Stock Market Price Prediction Using Temporal Fusion Transformer and Llama 3.2

Nithish Kumar
Advisor: Prof. Anton Selitsky
Department of Computer Science
Golisano College of Computing and Information Sciences
Rochester Institute of Technology
Rochester, NY 14623
nk6825@rit.edu

***Abstract*—This project proposes a novel market price prediction tool using a Temporal Fusion Transformer (TFT) trained on historical hourly stock data and market news. To incorporate news context, Llama is employed to process articles and generate semantic news vectors. These vectors, mapped to their respective timestamps, enable the model to explain sudden market movements while learning general patterns from the stock data.**

***Index Terms*—Stock Market Prediction, Temporal Fusion Transformer, Llama, Large Language Model (LLM), News Embedding, Time-Series Analysis, Market Sentiment, Long Short Term Memory (LSTM), Bidirectional Encoder Representations from Transformers (BERT), FinBERT, Short-Term Prediction.**

I. Introduction

The stock market plays a vital part in the economy, allowing companies to raise capital and providing investors with opportunities to increase their wealth in return for investing in them. Changes in equity market prices reflect investors' sentiment and expectations for the respective companies. However, this can be risky for any investor due to price fluctuations and market uncertainties. Predicting the stock market has become one of the most sought-after and unsolved long-standing challenges due to its high volatility and random nature. According to the Efficient Market Hypothesis (EMH), stock prices already reflect all available information, making it nearly impossible to consistently outperform the market without taking on additional risk. Despite this, many investors and researchers still pursue stock price prediction as a way to gain an edge in the market. Accurate predictions can serve as a backbone for improved portfolio management and better returns, even as debates around market efficiency continue.

Traditional approaches to predict prices have relied on statistical methods like Autoregressive Integrated Moving Average (ARIMA) models and simple linear regression. However, with advancements of machine learning/deep learning and the abundance of big data, neural network models have gained popularity for capturing complex patterns in stock price movements. The rise of deep learning has enabled more accurate predictions by using sophisticated models like LSTM that can handle sequential data and are suited for capturing temporal patterns in price movements.

However, stock price movements can't be explained just by historical prices and technical indicators. There are several other factors that affect the market, especially sentiment-driven events, since the market is driven by the fear and greed of the people. News related to a company, industry, or the overall economy can have a huge impact on the overall investor sentiment, which in turn influences stock prices. For example, a positive earnings report or the announcement of a major partnership can drive up stock prices, while negative news like corporate scandals or regulatory issues can lead to sharp declines. This complexity has mandated the inclusion of sentiment analysis in stock price prediction models, where Natural Language Processing (NLP) techniques are used to extract valuable insights from news articles, social media, and financial reports.

In this project, we propose a novel stock market price prediction tool using deep learning that integrates technical analysis of hourly stock price data with financial news information, aligning it to each corresponding hour. The primary objective of this tool is to predict hourly stock prices for the following week, helping investors with short-term trading and investment strategies. The core components of the prediction tool are a TFT model, which is designed to capture temporal patterns and handle time series data effectively, and a Llama for sentiment analysis, which processes financial news to help the TFT understand changes that can't be explained by temporal data. The news articles are timestamped and mapped to hourly stock price data, ensuring that the sentiment extracted from the news is aligned with the time period in which it affects the stock price. This alignment between temporal stock data and news sentiment is crucial for improving the accuracy of the predictions.

The TFT is a state-of-the-art deep learning model that excels at multi-variate time series forecasting. It is specifically designed to handle complex temporal relationships, missing data, and other challenges that arise in time series prediction. TFT is able to provide both accurate point forecasts and

interpretability by learning which variables are important at each time step. The TFT leverages self-attention mechanisms, which enable the model to capture complex patterns and dependencies in the data that may be missed by traditional sequential models like LSTM. This self-attention allows the TFT to weigh the importance of different time steps and features more effectively, offering improved accuracy in time series forecasting.

News analysis, on the other hand, is performed using a fine-tuned version of Llama 3.2 called Llama-Instruct. The Llama processes the news articles related to the stocks being analyzed and generates a vector representation of the sentiment for each news item. This vector is then concatenated with the price data, allowing the model to incorporate both the technical data from the stock prices and the sentiment derived from news in its predictions. The integration of sentiment analysis with technical analysis is expected to improve the accuracy of the predictions, especially in cases where external factors cause sudden stock price movements.

