include feedback connections, enabling them to leverage temporal dependencies in data sequences. The 1 picture, shows the difference between RNN and LSTM.

LSTMs are specifically engineered to address the problems of vanishing or exploding gradients that can arise when training traditional RNNs on sequential data. Consequently, they are particularly effective for tasks involving sequential data, such as natural language processing (NLP), speech recognition, and time series forecasting.[14]Long Short-Term Memory (LSTM) networks, a special type of Recurrent Neural Network (RNN), have the capability to learn long-term dependencies. They were introduced by Hochreiter et al. (1997) [9] and have since been refined and popularized by numerous researchers.[2] LSTM networks are designed to address the vanishing gradient problem that afflicts traditional RNNs, achieved through a unique gating mechanism that controls the flow of information. [5] An LSTM unit is composed of three key gates: the Forget Gate, the Input Gate, and the Output Gate. The Forget Gate decides

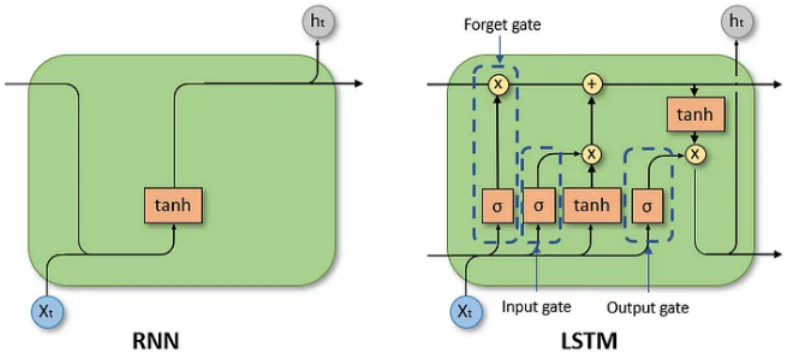

Figure 1: RNN vs LSTM [6]

which information to discard from the cell state, while the Input Gate determines which input values should be used to update the cell state. Meanwhile, the Output Gate governs the output based on the cell state. [5]This intricate gating mechanism enables LSTMs to retain information over long sequences, making them adept at processing time series data such as stock prices. 2

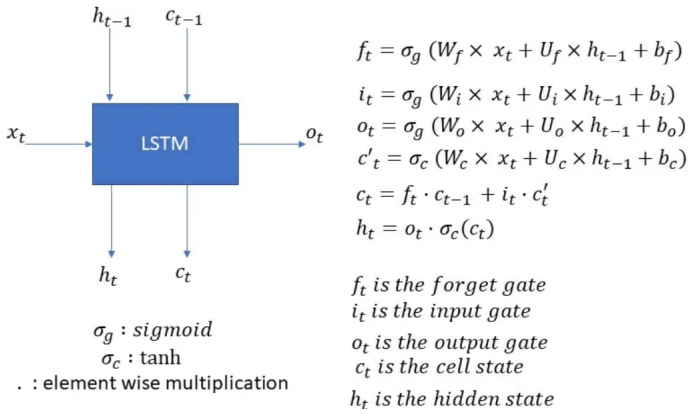

Figure 2: LSTM input outputs and the corresponding equations for a single timestep[13]

2.3 Transformer

Contrary to RNNs and LSTMs, Transformers do not follow a sequential data processing approach. Instead, they utilize self-attention mechanisms to simultaneously assess the significance of various words within an input sequence. This

method enables parallel processing and effectively captures relationships between distant words. [14] The architecture of Transformers incorporates several critical elements. First, the self-attention mechanism allows the model to focus on different parts of the input sequence simultaneously, improving its capacity to capture long-range dependencies. Second, positional encoding provides information about the position of each word in the sequence, aiding the model in understanding the sequential order of the input. Finally, the feed-forward network processes the weighted inputs obtained through self-attention, generating the final output by integrating information from across the sequence.[12]

2.4 Theory behind Sentiment Analysis and its Application

For sentiment analysis, pre-trained Transformer models such as the [1]"twitter-roberta-base-sentiment-latest" from Hugging Face have been used. These models have been fine-tuned on extensive datasets to classify the sentiment which are expressed in text accurately. In this project, we utilize the "twitter-roberta-base-sentiment-latest" to perform sentiment analysis on Twitter data.

