Using transformer in stock trend prediction

Zhichen Liu

School of Computer Science and Engineering, Southern University of Science and Technology, Shenzhen, Guangdong, 518000, China

12011125@mail.sustech.edu.cn

Abstract. Large transformer model had achieved good results in many tasks, such as computer vision (CV) and natural language processing (NLP). However, in financial domains, the application of large deep learning models is rarely observed. Stock Trend Prediction (STP) is a task that using Limit Order Books (LOBs) to predict the future stock price trend by the sequence of historical limit order information, the trend can be Current works are mostly based on the structure of Convolutional Neural Network (CNN) + Recurrent Neural Networks (RNN). This structure is hard to parallel and cannot make full use of GPU resources. It is also difficult to increase the dimension to fit more complex data and performs poor when time sequence is long. Recently, some works proposed that CNN + Transformer model can also work is solving this task. This paper verifies that Transformer can be directly used into STP task and gain a good result, and proposes a novel Transformer-based model, Transformer-LOB, to enhance the basic transformer model performance. This model uses attention mechanisms to extract temporal information rather than using RNN, which utilizes the GPU effectively. Since all the feature extractions are based on transformer modules, the model is scalable and easy to parallel. Transformer-LOB is tested on FI-2010 LOB dataset and SZ-2015 LOB dataset, and outputs ideal results on both datasets.

Keywords: stock trend prediction, limit order book, transformer, neural networks, deep learning

1. Introduction

Stock Trend Prediction (STP) is a highly popular problem in financial area, as it directly impacts investors' returns in the stock market. Due to the high volatility, randomness, and the limited information contained in the trading prices, predicting stock trends solely based on the stock's trading price is a difficult and challenging task [1-3]. Therefore, using Limit Order Book (LOB) to forecast stock trends is a viable approach.

LOB is widely used in stock markets. Over half of the stock markets use LOB to record detail information of each trade [4]. LOB is used to store the records of limit order data. A limit order is a request to transact in a market with the price not exceeding the limited threshold given by the limit order, such as the highest price to buy the stock, or the lowest price to sell the stock. A LOB contains all pending limit orders based on a periodical timestamp of a given stock. Basing on the limit order data, more features are included into the STP task, making it much easier to predict the stock trend. However, it is worth noting that LOB data is still a non-stationary data that contains much randomness. Some operations, such order cancellation, auction and dark pools will lead to variation of the data [5]. As a result, using LOB to predict stock trends remains a challenging task.

Existing works indicate that LOB data is predictable [6,7]. Using machine learning methods to predict stock trends is a feasible way. It is widely recognized that stock prices often exhibit highly complex and non-linear behavior, suggesting the presence of numerous high-dimensional features [8]. Machine learning algorithms are well-suited for analyzing this type of data [9]. Alec N. Kercheval and Yuan Zhang proposed a method using Support Vector Machine to predict the hand-crafted features extracted from LOB data [10]. Nikolaos Passalis et al. proposed a new method that using Bag-of-Features model and neural network to predict the stock trends [11]. These works need to do feature extraction and prediction respectively. Avraam Tsantekidis et al. firstly used a Convolutional Neural Network (CNN) to classify LOB data in STP, after 7 months, they proposed another work using a Long Short-term Memory (LSTM) to predict the stock trends [12,13]. These works introduced deep learning methods into STP tasks and gained considerable improvements. Zihao Zhang et al. firstly combined CNN structure for feature extraction with RNN structure for time series prediction, and proposed a new model DeepLOB [14]. This work significantly enhanced the accuracy of the STP task to over 80%. Later works such as DeepFolio based on this CNN+RNN structure and improved the accuracy [15]. Peng Yang et al. proposed a CNN + Transformer model named One-dimensional Convolution Embedding Transformer (OCET), which introduced Transformer model into STP task [16]. In FI-2010 dataset, OCET reached the state-of-the-art (SOTA). However, while facing larger and more complex data, OCET cannot effectively increase its dimension to fit such dataset due to its CNN + Transformer structure.