In recent years, there has been considerable research on combining machine learning models with sentiment analysis for financial forecasting. Many of these studies have shown promising results, particularly in the short-term prediction of stock prices. For instance, researchers have successfully employed LSTM models along with sentiment scores derived from Twitter data and news articles using FinBERT to predict stock price movements. However, these models often struggle with the interpretability of results and the ability to handle long-term dependencies in the data. Additionally, LSTM models, while powerful, are not inherently designed to provide multi-horizon forecasts. This is where the TFT model stands out, as it not only robust to missing values but also leverages a self-attention mechanism to effectively capture complex dependencies and long-term relationships in time series data. Unlike LSTMs, which are primarily focused on single-step predictions, the TFT excels in multi-horizon forecasting, allowing for simultaneous predictions across multiple time steps.

The TFT demonstrated a significant improvement in prediction accuracy compared to traditional models that rely solely on historical data. The TFT outperformed the baseline model, particularly in early timesteps, where its ability to capture complex patterns and dependencies proved advantageous. When trained on a comprehensive dataset that incorporated both historical data and external factors, such as news articles, the TFT further improved its overall performance. This highlights the effectiveness of integrating external information into the model, as it enhanced predictive accuracy, especially for later timesteps, by providing valuable context that helped the model account for market shifts driven by news events.

Overall, the model delivered strong predictive performance, providing accurate hourly stock price forecasts for the following day. This makes it a valuable tool for investors seeking to capitalize on short-term trading opportunities.

II. Background and Related Work

Financial forecasting has become an invaluable tool for investors, analysts, and institutions, providing critical insights to predict market trends and shape trading strategies. Statistical models have long served as foundational approaches in this domain, each bringing unique strengths to understanding market behavior. The ARIMA model, introduced by Box and Jenkins in the 1970s, represents a structured method for modeling time series data. While ARIMA is well-suited for short-term, linear predictions, it requires stationary data, which limits its applicability in markets exhibiting trends and seasonality [1]. To address financial data's inherent volatility, Engle introduced the Autoregressive Conditional Heteroscedasticity (ARCH) model, which accounts for time-varying variance, making it suitable for periods of heightened market fluctuation. Bollerslev's Generalized Autoregressive Conditional Heteroscedasticity (GARCH) model further extended ARCH by incorporating both past variances and residuals, providing a more robust framework for volatile financial data [2].

However, the linear nature of ARIMA and GARCH limits their effectiveness in capturing the complex, non-linear relationships often present in financial data. Due to these limitations, Artificial Neural Networks (ANNs) gained popularity because of their ability to capture non-linear patterns. Their flexibility allows ANNs to capture the dynamic nature of stock prices more effectively. [3].

Despite their adaptability, ANNs have limitations, notably their susceptibility to overfitting, especially with small or unrepresentative datasets. While ANNs often perform well on training sets, they can struggle with volatile or unseen market data, leading to inconsistent results [3].

Recurrent Neural Networks (RNNs), offer a unique structure designed to handle sequential data by retaining information from previous time steps. However, traditional RNNs face challenges with learning from longer sequences due to the issue of vanishing gradients, where information from earlier time steps fades as it is passed through multiple layers. This limitation makes it difficult for RNNs to capture long-term dependencies, which are crucial in financial forecasting. To address this, LSTM networks were developed as an advanced form of RNNs. LSTMs use specialized gating mechanisms to control the flow of information, allowing them to retain important information over longer periods and effectively learn long-term dependencies in time series data. This capacity has made LSTMs particularly effective in financial forecasting, as they can capture both short- and long-term patterns that influence stock prices. However, their computational demands and need for large datasets pose

challenges, especially for real-time prediction applications [4].

Market sentiment, particularly derived from financial news, often correlates with stock price movements. Positive sentiment is typically associated with price increases, while negative sentiment tends to coincide with declines. With the growing availability of unstructured data, such as financial news and social media content, text mining has become increasingly important in finance. The introduction of Pre-trained Language Models (PLMs) like BERT marked a significant advancement in Natural Language Processing (NLP), allowing models to be pre-trained on large corpora and then fine-tuned for specific tasks. In the financial domain, FinBERT — a specialized variant of BERT was designed to understand the unique language and sentiment of financial text, improving performance on tasks like sentiment analysis. By utilizing multi-task learning and unsupervised transfer learning, FinBERT deepens the understanding of financial texts, providing valuable insights that enhance stock prediction models [5]. Building on this, Kim et al. compared two models: LSTM-ONLY, a basic LSTM, and LSTM-NYT, where sentiment extracted by FinBERT from financial news articles is incorporated into the LSTM model for S&P 500 Index forecasting. Their study revealed that the LSTM-NYT model, which integrates sentiment analysis, significantly outperformed the standalone LSTM model [6].