3 Dataset

The dataset[11] contains 80k tweets of the top 25 most watched stocks on Yahoo Finance from 2021-09-30 to 2022-09-30. It also includes the stock market price and volume data for the respective dates and stocks. The dataset comprises of 2 *csv* files : *stock_tweets.csv* and *stock_yfinance_data.csv*. The *stock_tweets.csv* consists of 4 columns comprising of *Date* of the tweet, full text of the *Tweet*, *Stock Name* i.e. stock ticker name and *Company Name* i.e. full company name of the corresponding tweet and stock ticker. The structure of *stock_tweets.csv* is shown in Fig. 3

	Date	Tweet	Stock Name	Company Name
0	2022-09-29 23:41:16+00:00	Mainstream media has done an amazing job at br...	TSLA	Tesla, Inc.
1	2022-09-29 23:24:43+00:00	Tesla delivery estimates are at around 364k fr...	TSLA	Tesla, Inc.
2	2022-09-29 23:18:08+00:00	3/ Even if I include 63.0M unvested RSUs as of...	TSLA	Tesla, Inc.
3	2022-09-29 22:40:07+00:00	@RealDanODowd @WholeMarsBlog @Tesla Hahaha why...	TSLA	Tesla, Inc.
4	2022-09-29 22:27:05+00:00	@RealDanODowd @Tesla Stop trying to kill kids,...	TSLA	Tesla, Inc.

Figure 3: Structure of *stock_tweets.csv*

The *stock_yfinance_data.csv* contained 8 columns with *Date*, *Open* i.e Opening price of the stock, *High* i.e. Highest trading price of the stock, *Low* i.e. the lowest trading price of the stock, *Close* i.e. the closing price of the stock, *Adj Close* i.e. the adjusted closing price of the stock, *Volume* i.e. the volume of the stocks traded on the day & *Stock Name* i.e. stock name as displayed in the ticker of the corresponding date. The structure of *stock_yfinance_data.csv* is shown in Fig. 4.

	Date	Open	High	Low	Close	Adj Close	Volume	Stock Name
0	2021-09-30	260.333344	263.043335	258.333344	258.493347	258.493347	53868000	TSLA
1	2021-10-01	259.466675	260.260010	254.529999	258.406677	258.406677	51094200	TSLA
2	2021-10-04	265.500000	268.989990	258.706665	260.510010	260.510010	91449900	TSLA
3	2021-10-05	261.600006	265.769989	258.066681	260.196655	260.196655	55297800	TSLA
4	2021-10-06	258.733337	262.220001	257.739990	260.916656	260.916656	43898400	TSLA

Figure 4: Structure of *stock_yfinance_data.csv*

stock_tweets.csv was used as input to the pre-trained Transformer[1] model to obtain sentiment scores (positive, neutral, negative) of every tweet. The sentiment scores were combined with the *stock_yfinance_data.csv* and used as the inputs to the LSTM. The sentiment scores of tweets are as shown in 5

	Date	Tweet	Stock Name	Company Name	positive	neutral	negative
0	2022-09-29 23:41:16+00:00	Mainstream media has done an amazing job at br...	TSLA	Tesla, Inc.	0.0573	0.2339	0.7088
1	2022-09-29 23:24:43+00:00	Tesla delivery estimates are at around 364k fr...	TSLA	Tesla, Inc.	0.3148	0.6753	0.0099
2	2022-09-29 23:18:08+00:00	3/ Even if I include 63.0M unvested RSUs as of...	TSLA	Tesla, Inc.	0.0785	0.8904	0.0311
3	2022-09-29 22:40:07+00:00	@RealDanODowd @WholeMarsBlog @Tesla Hahaha why...	TSLA	Tesla, Inc.	0.0120	0.0724	0.9156
4	2022-09-29 22:27:05+00:00	@RealDanODowd @Tesla Stop trying to kill kids,...	TSLA	Tesla, Inc.	0.0052	0.0452	0.9497

Figure 5: RoBERTa model sentiment scores for Tweets

4 Implementation

We first utilized a pre-trained Transformer model designed to assess the sentiment of Twitter posts. The sentiment features derived from these posts, combined with historical prices, were then fed into a Long Short-Term Memory (LSTM) model to predict future prices.

4.1 Data Preparation

The dataset was quite clean as the author had already performed data cleansing, including handling posts discussing multiple stocks by leaving the stock name column empty. We decided to use all the available features in our analysis.

