DOI: 10.54254/2755-2721/96/20241351

Reviews on Transformer-based Models for Financial Time Series Forecasting

Heyi Lin

School of Science, The Hong Kong University of Science and Technology, Hong Kong, China

hlinaw@connect.ust.hk

Abstract. The emergence of competitive deep learning models has increasing attached attention from both the academia and industry. Thus, as one of the fields that tend to chase the state-of-art and fashion technological trend, some previous work in financial time series forecasting has turned to deep learning models, including transformer-based models. While an examination work questioning the effectiveness of transformers for general time series forecasting (TSF) in 2022, researchers are keen to work on the creative design of transformer-based neural network architectures and related improvements. On the other hand, since the success of ChatGPT in 2023 as the milestone of transformers and Large Language Models (LLMs), an alternative method is put forward that implements domain-specific LLM in financial text to obtain sentiment information or generate trading signals, which does not solve the forecasting problem but provide support in decision making in investment. This review will scan through the history of the above models and methodologies in financial time series forecasting.

Keywords: Transformer, Time Series Forecasting, Financial Time Series.

1. Introduction

The emergence of competitive deep learning models has increasingly attracted attention from both academia and industry, and finally from the public right after the launch of ChatGPT (GPT-3.5 and GPT-4) [1]. Before ChatGPT, transformers with self-attention mechanisms made significant progress on natural language translation, and following on distinct tasks including Natural Language Processing (NLP), speech recognition and computer vision [2-5]. Transformers therefore can be regarded as the most successful sequence modeling architecture.

On the other hand, time series forecasting (TSF) is a well-known area with a myriad of practical applications in different subjects [3]. The applications specialized in finance are also crucial and attractive, as there are many temporal market data in a financial market with an intrinsic nature as time series straightforward, for example, stock prices and interest rates. Though, seems to resemble to time series from other sources, financial time series usually have a significantly higher degree of volatility or uncertainty, which indicates that there are more obstacles for achieving prediction accuracy than the general TSF tasks [3, 6].

Therefore, as one of the fields that tend to chase the state-of-art and fashion technological trend, finance has been drown to deep learning, with another purpose of solving or alleviating difficulties in financial TSF tasks. According to the proliferation of deep learning models, some previous work in financial time series forecasting have turned to these models and architectures, including Transformers and later Large Language Models (LLMs) [3, 4, 7, 8].

In this review, the author attempts to explore the historical evolution of transformer-based models and methodologies. The review will commence with the introduction of the vanilla Transformer architecture, which is followed by the important examination work presented in 2022. And the post-examination exploration and progress will be described, with an alternative method through LLMs will be mentioned. The review is expected to provide a general insight and assist in stimulating possible innovations in the future.

2. Background and Base Model Architecture

In 2017, the research team of Vaswani, etc. put forward a remarkable neural network architecture which solely contains the multi-head self-attention mechanism without any recurrent deep learning structure [2]. The details can be referred to the source work. In brief, there are two stacks in the vanilla transformers: encoder and decoder. There are 6 identical layers in the encoder, and each layer contains a multi-head attention sublayer and a feed-forward network sublayer, with layer-normalization [9]. The decoder has almost the same structure, but each of its layers has an additional masked attention sublayer over the output of the encoder.

The self-attention mechanism is the core of any Transformer. Specifically, the typical scaled dot-product attention layer in the vanilla consists of the inputs of query, key and value representations, with the corresponding packed matrices Q , K of dimension d_k , and V of dimension d_v . Since the variance of the dot product will be d_k if all elements of Q and K will be mutually independent with mean 0 and variance 1, the scaling factor should be $d_{\mathrm{k}}^{-1/2}$ [2, 5]. Therefore, the output of the attention layer is

$$Attention(Q,K,V) = Softmax[\frac{QK^T}{{d_k}^{1/2}}]V \tag{1}$$

Also, the positional encoding as an additional process right after embedding is added to preserve part of the positional information for each element in an input array. The typical form of positional encoding takes sine and cosine functions of different frequencies [2, 5].

With flexible attention weights from learnable query for key, the self-attention mechanism allows the model to consider all positions of the input sequence with flexible focus on a particular fraction of input positions when generating each output. Because time series data usually has a long-distance dependence, the self-attention mechanism should also be efficient in capturing this long-distance dependence, as the similar performance in NLP tasks.