One of the key limitations of many deep learning models is their lack of interpretability, often referred to as the "black-box" nature of these models. The TFT addresses this issue by incorporating interpretability into a powerful forecasting framework. Unlike traditional RNNs, TFT employs an attention-based mechanism that dynamically emphasizes the most relevant features during training, which is crucial in finance, where understanding the impact of different variables such as economic indicators, trends, and market sentiment can inform decision-making. TFT combines recurrent layers to capture short-term patterns with attention layers that capture long-term dependencies, providing a balance between local insights and broader temporal trends. Additionally, TFT includes feature selection, ensuring that only relevant data is considered, which refines the model's predictions by reducing noise. Studies show that TFT outperforms traditional deep learning models in multi-horizon forecasting tasks and offers interpretability by highlighting the most important input variables [7].

Hajek et al. took this a step further by integrating TFT with FinBERT. The FinBERT-TFT model incorporates both objective financial metrics and sentiment from financial news, offering a more comprehensive approach to stock price prediction. FinBERT extracts sentiment scores from textual data, which TFT then incorporates through its attention mechanism, allowing the model to weigh the importance of sentiment in relation to other variables dynamically. This integration enhances predictive accuracy and provides interpretability, making it possible to understand how news sentiment influences forecasts. By combining the strengths of both TFT's interpretability and FinBERT's domain-specific sentiment analysis, this model addresses limitations of previous approaches and offers a robust, transparent tool for financial forecasting [8].

Despite its advancements, FinBERT is not without its limitations. One of the main drawbacks is its relatively small size and fewer parameters compared to larger language models like Llama, which can limit its performance on more complex tasks. Additionally, FinBERT has a restriction on the length of input text it can process, typically performing best with shorter texts such as headlines or brief summaries, rather than full articles or long-form financial reports. This constraint means that FinBERT may struggle to capture the deeper context and nuances found in longer texts, reducing its overall utility in cases where a more comprehensive understanding of financial documents is required.

III. Methodology and Implementation

A. Data Collection

The data collection process involved gathering 5 years of hourly historical stock price data for TSLA and corresponding financial news articles. The stock price data was sourced from Databento, while news articles were retrieved from platforms such as Yahoo Finance, Marketflux.io, and the FNSPID dataset.

B. Llama Inference

The Llama 3.2 3B (3 billion parameters) model was employed for processing financial news articles and generating semantic embeddings. The "instruct" variant of Llama, fine-tuned for following instructions and understanding context, was specifically utilized to ensure accurate and relevant feature extraction from the news articles.

To handle lengthy financial news articles, Llama accommodates a maximum input length of 3072 tokens (approx. 2400 words). Articles exceeding this length were truncated in a fashion that ensures that the most relevant portions of the text were preserved.

The resulting embeddings, referred to as news vectors, captured the contextual, semantic, and sentiment-based nuances of the articles. These vectors were aligned with their respective hourly timestamps and integrated with the TFT model to enhance prediction accuracy by incorporating the influence of market news.

C. Data Preprocessing

The data preprocessing phase ensured that both stock price data and news articles were properly prepared for integration into the TFT model. Key steps included:

1) Handling Timestamps: Missing timestamps were filled to create a continuous time series. All timestamps were converted from UTC to Eastern Time to align with standard U.S. market hours.

2) Price Data Imputation: Missing stock prices, particularly during off-hours and weekends not included in the dataset, were filled with the last known closing prices to maintain data continuity.

3) News Vector Dimensionality Reduction: The 3072-dimensional output news vectors generated by Llama were passed through an encoder to reduce their dimensionality to 128. This feature reduction step minimized computational overhead while retaining the essential semantic information from the news data.

4) Normalization: Stock prices were normalized to ensure numerical stability and compatibility with the TFT model's input requirements.

These preprocessing steps ensured that the dataset was clean, well-aligned, and optimized for efficient model training and accurate prediction performance.