4.2 Tools and Technologies

We used Python as the programming language and TensorFlow for the machine learning tasks. Predicting sequences isn't a standard supervised learning problem, so we employed specialized prediction methods:

4.2.1 Recursive Prediction

The neural network generates outputs for input features, which are fed back into the network to generate subsequent predictions, continuing this process for the desired future time frame.

4.2.2 Sliding Window Method

We used a sequence of multiple prices to predict multiple future prices, adjusting the number of output nodes to match the target value vector's length. For each prediction, the closing price, or the stock price at the end of the trading day, was used.

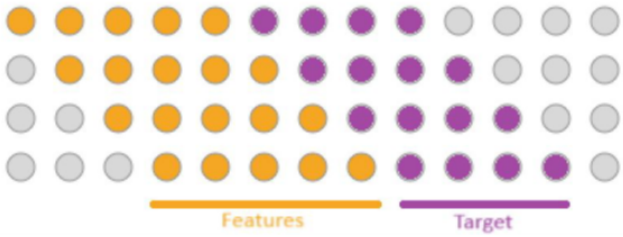

Figure 6: Sliding window method illustration.

4.3 Model Selection and Development

We chose a Transformer architecture for sentiment assessment [3] due to its state-of-the-art performance in sentiment analysis. Although LSTMs are not the latest in sequential prediction, we selected them because they fit well with our goals.

4.3.1 Model Architecture

The model architecture consisted of three LSTM layers followed by 25 dense nodes and the output layer, with the number of nodes based on the length of the prediction.

4.3.2 Training

The dataset was split into training and test sets based on the prediction method. The goal was to maximize training data. For recursive prediction, only one data point was necessary in the test set. For the sliding window method, more test data was required when the number of output nodes was smaller than the number of prices we wanted to predict. Data was rescaled using a min-max scaler before feeding it into the LSTM.

Parameter	Values
window_sizes	5, 10, 15
prediction_lengths	5, 10, 15
lstm_units_options	50, 100
batch_sizes	32, 64
dropout_rates	0.0, 0.2
predict_days	5, 10, 20

Figure 7: Hyperparameters used for training.

We trained the model for approximately 12 hours.

4.4 Evaluation and Metrics

Mean Squared Error (MSE) was used for evaluation due to its suitability for time series data.

4.4.1 Recursive Prediction Results

Initial results showed that recursive prediction was ineffective.

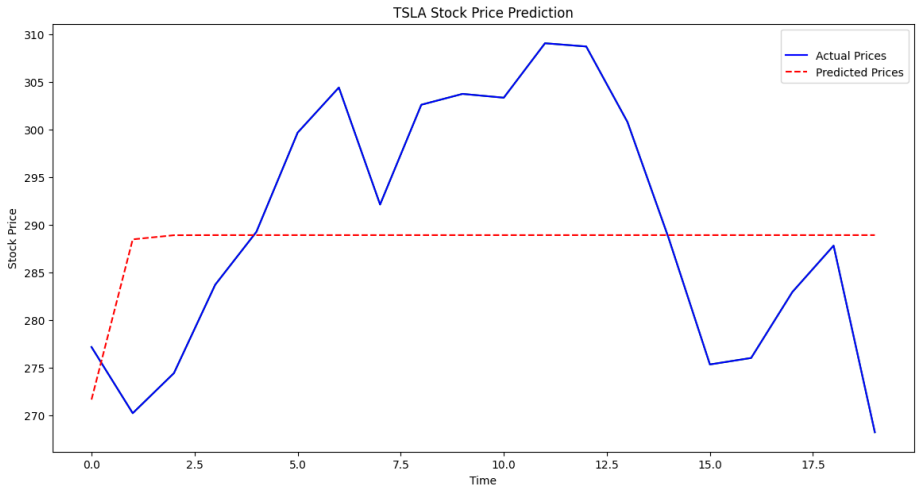

Figure 8: Recursive prediction results.

4.4.2 Sliding Window Results

The best hyperparameters for 20-day predictions turned out to be:

Table 1: Best hyperparameters for sliding window method.

