Enhancing Economic Time Series Prediction with News Text Data and Numerical Data: A Transformer-Based Approach

Shangyang Mou[1] Wenting Zhang[1] Takuji Kinkyo[1] Shigeyuki Hamori[1]
Jinhui Chen[2] Tetsuya Takiguchi[1] Yasuo Ariki[1]
[1] Kobe University [2] Wakayama University
205x610x@stu.kobe-u.ac.jp

Abstract

Financial time series such as stock prices can be hard to predict as it is difficult to model short-term and long-term temporal dependencies between data points. In addition, they are influenced by underlying fundamentals that are conveyed through various news feeds. Existing methods for predicting financial time series typically focus on either numerical data or textual information. When considering textual data, the prevailing approach is to utilize sentiment information extracted from news sources as features. This paper introduces a novel methodology that directly integrates financial news content with corresponding time series data, to enhance the accuracy of financial time series forecasting.

1 Introduction

Financial time series forecasting is a challenging signal processing problem, largely because the underlying financial system that generates such data is enormously complex, resulting in data that can exhibit both non-linear and non-stationary characteristics [1]. However, financial data prediction plays a pivotal role in various aspects of the financial domain, from individual investment decisions to macroeconomic policy making [2].

With the growing availability of big data, there has been an increase in research into financial date analysis. For example, from automated trading systems [3, 4] to stock price prediction [5, 6], the availability and analysis of large financial datasets have enhanced the accuracy and speed of market predictions and investment strategies, significantly impacting the field of financial technology.

In financial time series forecasting, unlike data from weather, traffic, or electricity [7, 8], financial time series are influenced by a variety of complex factors. These include historical data as well as macroeconomic policies, global events, and related economic indices [9, 10]. Existing approaches in financial series prediction have utilized either textual or numerical data, with some models considering both types concurrently [11, 12]. The most methods employing financial text have predominantly relied on sentiment analysis, using sentiment scores as features to assist model predictions [13, 14]. These models typically focus on predicting the direction in which the data will fall or rise, rather than directly predicting the data series. However, inaccuracies in sentiment analysis, especially mispredicted sentiment labels, can notably affect the quality and accuracy of financial predictions, leading to significant deviations in forecasted outcomes.

Recently, transformer models have become increasingly popular for time series forecasting tasks, showing impressive outcomes [8, 7, 15, 16, 17]. However, [18] claims that transformers may not be as effective as anticipated, with results showing that linear models outperform the more complex transformer-based models. Despite this, considering the transformers' robust language processing capabilities [19], we explore whether it can effectively handle textual features combined with numerical data for financial data forecasting.

2 Proposed Method

This paper proposes a methodology that directly uses news text as a feature combined with numerical data for forecasting. The approach, implemented on transformer-based models, has yielded promising results. The effectiveness of this method illustrates the potential of integrating complex language models with traditional numerical time series analysis, offering a more comprehensive understand-

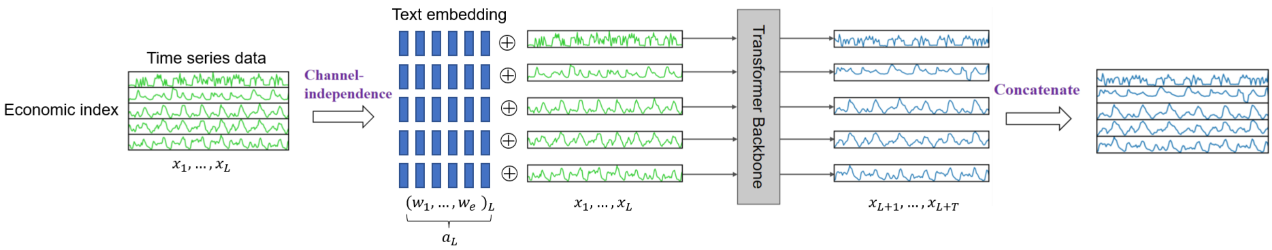

Figure 1 FRAMEWORK ARCHITECTURE.

ing of the financial data. For the time series forecasting tasks, we consider the following problem: given a collection of multivariate time series samples with look-back window $L : (x_1, ..., x_L)$ where each x_t at time step t is a vector of dimension M, we would like to forecast T future values $(x_{L+1}, ..., x_{L+T})$.