D. Technology and Hardware

The implementation was developed using PyTorch, a versatile deep learning framework widely recognized for its flexibility and ease of use in defining and training neural networks. PyTorch's seamless integration with CUDA allowed the model to leverage GPU acceleration, significantly speeding up the training and inference processes.

To streamline the training pipeline, PyTorch Lightning was employed. This framework provided a structured approach to managing training loops, validation processes, and logging, reducing boilerplate code and improving workflow efficiency. PyTorch Lightning also facilitated the integration of advanced features such as mixed precision training, which was essential for optimizing GPU performance.

The PyTorch Forecasting module was utilized to implement the TFT efficiently. This specialized library offered pre-configured components designed specifically for time-series forecasting tasks, reducing development time while ensuring high model performance.

For the inference of Llama, the Transformers library was used. This library provided pre-built utilities and APIs for efficiently loading and running large language models like Llama.

The model was trained and evaluated on a hardware setup comprising an NVIDIA GPU with 8 GB VRAM and an Intel i7 12th-gen CPU.

E. Optimizations

To enhance the efficiency of the system and optimize its resource usage, several key optimizations were implemented during both training and inference:

1) Data Type Optimization: All data was converted to `float32` or smaller data types, reducing memory consumption while preserving numerical precision.

2) Mixed Precision Training: The TFT was trained using `bfloat16` mixed precision on GPUs. This approach significantly reduced computational overhead, speeding up the training process without sacrificing model accuracy.

3) Quantization: An `int4` quantized version of Llama was utilized for processing financial news articles. This reduced the model's memory footprint and inference time, enabling rapid semantic embedding generation from large volumes of news data.

These optimizations enabled the model to run efficiently despite hardware constraints, ensuring that training and inference processes could be completed within practical timeframes without exceeding memory limits.

F. Model Training

The model is trained to predict the stock price for the next day, encompassing 24 timesteps, based on the past week of data (168 timesteps). During training, the model learns to capture the underlying patterns in the stock prices, while also considering the impact of financial news through the news vectors. The training process focuses on optimizing the model's ability to make accurate short-term predictions.

1) Loss Function: The model employs a Time-Weighted Mean Squared Percentage Error (TW-MSE) as the loss function. This choice prioritizes prediction accuracy for more immediate timesteps, which is essential for short-term forecasting. In particular, early predictions carry more weight, ensuring that the model focuses on minimizing the error in these crucial timesteps while still accounting for longer-term predictions.

2) Optimizing Hyperparameters: Several hyperparameters were carefully tuned during the training process to optimize model performance. These hyperparameters play a key role in controlling the learning dynamics, model capacity, and generalization ability. The following hyperparameters were optimized:

- **Learning rate:** Controls the step size during optimization, affecting how quickly the model converges.
- **Hidden size:** Determines the number of units in the model's hidden layers, influencing its ability to capture complex patterns in the data.

- **Hidden continuous size:** Defines the size of the continuous state space in the model, which impacts the representation of temporal dependencies.
- **News vector size:** Specifies the dimensionality of the vector representation of news articles, affecting the granularity of the information passed to the model.
- **Attention head size:** The number of attention heads in the multi-head attention mechanism, determining how the model attends to different parts of the input sequence.
- **Gradient clip size:** Limits the size of gradients during backpropagation to prevent gradient explosion, ensuring stable training.

IV. Results and Evaluation

A. Evaluation Metric

To assess the performance of the proposed model, we utilized the MAPE as the primary evaluation metric. MAPE is widely used for regression tasks and provides a clear indication of prediction accuracy by comparing the absolute percentage error between the predicted and actual stock prices. Lower MAPE values indicate better predictive performance, with the model's ability to make accurate short-term predictions being a key focus.

$$MAPE = \frac{1}{n}\sum_{i=1}^{n}\left|\frac{\hat{y}_i - y_i}{y_i}\right| \times 100$$

Where:

$\hat{y}_i$ = Predicted value for the i^{th} data point

y_i = Actual value for the i^{th} data point

n = number of observations

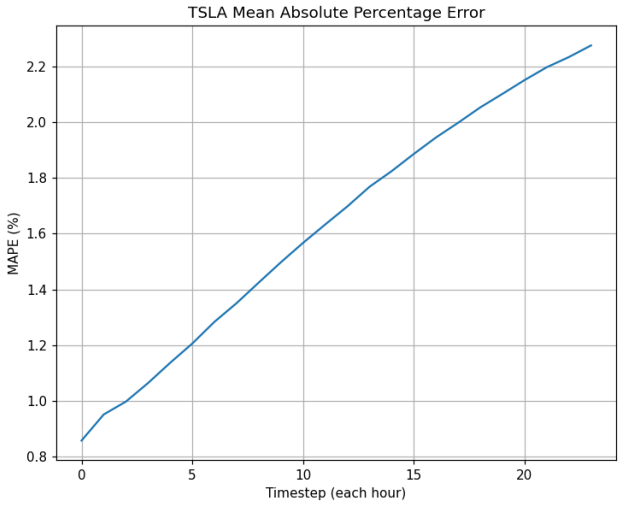

Fig. 1: MAPE for each timestep

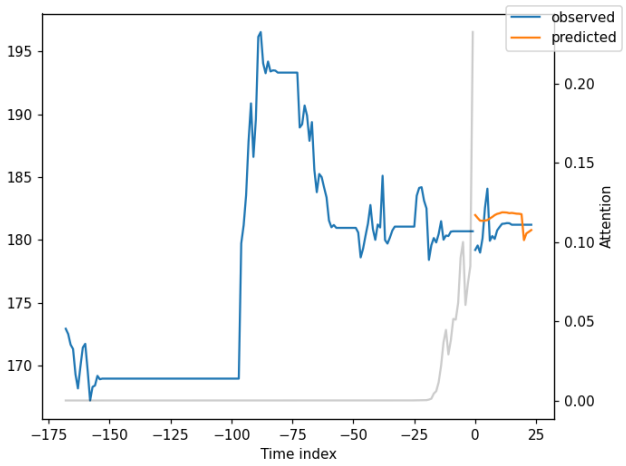

Fig. 2: Example prediction

B. Results

The TFT model demonstrated strong performance in predicting stock prices. On the first timestep, the model achieved a MAPE of 0.8%, significantly outperforming the baseline LSTM model, which recorded a MAPE of 3.4%. This shows that the TFT can capture important short-term trends and market fluctuations more accurately. Furthermore, the model trained on both historical stock price data and financial news articles improved the overall prediction accuracy, reducing the MAPE across all timesteps from 2.4% (price-only model) to 1.6%. These results underscore the value of incorporating external sentiment sources, such as news articles, into stock price forecasting models.

C. Model Comparison

The performance of the TFT was compared against the popular LSTM model for time-series forecasting. As discussed earlier, the TFT outperformed the LSTM model in terms of prediction accuracy, which can be attributed to its attention mechanisms that enable it to better capture long-term dependencies and adapt to complex patterns in the data. However, models relying solely on price data struggled to account for sudden market shifts driven by news events, leading to higher MAPE values.

Building on the comparison with the LSTM, we also evaluated the TFT with Llama against a model that combines LSTM and FinBERT (a BERT variant fine-tuned for financial sentiment analysis). While the LSTM + FinBERT model successfully integrated sentiment analysis, its performance still lagged behind the TFT. This can be attributed to the TFT's unique ability to simultaneously process both time-series data and sentiment signals using its attention mechanisms, allowing it to more effectively capture the complex relationships between market trends and news events.

V. Conclusion

In this study, the TFT demonstrated superior performance in predicting stock prices compared to traditional models like

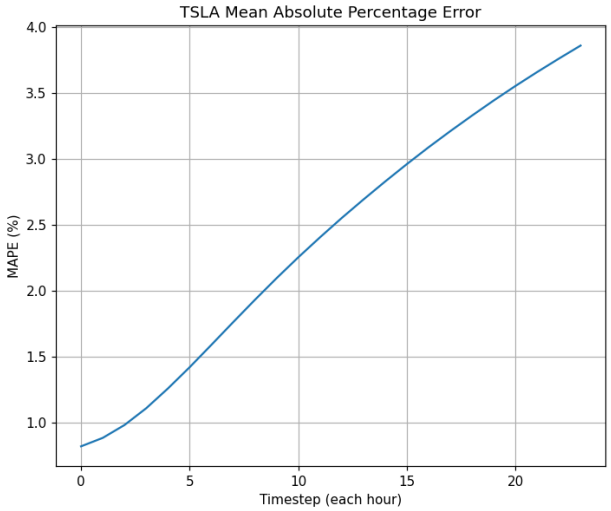

Fig. 3: MAPE for each timestep of price-only model

Model	MAPE Next Hour	MAPE Overall
LSTM	3.4%	5.7%
LSTM + FinBERT	2.6%	4.4%
TFT	0.88%	2.4%
TFT + Llama	0.86%	1.6%

TABLE I: MAPE comparison of different models

LSTM networks. The integration of sentiment analysis, derived from financial news articles, contributed to a significant improvement in the model's overall MAPE, highlighting its effectiveness in capturing both market trends and the impact of external events. This suggests that incorporating sentiment analysis into forecasting models can enhance their predictive power, making them valuable tools for short-term market predictions.