Hyperparameter	Values
Window Sizes	10
Prediction Lengths	15
LSTM Units	50
Batch Sizes	32
Dropout Rates	0

The sliding window method performed better and was able to overcome random walk systematically in the case of META stock. According to the Efficient Market Hypothesis, markets are efficient, and all information is already reflected in stock prices, which explains why our results were not particularly surprising [7].

Table 2: Model performance comparison using sentiment features.

Stock Name	Sentiment Used	Models Trained	MSE of Model	MSE of Random Walk
TSLA	TRUE	50	18.350067	15.877408
META	TRUE	10	9.815262	12.054089
AMZN	TRUE	10	16.294937	5.771485
AAPL	TRUE	10	11.1573	4.353738
GOOG	TRUE	10	10.750604	5.154606
NFLX	TRUE	10	20.153626	8.664131

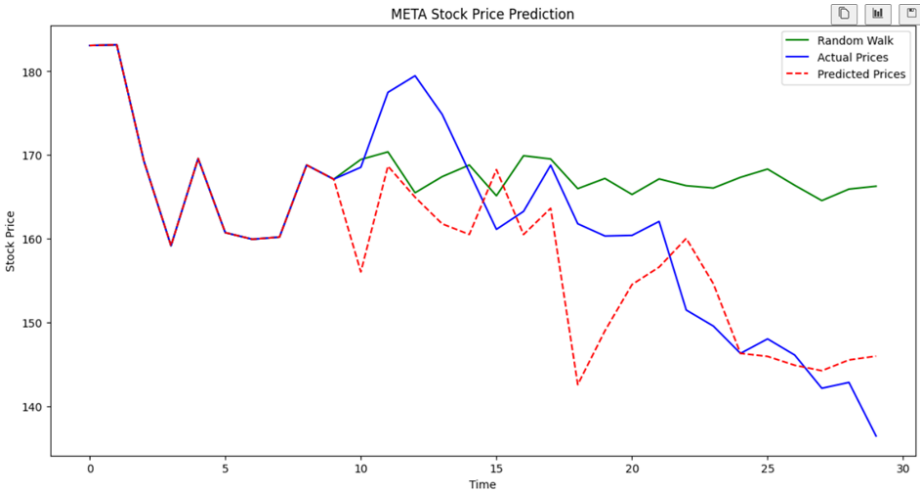

Figure 9: Prediction results for META stock.

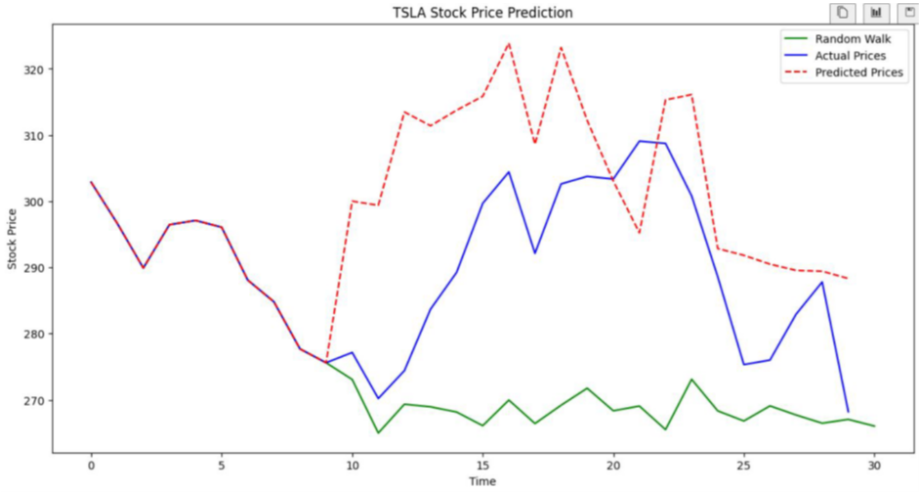

Figure 10: Prediction results for TSLA stock.