Forward Process. We define the i-th univariate series of length L, starting at time index 1, as $x^{(i)}_{1:L} = (x^{(i)}_1, ..., x^{(i)}_L)$, where $i = 1, ..., M$. The input numerical data $(x_1, ..., x_L)$ is divided into M univariate series $x^{(i)} \in \mathbb{R}^{1\times L}$, and each is processed independently through the transformer backbone according to PatchTST [16] model's channel-independence setting. Then the corresponding news text data denoted as $a_t = (w_1, ..., w_e)_t$ w_t is concatenated to each univariate time series. Since we only use the latest day's news text embedding w_L, for each time series $(x_1, ..., x_L)$, the final input to the transformer becomes a concatenation of the numerical series and the textual embedding: $z^{(i)} = (x^{(i)}_{1:L}, a_L)$, with our objective being to forecast T future values $(x_{L+1}, ..., x_{L+T})$.

2.1 Model Architecture

In the study by Nie et al. [16], a novel transformer architecture, named PatchTST, was introduced, innovatively incorporating two pivotal concepts - channel independence and patching - specifically designed for enhancing time series forecasting.

Channel independence. Channel independence is a property of the PatchTST model that allows different channels of the input to be processed independently. In traditional transformers, the same set of attention weights is used for all channels, which limits the model's ability to capture fine-grained information in each channel. In contrast, the PatchTST model applies attention weights separately to each channel, allowing it to better capture the unique features and patterns in each channel.

Patching. Patching is a technique to alleviate the computational burden of self-attention. Rather than attending to every position in the sequence, the input sequence is partitioned into smaller sub-sequences known as patches. Self-attention is then computed between the patches. This approach enables the model to handle longer sequences while avoiding memory constraints and facilitating quicker inference. Additionally, patching enables the capture of localized semantic information that can not be available when using individual point-wise input tokens.

By dividing the input into patches and processing each channel independently, the model can efficiently capture complex patterns and relationships across the entire sequence. Inspired by this, we extend the input to combine with news text, as shown in Figure 1, for each channel we concatenate the correspond news text embedding to the time series.

Transformer Encoder. PatchTST uses the vanilla transformer encoder that maps the observed signals to the latent representations. Initially, the numerical data and textual feature embeddings are concatenated, forming an enriched input sequence. This combined sequence is then mapped to the transformer's latent space of dimension D via a trainable linear projection $W_p \in \mathbb{R}^{D\times P}$, where P is the dimension after concatenation. Additionally, a learnable additive position encoding $W_{pos} \in \mathbb{R}^{D\times N}$ is applied to maintain the temporal order of the concatenated patches: $z^{(i)}_d = W_p z^{(i)}_p + W_{pos}$, where $z^{(i)}_d \in \mathbb{R}^{D\times N}$ denote the combined input fed into the transformer encoder. In the multi-head attention mechanism, each head $h = 1, ..., H$ transforms the input into query matrices $Q^{(i)}_h = (z^{(i)}_d)^T W^Q_h$, key matrices $K^{(i)}_h = (z^{(i)}_d)^T W^K_h$, and value matrices $V^{(i)}_h = (z^{(i)}_d)^T W^V_h$, where $W^Q_h, W^K_h \in \mathbb{R}^{D\times d_k}$ and $W^V_h \in \mathbb{R}^{D\times D}$. The scaled dot-product attention mechanism then generates the attention output $O^{(i)}_h \in \mathbb{R}^{D\times N}$ as:

$$\begin{aligned}(O^{(i)}_h)^T &= Attention(Q^{(i)}_h, K^{(i)}_h, V^{(i)}_h) \\ &= Softmax\left(\frac{Q^{(i)}_h (K^{(i)}_h)^T}{\sqrt{d_k}}\right)V^{(i)}_h\end{aligned} \tag{1}$$

The multi-head attention block incorporates BatchNorm layers and a feed-forward network with residual connections. Finally, a flatten layer followed by a linear head is used after the encoder to obtain the prediction result $\hat{x}^{(i)}_{L+1:L+T} = (\hat{x}^{(i)}_{L+1}, \ldots, \hat{x}^{(i)}_{L+T}) \in \mathbb{R}^{1\times T}$.