3. Examinations and Doubts

After the emergence and success of the vanilla Transformer architecture, the blueprint for the development of transformer-based models was initially promising with abundant focus. For instance, as early variates of the original transformer architectures particularly for time series forecasting, ConvTrans and LogTrans were put forward in the same work in 2019 [10]. ConvTrans proposes the convolutional self-attention mechanism, which employ a filter on the attention weights to prevent future information leakage and keep the prediction solely on the mechanisms of the transformers. LogTrans refers to LogSparse Transformer, which contains the LogSparse mechanisms to reduce the total cost of computational memory usage for attention scores from $O(L^2)$ to $O(L \log L)$ for every input sequence with length L . Other transformer-based innovations on architectures as Informer and FEDformer demonstrates the efforts of the researchers in the field of Long-term time series forecasting.

Unfortunately, in 2022, an examination work questioned the effectiveness of transformers for Long-term Time Series Forecasting (LTSF) tasks, with the state-of-art transformers during the specific timestamp in 2022 compared to "embarrassing simple" baseline models, including *LTSF-Linear* with two variates *DLinear* / *NLinear*, and *Repeat* [11]. *LTSF-Linear* contains no other layers but only one temporal simple linear layer, and *Repeat* is a plain model only repeating the last value in the look-back window. *DLinear* adds the Decomposition scheme, which is used in FEDformer into the linear layer, and *NLinear* subtracts the input by the last value of the sequence to solve the problem of distribution shift [11, 12].

The examination work shows that the *LTSF-Linear* model astonishingly outperforms transformer-based models by 20% ～ 50% for multivariate general time series forecasting and other simple baseline models significantly surpass the transformer models. These results prove that transformers-based models designed at that time is not appropriate on working the LTSF task, as the research group explains that the self-attention mechanism is permutation-invariant and will ignore some of the temporal / positional information of time series in nature.

4. Post-examination Innovations

However, there are still new research focusing on both the general and financial TSF task with transformers and similar architectures. For instance, one of the research teams in 2022 provide a revised layer design called channel-independent patch time series Transformers (PatchTST), including dividing the original time series into subseries-level patches and design each channel that contain only one univariate time series with the same embedding and Transformer weights [13]. Another example later in 2023 is iTransformer, which apply attention and feed-forward network layers, which is the same as the vanilla Transformer, on the inverted dimension, to improve the unsatisfactory performance for large lookback window and alleviate the problem of redundant attention maps capturing the noise as information [2, 14].

In details, iTransformer employs a unique method of mapping the entire sequence of each variable to a variable token. Traditional Transformer models typically embed multiple variables with the same timestamp into indistinguishable channels, which may result in the erasure of multivariate correlations. ITransformer independently embeds the entire time series of each variable into a variable token, which can expand the local receptive field and better utilize attention mechanisms for multivariate association. For example, in a time series prediction task containing multiple sensor data, the time series of each sensor is independently embedded as a variable token, which can better capture the correlation between different sensors [14].

The iTransformer's use of self-attention mechanism to capture multivariate correlations is innovative. The self-attention mechanism allows the model to consider all positions of the input sequence, not just the earlier parts, when generating each output. In iTransformer, self-attention mechanism is applied to embedded variable tokens, which obtain queries, keys, and values through linear projection, calculate the pre-Softmax score, and reveal the correlations between variables. This method provides a more natural and interpretative mechanism for multivariate sequence prediction. By capturing multivariate correlation between variables, this method also has potential to contribute to specific financial trading strategies, such as pairs trading [15].

On the other hand, the sudden success of ChatGPT put NLP tasks and Large Language Models (LLMs) on the radar of researchers and practitioners in finance. In 2019, based on general-purpose Bidirectional Encoder Representations from Transformers (BERT), a domain-specific model FinBERT for NLP tasks in finance was introduced [16, 17]. And later in 2023 after the success of ChatGPT, a proprietary model called BloombergGPT was put forward as the first LLM specialized for the financial domain [8]. Right after BloombergGPT, FinGPT was raised as an open-source financial LLM, which can analyze sentiment in finance corpus and generate trading signals [7]. The trading signal generation processes can be viewed as an alternative or implicit solution to the financial TSF tasks, which do not provide concrete predictions on the time series but directly support decision making in practice.

5. Conclusion

In this work, the author briefly introduces the history of the transformer-based models and methodologies applied to financial time series forecasting. Though with substantial obstacles in the architectural innovations and the fact that the Transformer-based methodologies do not surpass the other deep learning architectures significantly, their developments are still ongoing and hopefully provide different perspective on the financial industry in the future.