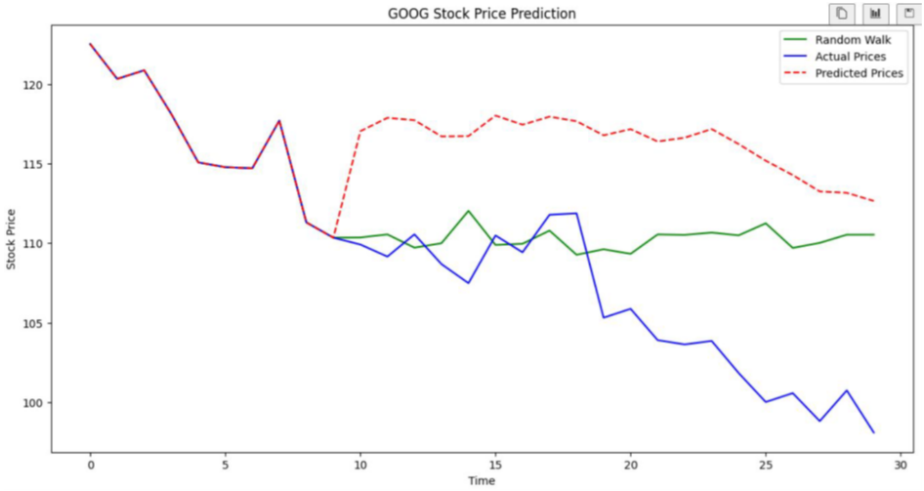

Figure 11: Prediction results for GOOG stock.

One possible explanation for our model performing better on META is due to its higher volatility, which means a wider range of prices. Another reason could be that consumer investors interested in Tesla tend to react to Twitter posts more frequently than those interested in other stocks. It might also be a combination of both factors.

4.5 Comparison and Other Findings

We compared models using sentiment analysis with those relying solely on historical prices and trading volumes. The aim was to determine when sentiment analysis improves predictions and when traditional methods suffice. We observed that in each case, the models using sentiment performed better than those that did not, except for Netflix.

Table 3: Model performance comparison with and without sentiment analysis.

Stock Name	Sentiment Used	Models Trained	Average MSE of Model	Average MSE of Random Walk
TSLA	TRUE	50	18.350067	15.877408
TSLA	FALSE	50	23.878203	15.94572
META	TRUE	10	9.815262	12.054089
META	FALSE	10	10.336497	12.018995
AMZN	TRUE	10	16.294937	5.771485
AMZN	FALSE	10	17.722629	5.667498
AAPL	TRUE	10	11.1573	4.353738
AAPL	FALSE	10	12.378398	4.333616
GOOG	TRUE	10	10.750604	5.154606
GOOG	FALSE	10	11.532473	5.168091
NFLX	TRUE	10	20.153626	8.664131
NFLX	FALSE	10	18.228342	8.709884

4.5.1 Permutation Importance

We evaluated feature importance using permutation importance, which measures the change in model performance when feature values are shuffled. This method revealed that some features were more important than others, with smaller permutation MSE indicating higher importance.

Table 4: Tesla average importance vector of 19 models.

Feature	Positive	Neutral	Negative	Open	High	Low	Close	Adj Close	Volume
Permutation MSE	-0.0000043	0.0000024	0.0000003	-0.0000205	-0.0000218	-0.0000358	-0.0000327	-0.000036	-0.0000003

4.5.2 Correlation of Sentiment and Price

We explored the correlation between sentiment time series and closing prices, noting a strong correlation in Tesla and Netflix stocks. This suggests that sentiment analysis can be a useful tool for predicting stock market trends.

Table 5: Correlation of sentiment and price.

Ticker	Positive	Neutral	Negative
META	0.087405	0.096436	-0.217122
AMZN	0.156345	-0.036205	-0.184361
AAPL	0.099462	-0.019465	-0.108686
GOOG	0.030823	-0.054853	0.023279
NFLX	0.231488	-0.031491	-0.24741
TSLA	0.343456	-0.029938	-0.416675

5 Discussion

Exploring additional models such as Generative Adversarial Networks (GANs) could enhance predictions. Although GANs seemed promising, they were deemed too advanced for our current scope. Adding technical indicators and exploring broader hyperparameter combinations in models like Transformers might also improve performance.

Due to time constraints, we were unable to conduct proper experimentation, although it took around 15 hours for the computer to generate the above results.