FinBert. To effectively process news text, we employ the FinBERT model [20], a domain-specific pre-trained BERT architecture specialized for financial contexts. FinBERT is adept at extracting features from financial texts, enabling it to capture the complex sentiment and thematic content inherent in financial news. In our approach, FinBERT is utilized to generate feature embeddings which serve as a rich, contextual representation of the news data.

The integration of textual features and quantitative time series data significantly enhances the model's ability to forecast time series with greater accuracy, leveraging the rich contextual information present in the news text to provide a more comprehensive understanding of the factors influencing the time series trends.

2.2 Model Input

For our study, we constructed a dataset focused on the U.S. economy by utilizing a collection of news headlines scraped from the official websites of CNBC, the Guardian, and Reuters[1]. These headlines provided an overview of the U.S. economy and stock market trends over the past one to two years. To complement this textual data, we aligned it with corresponding period economic time series data, including indices such as the SP 500, MSCI US, NYSE American Composite, Nasdaq Composite, Dow Jones Industrial Average, and the US Dollar Index.

Given the significantly higher dimensionality of textual data compared to numerical data, and the fact that news is not available every day, we only select latest day's news. On days without news, we inserted a 'no news' marker. In our transformer-based approach, we added a masking layer to handle these 'no news' instances by masking them, so the model do not pay attention to them. To align the news data with the corresponding index in dimension and retain as much textual information as possible, we concatenate the latest day's news with a historical window of time series data, as the latest updates tend to have the greatest impact. For instance, to forecast the time series for the upcoming 12 days, we utilise the previous 96-day time series and the latest day's news, reflecting the immediacy and relevance of the latest news in influencing economic trends.

1) https://www.kaggle.com/datasets/notlucasp/financial-news-headlines

Considering that the U.S. stock market typically has 256 trading days per year, coupled with the scarcity of financial news datasets, we constructed a dataset containing 611 samples. The dataset consists of economic data for the period 2018-03-01 to 2020-07-02 and corresponding news texts. The choice of this time period and the resulting size of the dataset is dictated by the amount of relevant news content available for analysis. The dataset thus provides a comprehensive view of the economic landscape over a given period, combining quantitative market data with qualitative news narratives to provide an integrated approach to financial analysis.

3 Experiments

In this section we describe the details of the training process and the experiment setting.

Loss Function. Consistent with PatchTST [16] we use the Mean Squared Error (MSE) loss to measure the discrepancy between the prediction and the ground truth. The loss in each channel is gathered and averaged over M time series to get the overall objective loss:

$$L = E_x \frac{1}{M} \sum_{i=1}^{M} ||\hat{x}^{(i)}_{L+1:L+T} - x^{(i)}_{L+1:L+T}||_2^2 \tag{2}$$

Model Parameters. In Table 1, we compare our dataset US Economy with other popular datasets dedicated to multivariate time series forecasting tasks. Notably, our dataset appears to be relatively small, primarily due to the scarcity of news text data. This peculiarity prompted specific adjustments in our model's parameters. Original PatchTST model configuration consists of 3 encoder layers with the head number $H = 16$ and the latent space dimension $D = 128$. However, in our adapted model that incorporates news text data, we found it necessary to modify the architecture to better capture the complexity introduced by the additional textual information. To this end, we increased the number of encoder layers to 12, providing a deeper and more nuanced processing capability for the combined numerical and textual data. Additionally, with our dataset variables, we optimized the head number to 4, ensuring efficient processing while capturing the intricate patterns in the combined numerical and textual inputs.

Results Analysis. Based on our experiments, integrating news text as a feature into the corresponding economic data time series has demonstrated significant im-

Datasets	Weather	Traffic	Electricity	ETTm1	US Economy(ours)
Features	21	862	321	7	6
Timesteps	52696	17544	26304	69680	611

Table 1 Dataset Comparison.

provements in multivariate time series forecasting. Specifically, as shown in Table 2 when only economic data was used, MSE = 0.5779. However, upon incorporating the corresponding news text as an additional feature, there was a notable reduction in the MSE to 0.4773. This substantial decrease underscores the efficacy of including news text in enhancing the accuracy of our forecasting model. The results clearly indicate that the contextual insights provided by the news text play a crucial role in predicting economic trends more accurately, thereby validating our approach of combining qualitative textual data with quantitative time series data.