Transformer is a fast, parallelable, and expandable model, it had an outstanding performance in Natural Language Processing (NLP) tasks [17]. Based on Transformer, large language models such as BERT and GPT are proposed and achieved the best performance on text classification tasks and text generation tasks separately [18-21]. Large model such as Vision Transformer (ViT) used Transformer model in Computer Vision (CV) and performed well on CV tasks such as image recognition [22].

Inspired by Transformer and the following works, this paper attempts to incorporate Transformer into the STP task and proposes a Transformer-based stock prediction model, Transformer-LOB. Transformer-LOB using a learnable time embedding layer to embed temporal information into the input data and using Transformer to extract the features inside LOB data.

The experiment is performed on two datasets. FI-2010 is an open-source public dataset from Nasdaq Nordic stock market [23]. Another dataset is a non-public dataset from China A-share market, contains data for 180 stocks in 2015, which is referred to as CN-A-share-2015 in this paper. Transformer-LOB reach the same accuracy level as OCET on FI-2010 and achieved an ideal result with 83.6% accuracy on CN-A-share-2015.

2. Data, normalization and labeling

2.1. Limit order books

This paper followed the definition of article Limit Order Books [24]. A LOB contains two different types of orders, ask orders, and bid orders. An ask order is an order requests to sell a share over the given price, a bid order is an order requests to buy a share under the given price. At time t, the ask order set is defined by $P_a(t)$, $P_b(t)$ is the same. $V_a(t)$ is used to define the set of volumes of ask orders at the given timestamp t, $V_b(t)$ is the same. There are multiple levels of data contains in the order set. The 1st level of ask order is the lowest price in $P_a(t)$, defined as $p_a^1(t)$, where the volume of ask orders that located in this price is defined as $v_a^1\ (t)$. The 1st level of bid order is the highest price in $P_b(t)$, defined as $p_b^1(t)$, and the volume, $v_b^1\ (t)$. Followed by the definition, $P_a(t)$ contains $p_a^1(t)$, $p_a^2(t)$,..., $p_a^n(t)$, where n is a given number, denoting the level of the LOB. LOB data is changing over time. Figure 1 shows the structure of a LOB data at timestamp t and t + 1. The level 1 and level 2 ask orders are cleared when time go from t to t + 1.

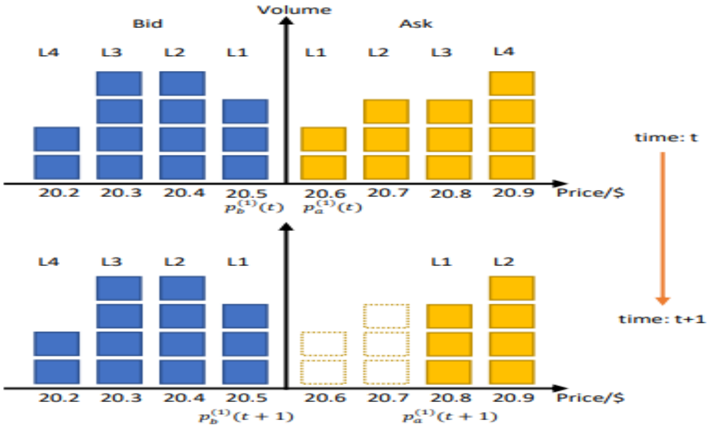

Figure 1. Basic structure of LOB data, where the level 1 and 2 ask orders change when timestamp comes from t to t + 1 [14].

2.2. *Input data structure*

A limit order at given timestamp t contains $P_a(t)$, $V_a(t)$, $P_b(t)$, $V_b(t)$. A n-level limit order at timestamp t can be denoted as $x_t = [p_a^i(t), v_a^i(t), p_b^i(t), v_b^i(t)]_{i=1}^n$. An input data is a matrix that contains the past k timestamps, defined as $X = [x1, x2, ..., xi, ..., xl]$, which is a $l \times 4n$ matrix [14]. Figure 2 shows the input data structure.

