CS371: Final Project Report
Multimodal Transformer Forecaster for Next-Day TSLA Price

Melese Medhin_20210727

1. Introduction

Predicting the next-day closing price of highly volatile stocks like Tesla (TSLA) is a difficult yet important task for both investors and risk managers. Traditional models such as ARIMA or LSTM often struggle with capturing long-term dependencies and adapting to sudden market shifts. With the growing impact of social media on stock movements, especially for companies like TSLA, combining public sentiment with historical price data has become a promising direction. This project is motivated by the idea that a Transformer-based model, known for its ability to handle long sequences and dependencies, can better capture the complex interactions between price movements and public sentiment.

2. Problem definition & challenges

The goal of this project is to predict the next-day closing price of TSLA stock using a 10-day look-back window. Each input day includes six features: Open, High, Low, Close, Volume, and a Twitter-based sentiment score. The task is formulated as a regression problem, where the model outputs a single predicted closing price for the next day.

This problem is not trivial for several reasons:

Long-Range Dependencies: Stock prices can be influenced by patterns and signals that span multiple days. Simple models fail to capture these long-term relationships effectively.

Multimodal Alignment: Integrating numerical price data with daily-aggregated, often noisy sentiment data from Twitter introduces challenges in aligning and balancing the two modalities.

Scale Imbalance: The Volume feature can have values much larger than the price or sentiment scores, making proper normalization essential to prevent biased learning.

3. Related Works

Mittal and Goel (2012) used Twitter sentiment and boosted regression trees to predict next-day stock prices, showing that social media data can improve accuracy over historical baselines. Nguyen et al. (2015) compared lexicon-based and machine learning sentiment methods for S&P 500 forecasting and found that supervised classifiers performed better when aligned with intraday prices.

Recent works have applied large language models (LLMs). A 2025 study used FinBERT with a RAG pipeline on financial news and found that negative sentiment had a stronger short-term impact, though the model had a low R^2. Another study (2024) on the Chinese market used BERT for comment sentiment and fed the scores into a Bi-LSTM, which improved RMSE for large-cap stocks.

These studies show that sentiment helps, especially with advanced NLP, but also highlight challenges like noise, alignment issues, and low predictive power in some cases. This project builds on that by combining price and sentiment data using a Transformer.

4. Methodology

This project frames the task as learning a mapping from a 10-day window of six features (OHLCV + Twitter sentiment) to predict the next-day TSLA closing price. I use a Transformer encoder to model long-range temporal dependencies and combine both price and sentiment data in a single pipeline.

Significance and Novelty

Instead of relying on LSTM or other recurrent models, the architecture uses a Transformer encoder with sinusoidal positional encoding. This helps capture long-range temporal patterns without the vanishing gradient problem. To integrate public sentiment, tweets mentioning TSLA were first cleaned and lemmatized using standard NLP steps (tokenization, stopword removal, POS tagging, and lemmatization). Then, sentiment polarity scores were computed using the TextBlob library. These scores were aggregated daily and added as the sixth input feature, enabling the model to attend to both market trends and social sentiment.

This end-to-end design helps address three key challenges: (1) long-range dependencies via self-attention, (2) scale imbalance by applying normalization across features, and (3) multimodal alignment by fusing sentiment and price at the input level.

Model Architecture

The model follows this pipeline:

- Project each 6D input vector to a 64D embedding
- Add sinusoidal positional encoding
- Pass through 2 layers of TransformerEncoder (4 heads, FF dim=128)
- Apply global average pooling
- Use a final dense layer to predict the next-day closing price

Training Details

- Look-back window: 10 days
- Optimizer: Adam, LR = 1e-4
- Batch size: 32, Epochs: 100
- Dropout: 0.1
- All seeds are fixed for reproducibility (NumPy, PyTorch)

5. Experiments and Results

Dataset, Computer Resources and Experimental Design

This project uses two main datasets: TSLA OHLCV data and Twitter sentiment data. The price data was collected from Yahoo Finance using the FinanceDataReader library and covers the period from January 1, 2020, to December 31, 2020. It includes six features for each of the 252 trading days: Open, High, Low, Close, Volume, and Adjusted Close.

The model was trained on the standard version of Google Colab.

The combined dataset was split chronologically, with 80% used for training and the remaining 20% for testing, in order to preserve the time-series structure. All input features, including sentiment, were scaled using Min-Max normalization. A 10-day sliding window approach was applied to generate input sequences, resulting in input tensors with shape $N \times 10 \times 6$ where each sample contains 10 consecutive days of six features. The model was trained to predict the next-day closing price based on these input windows.

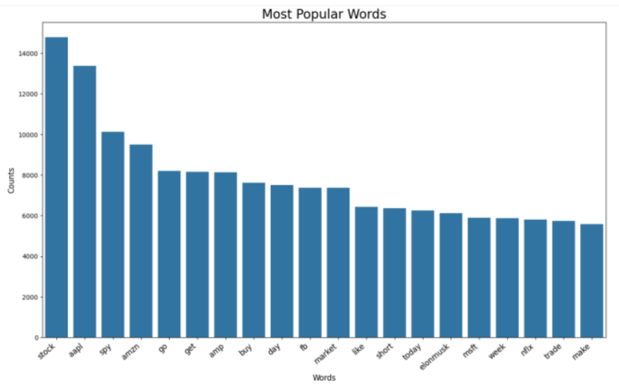

Figure 1 : shows the most popular words in the Twitter data

The bar chart in Figure 1 shows that the most frequent words in the Tesla-related Twitter dataset are finance-related, with "stock" being the most common. Popular stock tickers like "aapl", "spy", and "amzn" also appear often, along with action words such as "go", "get", and "amp". Terms like "market", "trade", and "make" further confirm the dataset's focus on financial discussions. This supports the use of the dataset for sentiment-based stock price prediction.