	L	N	MSE
Num	96	12	0.5779
Num+News	96	12	0.4773

Table 2 Multivariate time series forecasting results with look-back window L and number of input tokens N are predicted.

In our comparative analysis of model performance with and without news content shown in Figure 2, we observed a marked contrast in the optimization trajectories. Using solely economic data, the model rapidly converged to its minimum MSE of 0.5779 within just 7 epochs, after which it began to overfit, as indicated by the subsequent increase in validation loss. This swift convergence suggests that the model finds it relatively straightforward to learn from numerical data, which, although indicative of economic trends, lacks the complexity inherent in textual information.

Conversely, when we incorporated news text into the forecasting model, the optimal MSE of 0.4773 was achieved only after a considerably longer training period of 70 epochs. This prolonged learning curve highlights the complexity that textual data introduces to the model, necessitating a more nuanced and detailed learning process. Notably, the model trained with text data did not exhibit signs of overfitting within the observed epochs, suggesting that it was able to continue learning and integrating insights from the textual content effectively.

The extended training required for the text-enhanced model is indicative of its process to assimilate and interpret the multifaceted influences captured in textual data, which are often non-linear and abstract compared to numerical

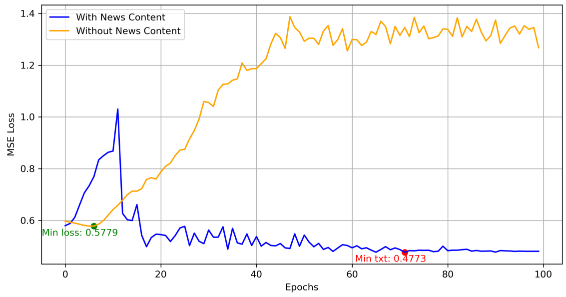

Figure 2 Comparison of MSE Loss with & without News Content.

data. The improved MSE with textual data integration demonstrates the model's enhanced capability to discern and synthesize the latent factors affecting economic tendencies, resulting in wiser and more precise forecasts.

These observations underscore the comprehensive nature of forecasts that incorporate qualitative insights from text data, affirming the value of combining diverse data types for a deeper understanding of economic movements. This approach aligns with the intricate realities of financial markets, where qualitative factors often steer the quantitative indicators, suggesting a promising direction for advancing financial time series forecasting methodologies.

4 Conclusions

In conclusion, our method integrates economic news text with economic data for multivariate time series forecasting. This approach adds influential news content to the prediction process, enhancing traditional numerical data analysis. Our experiments demonstrated that features extracted from news text can improve economic sequence forecasting.

However, a key limitation is the limited availability of financial news that high-related to specific economic indicators. The sheer volume of market news makes it challenging to identify news that directly affects certain economic metrics. Moreover, aligning them temporally with economic data is a non-trivial task. Therefore, our validation is conducted on a relatively small dataset.
Moving forward, we plan to extend the scope of our dataset to further validate our findings. A larger dataset will allow a more thorough evaluation of our model and the broader impact of text-enhanced time series forecasting. This direction of research holds the potential to significantly improve the precision of economic predictions by incorporating a richer spectrum of qualitative information.

References

[1] Adam Atkins, Mahesan Niranjan, and Enrico Gerding. Financial news predicts stock market volatility better than close price. **The Journal of Finance and Data Science**, Vol. 4, No. 2, pp. 120–137, 2018.

[2] Daniel Broby. The use of predictive analytics in finance. **The Journal of Finance and Data Science**, Vol. 8, pp. 145–161, 2022.

[3] Yang Liu, Qi Liu, Hongke Zhao, Zhen Pan, and Chuanren Liu. Adaptive quantitative trading: An imitative deep reinforcement learning approach. In **Proceedings of the AAAI conference on artificial intelligence**, Vol. 34, pp. 2128–2135, 2020.