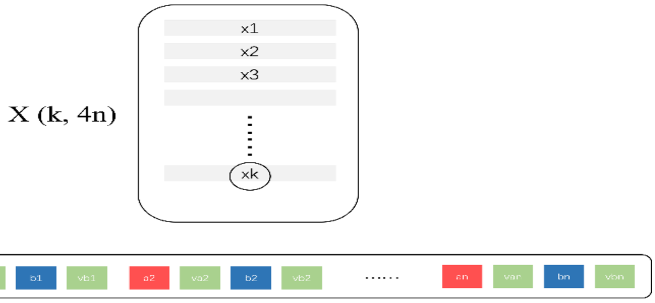

Figure 2. Structure of input data.

2.3. *Labeling*

Labeling is done after data normalization. The origin LOB data will be labeled at each timestamp based on the mid-price, where the mid-price is defined by the formula.

$$p_{mid}(t) = \frac{p_a^1(t) + p_b^1(t)}{2} \tag{1}$$

It is hard to directly know the transaction price at a given moment from the LOB data, because a LOB usually does not contain the information about transaction price. However, since transaction price will fall between the 1-level ask price and 1-level bid price, it is feasible to approximate the current transaction price by using the average of these two prices.

Two ways of labeling are proposed by works of Ntakaris and Tsantekidis [12,23]. Both works need to compute the mean of mid-price in each size of time window k. m−(t) is the mean of previous k timestamps, while m+(t) is the mean of next k timestamps.

$$m_-(t) = \frac{1}{k}\sum_{i=0}^{k} p_{mid}(t-i) \tag{2}$$

$$m_{+}(t) = \frac{1}{k}\sum_{i=0}^{k} p_{mid}(t+i) \tag{3}$$

After getting the mean value of mid-price, there are two ways to calculate the percentage change l(t).

$$l(t) = \frac{m_{+}(t) - p_{mid}(t)}{p_{mid}(t)} \tag{4}$$

$$l(t) = \frac{m_{+}(t) - m_{-}(t)}{m_{-}(t)} \tag{5}$$

The Equation 4 is proposed by Adamantios Ntakaris et al., and equation 5 is proposed by Avraam Tsantekidis et al. [12,23]. From the results in work DeepLOB (shown by Figure 3), Equation 5 provides a smoother labeling approach compared to Equation 4 [14]. This indicates that the results from Equation 5 contain less noise. More noise in labels implies greater volatility in the prediction outcomes, which can lead to redundant trading actions and consequently higher transaction costs.

2.4. *Dataset and normalization*

FI-2010.

FI-2010 is a public dataset of high-frequency (0.3s/data) LOB data. It collects LOB data from the Nasdaq Nordic stock market, contains ten days of trading data for five stocks. FI-2010 is a 10-level LOB dataset. It provides three ways for data normalization, Z-score, min-max scaler, and decimal precision approach, this paper uses the version of data normalized by decimal precision approach. FI-2010 uses Equation 4 to label its data.

CN-A-share-2015.

CN-A-share-2015 is a non-public raw LOB dataset of high-frequency (3s/data) collected from China A-share market, including trading data for 180 stocks throughout the entire year of 2015. The whole dataset is too large, as a result, a small dataset selected from the origin data randomly is used as a substitute in experiment, which contains three months of trading data for ten stocks. It is a 5-level LOB dataset. In this work, the data normalization method is min-max scaler, and the labeling method follows Equation 5.

FI-2010 is a highly frequent dataset. Since it is a 10-level LOB dataset, it contains more information about the market, which makes it is easier to predict. However, the dataset is relatively small, which limits the persuasiveness of the experiment results generated by such a scaled-down dataset compared to the extremely large and complex real word data.