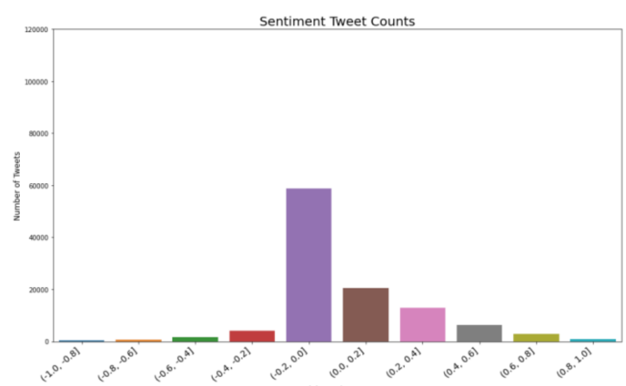

Figure 2: sentiment score distribution of Twitter data

In the above figure, the sentiment score distribution of the tweets is centered around neutral to slightly positive values, with a peak around the 0.0 to +0.2 range. This suggests that most tweets reflect mild optimism about the market. The distribution is fairly symmetrical, covering the full range from -1.0 to +1.0, ensuring diverse sentiment signals. This balance and coverage support reliable daily aggregation and help avoid model bias.

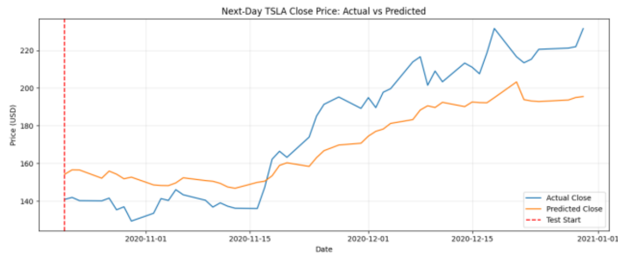

Figure 3: Transformer based approach using sentiment scores: Actual vs predicted price

Method	Loss
LSTM(without sentiment)	232.8
LSTM(with sentiment)	362.6
Transformer(with sentiment)	352.8

Table 1: Performance of different methods

In figure 4, the predicted line generally follows the upward trend of the actual prices, demonstrating that the model is able to capture the overall direction of the market. However, the predicted values tend to be smoother and slightly lag behind the actual prices, especially during periods of rapid price increases. This is typical for regression models on volatile financial data, as extreme movements are challenging to capture precisely. The visualization confirms that the Transformer model, while not perfect, provides reasonable forecasts for TSLA's next-day closing price.

This table compares the test loss (Mean Squared Error, MSE) of three different models:

- **LSTM (without sentiment):** A baseline LSTM model trained only on price features (OHLCV), achieving the lowest test loss (232.8).
- **LSTM (with sentiment):** An LSTM model that incorporates daily Twitter sentiment scores as an additional feature, but with a higher test loss (362.6), indicating that sentiment did not improve the LSTM's performance in this case.
- **Transformer (with sentiment):** The proposed Transformer-based model using both price and sentiment features. Its test loss (352.8) is slightly better than the LSTM with sentiment, but does not outperform the price-only LSTM.

The results indicate that, for this dataset and feature set, the addition of sentiment features did not lead to improved performance for LSTM, and the Transformer model performed comparably to the LSTM with sentiment. This suggests that while advanced architectures like Transformers can model complex dependencies, the quality and informativeness of sentiment features are critical for realizing performance gains.

6. Future Direction

Future work could explore using more advanced sentiment models like FinBERT or RoBERTa to improve the quality of sentiment features and reduce noise. Incorporating additional sources such as financial news, Reddit, or StockTwits may provide a richer and more reliable sentiment signal. Expanding the dataset to include multiple stocks or indices could help the model learn broader patterns and generalize better. Finally, integrating macroeconomic indicators or alternative data sources like Google Trends could further enhance the model's robustness and predictive performance.

References

1. Nguyen, Thien Hai, Kiyoaki Shirai, and Julien Velcin. “Sentiment Analysis on Social Media for Stock Movement Prediction.” *Expert Systems with Applications*, vol. 42, no. 24, 2015, pp. 9603–9611. Elsevier, https://doi.org/10.1016/j.eswa.2015.07.052.

2. Nguyen, Thien Hai, and Kiyoaki Shirai. “Topic Modeling Based Sentiment Analysis on Social Media for Stock Market Prediction.” *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, Association for Computational Linguistics, 2015, pp. 1354–1364. https://doi.org/10.3115/v1/P15-1131.
3. Rao, Tarun, and Srivastava, Saket. “Twitter Sentiment Analysis: How to Hedge Your Bets in the Stock Markets.” *arXiv preprint arXiv:1212.1107*, 2012. https://arxiv.org/abs/1212.1107.