[4] Boming Huang, Yuxiang Huan, Li Da Xu, Lirong Zheng, and Zhuo Zou. Automated trading systems statistical and machine learning methods and hardware implementation: a survey. **Enterprise Information Systems**, Vol. 13, No. 1, pp. 132–144, 2019.

[5] Sreelekshmy Selvin, R Vinayakumar, EA Gopalakrishnan, Vijay Krishna Menon, and KP Soman. Stock price prediction using lstm, rnn and cnn-sliding window model. In **2017 international conference on advances in computing, communications and informatics (icacci)**, pp. 1643–1647. IEEE, 2017.

[6] Mehar Vijh, Deeksha Chandola, Vinay Anand Tikkiwal, and Arun Kumar. Stock closing price prediction using machine learning techniques. **Procedia computer science**, Vol. 167, pp. 599–606, 2020.

[7] Haoyi Zhou, Shanghang Zhang, Jieqi Peng, Shuai Zhang, Jianxin Li, Hui Xiong, and Wancai Zhang. Informer: Beyond efficient transformer for long sequence time-series forecasting. In **Proceedings of the AAAI conference on artificial intelligence**, Vol. 35, pp. 11106–11115, 2021.

[8] Haixu Wu, Jiehui Xu, Jianmin Wang, and Mingsheng Long. Autoformer: Decomposition transformers with auto-correlation for long-term series forecasting. **Advances in Neural Information Processing Systems**, Vol. 34, pp. 22419–22430, 2021.

[9] Rakesh Kumar Verma and Rohit Bansal. Impact of macroeconomic variables on the performance of stock exchange: a systematic review. **International Journal of Emerging Markets**, Vol. 16, No. 7, pp. 1291–1329, 2021.

[10] Torben G Anderson, Tim Bollerslev, Francis X Diebold, and Clara Vega. Micro effects of macro announcements: Real-time price discovery in foreign exchange. **American economic review**, Vol. 93, No. 1, pp. 38–62, 2003.

[11] Mark Cecchini, Haldun Aytug, Gary J Koehler, and Praveen Pathak. Making words work: Using financial text as a predictor of financial events. **Decision support systems**, Vol. 50, No. 1, pp. 164–175, 2010.

[12] Renáta Myšková and Petr Hájek. Mining risk-related sentiment in corporate annual reports and its effect on financial performance. **Technological and Economic Development of Economy**, Vol. 26, No. 6, pp. 1422–1443, 2020.

[13] Ryo Akita, Akira Yoshihara, Takashi Matsubara, and Kuniaki Uehara. Deep learning for stock prediction using numerical and textual information. In **2016 IEEE/ACIS 15th International Conference on Computer and Information Science (ICIS)**, pp. 1–6. IEEE, 2016.

[14] Bledar Fazlija and Pedro Harder. Using financial news sentiment for stock price direction prediction. **Mathematics**, Vol. 10, No. 13, p. 2156, 2022.

[15] Tian Zhou, Ziqing Ma, Qingsong Wen, Xue Wang, Liang Sun, and Rong Jin. Fedformer: Frequency enhanced decomposed transformer for long-term series forecasting. In **International Conference on Machine Learning**, pp. 27268–27286. PMLR, 2022.

[16] Yuqi Nie, Nam H Nguyen, Phanwadee Sinthong, and Jayant Kalagnanam. A time series is worth 64 words: Long-term forecasting with transformers. In **The Eleventh International Conference on Learning Representations**, 2022.

[17] Yong Liu, Tengge Hu, Haoran Zhang, Haixu Wu, Shiyu Wang, Lintao Ma, and Mingsheng Long. itransformer: Inverted transformers are effective for time series forecasting. **arXiv preprint arXiv:2310.06625**, 2023.

[18] Ailing Zeng, Muxi Chen, Lei Zhang, and Qiang Xu. Are transformers effective for time series forecasting? In **Proceedings of the AAAI conference on artificial intelligence**, Vol. 37, pp. 11121–11128, 2023.

[19] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need. **Advances in neural information processing systems**, Vol. 30, , 2017.

[20] Zhuang Liu, Degen Huang, Kaiyu Huang, Zhuang Li, and Jun Zhao. Finbert: A pre-trained financial language representation model for financial text mining. In **Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence**, IJCAI'20, 2021.