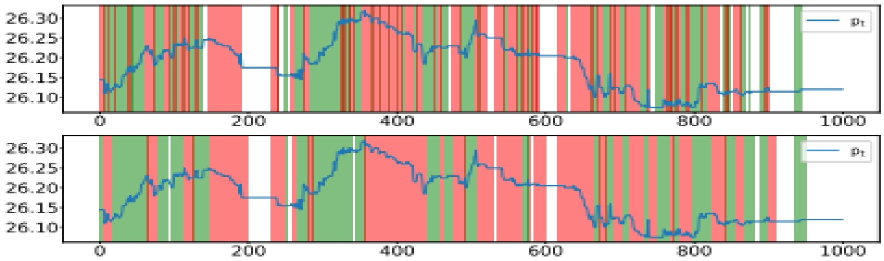

Figure 3. Comparison between two labeling methods in dataset FI-2010. The top one is labeled by Equation 4, the bottom one is labeled by Equation 5. The results show that the bottom one is obviously smoother than the top one [14].

CN-A-share-2015, compared to FI-2010, has a lower frequency, and lesser features per order. It is a 5-level LOB dataset, which make the prediction is harder than FI-2010. Since it is nearly 10 times larger than FI-2010, and contains more stocks than FI-2010, it is better to use a larger model to fit such a dataset. The results experiment on this dataset are much closer to the real-world scenario because it has not undergone any transformations to prevent leakage of real data, ensuring the highest level of authenticity.

3. Model

In the early work, the Transformer model was directly employed and achieved promising outcomes on FI-2010. To further improve the performance and achieve the SOTA level accuracy as OCET, this paper modified the Transformer model and proposed an adapted Transformer model for LOB data, Transformer-LOB.

3.1. Transformer-LOB architecture

Transformer is a model composes of only linear layers and attention mechanisms. It consists of an encoder block and a decoder block [17]. It this work, only Transformer encoder is used for feature extraction. Figure 4 shows the structure of the whole model.

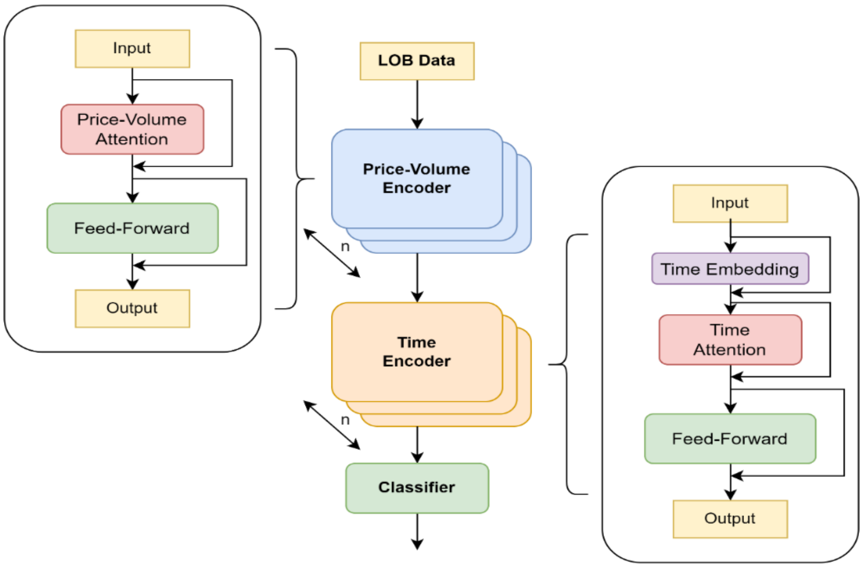

Figure 4. The structure of Transformer-LOB.
(Photo credit: original)

3.2. Price-volume encoder

For a high-frequency LOB data, price-volume features often show more importance than time series information as the frequency is higher. Thus, it is needed to extract the feature between prices and volumes. In Transformer-LOB, the price-volume encoder pays attention to the price-volume features and extracts the features between price and volume. During this step, no embedding or encoding is needed.

3.3. Time encoder

Time features always show it importance in LOB data, however, as the frequency goes lower, the importance of time features is higher, while price-volume features lower. Time encoder pays attention to the time features, the input data first need to pass a time embedding layer and go through a multi-head attention layer and feed-forward layer to extract the time feature.