VI. Future Work

Several avenues for future work are identified to further enhance and expand the capabilities of the proposed model:

1) Expand to a Broader Range of Stocks: The model could be extended to predict stock prices for a broader set of companies, such as those in the S&P 500 index, to assess its performance across various industries and market conditions.

2) Integrate with Portfolio Optimization: The predictive capabilities of the model can be integrated into a portfolio optimization tool. This would allow investors to optimize their stock holdings and improve returns based on the predicted movements of individual stocks.

3) Develop a Web Application: A user-friendly web application could be developed to visualize predictions using interactive graphs. This would allow users to explore predictions across various time scales, making the model more accessible and practical for real-world trading scenarios.

4) Incorporate Diverse Sentiment Sources: The model could be enhanced by integrating additional sentiment sources, such as social media platforms and financial forums, to capture a broader range of market sentiment. This could further improve the accuracy of predictions by accounting for real-time public sentiment.

References

[1] A. A. Ariyo, A. O. Adewumi and C. K. Ayo, "Stock Price Prediction Using the ARIMA Model," 2014 UKSim-AMSS 16th International Conference on Computer Modelling and Simulation, Cambridge, UK, 2014, pp. 106-112, doi: 10.1109/UKSim.2014.67.

[2] Atoi, Ngozi V. (2014) "Testing Volatility in Nigeria Stock Market using GARCH Models," CBN Journal of Applied Statistics (JAS): Vol. 5: No. 2, Article 4. Available at: https://dc.cbn.gov.ng/jas/vol5/iss2/4

[3] Dattatray P. Gandhmal, K. Kumar, Systematic analysis and review of stock market prediction techniques, Computer Science Review, Volume 34, 2019, 100190, ISSN 1574-0137, https://doi.org/10.1016/j.cosrev.2019.08.001. (https://www.sciencedirect.com/science/article/pii/S157401371930084X)

[4] M. A. Istiake Sunny, M. M. S. Maswood and A. G. Alharbi, "Deep Learning-Based Stock Price Prediction Using LSTM and Bi-Directional LSTM Model," 2020 2nd Novel Intelligent and Leading Emerging Sciences Conference (NILES), Giza, Egypt, 2020, pp. 87-92, doi: 10.1109/NILES50944.2020.9257950.

[5] Liu, Z., Huang, D., Huang, K., Li, Z., & Zhao, J. (2020). FinBERT: A pre-trained financial language representation model for financial text mining. In C. Bessiere (Ed.), Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence, IJCAI-20 (pp. 4513–4519). International Joint Conferences on Artificial Intelligence Organization. https://doi.org/10.24963/ijcai.2020/622.

[6] Kim J, Kim H-S, Choi S-Y. Forecasting the S&P 500 Index Using Mathematical-Based Sentiment Analysis and Deep Learning Models: A FinBERT Transformer Model and LSTM. Axioms. 2023; 12(9):835. https://doi.org/10.3390/axioms1209083.

[7] Temporal Fusion Transformers for interpretable multi-horizon time series forecasting, International Journal of Forecasting, Volume 37, Issue 4, 2021, Pages 1748-1764, ISSN 0169-2070, https://doi.org/10.1016/j.ijforecast.2021.03.012.

[8] Hajek, P., Novotny, J. (2024). Beyond Sentiment in Stock Price Prediction: Integrating News Sentiment and Investor Attention with Temporal Fusion Transformer. In: Maglogiannis, I., Iliadis, L., Macintyre, J., Avlonitis, M., Papaleonidas, A. (eds) Artificial Intelligence Applications and Innovations. AIAI 2024. IFIP Advances in Information and Communication Technology, vol 713. Springer, Cham. https://doi.org/10.1007/978-3-031-63219-8_3.

[9] OpenAI. (2024). ChatGPT (October 8 Version) [Large language model]. *https://chat.openai.com*. (Note: Personal communications generally do not appear in the reference list).