6 Related works

Numerous research efforts have examined the convergence of financial markets, sentiment analysis, and machine learning, which are showing the ability of these techniques to accurately forecast stock market trends. In this part, we discuss some other similar projects to this field.
A notable study conducted by Bollen et al. (2011) [4]explored how public sentiment derived from Twitter impacts the stock market. By employing a mood tracking tool, they assessed the sentiment of tweets and identified significant correlations between collective mood states and the Dow Jones Industrial Average (DJIA) index, and indicated that sentiment analysis could potentially forecast market movements.
Moreover, in the field of combining the sentiment analysis with machine learning models, Xu and Cohen (2018) [15]suggested a hybrid method that merged Twitter sentiment with several machine learning models to forecast stock market movements. Their research used Support Vector Machines (SVM) and Random Forest algorithms, and incorporates sentiment scores from Twitter alongside historical stock prices. They observed enhanced predictive performance, underscoring the benefit of integrating sentiment data with traditional financial metrics.

In another research effort, Huang et al. (2018) [10] combined social media sentiment analysis with LSTM networks to predict stock market trends. They fed sentiment scores which are derived from Twitter data, along with historical stock prices, into the LSTM model. Their results showed an enhancement in prediction accuracy, and confirmed the effectiveness of integrating sentiment analysis with deep learning models.
Another method involves using Generative Adversarial Networks (GANs) to forecast stock market trends. In a study by Zhang et al. (2019) [8], GANs generated synthetic market data, which, when combined with actual market data, enhanced the training set for predictive models. The researchers incorporated Twitter sentiment analysis with the GAN-generated data to more effectively capture market sentiments and dynamics.
To conclude, the convergence of sentiment analysis and machine learning techniques continues to demonstrate significant potential in accurately predicting stock market trends.

7 Conclusion

Even though our models weren't able to systematically overcome random walks and the significance of certain features wasn't higher than others, we have shown that models perform better when using sentiment features. This is especially true for stocks where sentiment features are correlated with the price of the stock.

8 References

[1] cardiffnlp/twitter-roberta-base-sentiment-latest · Hugging Face — huggingface.co. https://huggingface.co/cardiffnlp/twitter-roberta-base-sentiment-latest. [Accessed 27-05-2024].

[2] Deep Learning — deeplearningbook.org. https://www.deeplearningbook.org/. [Accessed 27-05-2024].

[3] F. Barbieri, J. Camacho-Collados, L. Neves, and L. Espinosa-Anke. Tweeteval: Unified benchmark and comparative evaluation for tweet classification. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: Findings*, pages 1644–1650. Association for Computational Linguistics, 2020.

[4] J. Bollen, H. Mao, and X. Zeng. Twitter mood predicts the stock market. *Journal of computational science*, 2(1):1–8, 2011.

[5] O. Calzone. An intuitive explanation of lstm. https://medium.com/@ottaviocalzone/an-intuitive-explanation-of-lstm-a035eb6ab42c, 2022.

[6] J. Dancker. A Brief Introduction to Recurrent Neural Networks — towardsdatascience.com. https://towardsdatascience.com/a-brief-introduction-to-recurrent-neural-networks-638f64a61ff4. [Accessed 27-05-2024].

[7] E. F. Fama. Efficient capital markets: A review of theory and empirical work. *The Journal of Finance*, 25(2):383–417, 1970.

[8] D. M. A. O. Han Zhang, Ian Goodfellow. Self-attention generative adversarial networks. https://arxiv.org/abs/1805.08318.

[9] S. Hochreiter and J. Schmidhuber. Long short-term memory. *Neural computation*, 9(8):1735–1780, 1997.

[10] B. Huang, Y. Ou, and K. M. Carley. Aspect level sentiment classification with attention-over-attention neural networks. In *Social, Cultural, and Behavioral Modeling: 11th International Conference, SBP-BRiMS 2018, Washington, DC, USA, July 10-13, 2018, Proceedings 11*, pages 197–206. Springer, 2018.

[11] kaggleStockTweets. Stock Tweets for Sentiment Analysis and Prediction — kaggle.com˙https://www.kaggle.com/datasets/equinxx/stock-tweets-for-sentiment-analysis-and-prediction. [Accessed 18-05-2024].

[12] B. Peng, S. Narayanan, and C. Papadimitriou. On limitations of the transformer architecture, 2024.

[13] M. Rastogi. Tutorial on LSTM: A computational perspective — towardsdatascience.com. https://towardsdatascience.com/tutorial-on-lstm-a-computational-perspective-f3417442c2cd. [Accessed 27-05-2024].

[14] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin. Attention is all you need, 2023.

[15] Y. Xu and S. B. Cohen. Stock movement prediction from tweets and historical prices. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1970–1979, 2018.