3.4. Learnable time embedding

OCET proposed a One-dimensional Convolution Embedding (OCE) layer to embedding the input data [16]. Inspired by OCE, Transformer-LOB uses a learnable time embedding layer to embed the input data. Learnable time embedding contains two one-dimensional convolution layers to draw attention to the different scales of short-term adjacent characteristics. The kernel size of the two convolution layers is (3, 1), with stride (1, 1), and (7, 1), with stride (1, 1) respectively, and padding is ’same’. Compared to the original position encoding method in Transformer, this method does not calculate the relation between price and volumes by hands, and all features in price-volume dimension share the same importance. Figure 5 shows the learnable time embedding structure and how it works.

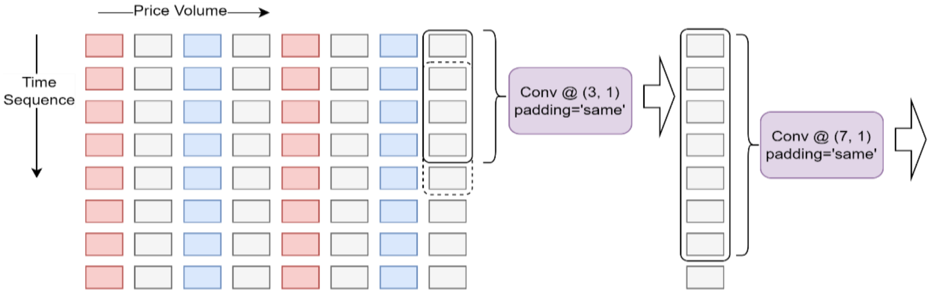

Figure 5. The structure of learnable time embedding and how it learns the time information. (Photo credit: original)

3.5. Multi-head attention and feed-forward

Since different encoders need to pay attention to different dimension of input data features, it is requested that Transformer-LOB need to set the attention head of two different encoders respectively. Since the size of different features’ dimension is different (for example, in FI-2010, the input data has price-volume features 40, and time features 100), the head number must also different.

As a result of different feature size, the Feed-Forward layer also contains different hidden dimension. To be short, the hidden dimension is directly set as 4 times as the input dimension.

3.6. Classifier

The classifier comprises two linear layers. The first layer extract price-volume features and reshape the input tensor into a one-dimensional vector that only contains time series information. The second layer extracts the time features and outputs three classification results.

In STP task, the classifier outputs a three-class classification.

3.7. Activation function and dropout

The activation functions inside transformer blocks are all GELU. Outside the transformer encoder layer (and inside the learnable time embedding layer), the other activation functions are LeakyReLU. Dropout modules are added into each Feed Forward module and Attention module. Besides, the learnable time embedding layer and the last classifier also contains a dropout.

4. Experiment and results

The experiment is done in a cloud server with a single NVIDIA 4090 24GB GPU. All the models are built by PyTorch, and share the same optimizer ADAM, the same learning rate 1e-4, the same batch size 512, and the same loss function Cross-Entropy loss. Tests are done within 50 epochs, each model will be tested 5 times, and the final performance is the mean value of 5 testing results. The experiment is done over FI-2010 and CN-A-share-2015. In FI-2010, the value of input data shape is (100, 40), where the sequence length L is 100, and the level N is 10. In CN-A-share-2015, the value of input data shape is (200, 20), where the sequence length L is 200, and the level N is 5.

Table 1 shows the Models and parameters on FI-2010 Experiment, and Table 2 shows Models and parameters on CN-A-share-2015. The two experiments focus on accuracy, precision, recall, and F1-score. The dataset will have different prediction horizon with k = 10, k = 50, and k = 100.

Table 1. Models and parameters on FI-2010 Experiment.

Model	Parameters
DeepLOB	n=40
DeepFolio	n=40
OCET	n=40, l=100, n_layer=2, dropout=0.1
Transformer-6	n=40, l=100, n_layer=6, head=4, dropout=0.15
Transformer-LOB-4	n=40, l=100, n_layer=4, encoder_head=4, decoder_head=2, dropout=0.15
Transformer-LOB-12	n=40, l=100, n_layer=12, encoder_head=4, decoder_head=2, dropout=0.15

Table 2. Models and parameters on CN-A-share-2015.