References

[1] Liu, Y., Han, T., Ma, S., Zhang, J., Yang, Y., Tian, J., ... & Ge, B. (2023). Summary of ChatGPT-related research and perspective towards the future of large language models. *Meta-Radiology*, 100017. https://doi.org/10.1016/j.metrad.2023.100017

[2] Vaswani, A. (2017). Attention is all you need. *Advances in Neural Information Processing Systems*. https://doi.org/10.48550/arXiv.1706.03762

[3] Lim, B., & Zohren, S. (2021). Time-series forecasting with deep learning: a survey. *Philosophical Transactions of the Royal Society A*, *379*(2194), 20200209. https://doi.org/10.1098/rsta.2020.0209

[4] Wen, Q., Zhou, T., Zhang, C., Chen, W., Ma, Z., Yan, J., & Sun, L. (2022). Transformers in time series: A survey. *arXiv preprint arXiv:2202.07125*. https://doi.org/10.48550/arXiv.2202.07125

[5] Bishop, C. M., & Bishop, H. (2023). *Deep learning: Foundations and concepts*. Springer Nature. 357-74. https://doi.org/10.1007/978-3-031-45468-4

[6] Zou, J., Zhao, Q., Jiao, Y., Cao, H., Liu, Y., Yan, Q., ... & Shi, J. Q. (2022). Stock market prediction via deep learning techniques: A survey. *arXiv preprint arXiv:2212.12717*. https://doi.org/10.48550/arXiv.2212.12717

[7] Yang, H., Liu, X. Y., & Wang, C. D. (2023). Fingpt: Open-source financial large language models. *arXiv preprint arXiv:2306.06031*. https://doi.org/10.48550/arXiv.2306.06031

[8] Wu, S., Irsoy, O., Lu, S., Dabravolski, V., Dredze, M., Gehrmann, S., ... & Mann, G. (2023). Bloomberggpt: A large language model for finance. *arXiv preprint arXiv:2303.17564*. https://doi.org/10.48550/arXiv.2303.17564

[9] Ba, J. L. (2016). Layer normalization. *arXiv preprint arXiv:1607.06450*. https://doi.org/10.48550/arXiv.1607.06450

[10] Li, S., Jin, X., Xuan, Y., Zhou, X., Chen, W., Wang, Y. X., & Yan, X. (2019). Enhancing the locality and breaking the memory bottleneck of transformer on time series forecasting. *Advances in neural information processing systems*, *32*. https://doi.org/10.48550/arXiv.1907.00235

[11] Zeng, A., Chen, M., Zhang, L., & Xu, Q. (2023, June). Are transformers effective for time series forecasting? In *Proceedings of the AAAI conference on artificial intelligence* (Vol. 37, No. 9, pp. 11121-11128). https://doi.org/10.48550/arXiv.2205.13504

[12] Zhou, H., Zhang, S., Peng, J., Zhang, S., Li, J., Xiong, H., & Zhang, W. (2021, May). Informer: Beyond efficient transformer for long sequence time-series forecasting. In *Proceedings of the AAAI conference on artificial intelligence* (Vol. 35, No. 12, pp. 11106-11115). https://doi.org/10.48550/arXiv.2012.07436

[13] Nie, Y., Nguyen, N. H., Sinthong, P., & Kalagnanam, J. (2022). A time series is worth 64 words: Long-term forecasting with transformers. *arXiv preprint arXiv:2211.14730*. https://doi.org/10.48550/arXiv.2211.14730

[14] Liu, Y., Hu, T., Zhang, H., Wu, H., Wang, S., Ma, L., & Long, M. (2023). itransformer: Inverted transformers are effective for time series forecasting. *arXiv preprint arXiv:2310.06625*. https://doi.org/10.48550/arXiv.2310.06625

[15] Han, C., He, Z., & Toh, A. J. W. (2023). Pairs trading via unsupervised learning. *European Journal of Operational Research*, *307*(2), 929-947. https://doi.org/10.1016/j.ejor.2022.09.041

[16] Devlin, J. (2018). Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*. https://doi.org/10.48550/arXiv.1810.04805

[17] Yang, Y., Uy, M. C. S., & Huang, A. (2020). Finbert: A pretrained language model for financial communications. *arXiv preprint arXiv:2006.08097*. https://doi.org/10.48550/arXiv.2006.08097