Model	Parameters
DeepLOB	n=20
DeepFolio	n=20
OCET	n=20, l=200, n_layer=2, dropout=0.1
Transformer-6	n=20, l=200, n_layer=6, head=4, dropout=0.15
Transformer-LOB-4	n=20, l=200, n_layer=4, encoder_head=4, decoder_head=2, dropout=0.15
Transformer-LOB-12	n=20, l=200, n_layer=12, encoder_head=4, decoder_head=2, dropout=0.15

4.1. Experiments on FI-2010

In experiments on FI-2010, the first 7 days LOB data is the training data, and the last 3 days LOB data is the testing data. Since FI-2010 dataset’s label is not very balance, the original paper suggests focusing more on the F1-score rather than accuracy [23]. The simple original Transformer is directly applied to the STP task; however, its performance cannot reach the level as the two CNN+RNN model. After adjustment, Transformer-LOB gain a slight improvement on the outcomes than the outcomes of previous SOTA OCET. Transformer based models show great advantage in accuracy and f1-score within a large k, while when the k becomes smaller, OCET performs even worth than CNN+RNN model. With the model layer becomes deeper, Transformer-LOB outperforms little than the shallow model. And the Experiment Result on FI-2010 is shown in the Table 3.

Table 3. Experiment Result on FI-2010.

Model	Accuracy %	Precision %	Recall %	F1 %
		k=10		
DeepLOB	77.62	77.35	77.62	77.32
DeepFolio	77.51	77.16	77.51	77.05
OCET	76.57	76.15	76.57	76.24
Transformer-6	67.35	67.44	67.35	67.43
Transformer-LOB-4	78.13	77.79	78.13	77.81
Transformer-LOB-12	**78.14**	77.8	78.14	**77.85**
		k=50		
DeepLOB	78.8	78.84	78.8	78.7
DeepFolio	80.16	80.32	80.16	80.18

Table 3. (continued).

OCET	79.92	79.85	79.92	79.79
Transformer-6	68.76	69.1	68.76	68.93
Transformer-LOB-4	81.21	81.19	81.21	81.19
Transformer-LOB-12	**81.25**	81.19	81.25	**81.2**
		k=100		
DeepLOB	79.79	79.8	79.79	79.78
DeepFolio	78.29	78.45	78.29	78.3
OCET	79.79	79.8	79.79	79.78
Transformer-6	70.11	70.32	70.11	70.25
Transformer-LOB-4	83.15	83.19	83.15	83.12
Transformer-LOB-12	**83.61**	83.6	83.61	**83.58**

4.2. Experiments on CN-A-share-2015

Since the price-volume features are less than FI-2010, CN-A-share-2015 needs to extend its input sequence length to 200 to include more features, which means it contains more time features than price-volume features. Another difficulty is the frequency between each LOB data, making the connection harder to learn. Whether the time feature extractor is strong enough determines the model can have a good performance or not. However, it is still difficult to gain the same performance than FI-2010.

Table. 4. Experiment Result on CN-A-share-2015.

Model	Accuracy %	Precision %	Recall %	F1-Score %
		k=10		
DeepLOB	74.86	74.04	74.886	73.62
DeepFolio	75.04	74.18	75.04	74.1
OCET	71.33	70.33	71.33	70.42
Transformer-6	66.23	66.57	66.23	66.4
Transformer-LOB-4	83.26	83.46	83.26	82.83
Transformer-LOB-12	**83.43**	83.68	83.43	**82.99**
		k=50		
DeepLOB	75.61	75.23	75.61	75.21
DeepFolio	78.89	78.37	78.59	78.33
OCET	87.66	87.95	87.66	87.68
Transformer-6	70.16	70.47	70.16	70.31
Transformer-LOB-4	87.8	87.92	87.8	87.74
Transformer-LOB-12	**87.87**	88.07	87.87	**87.8**
		k=100		
DeepLOB	77.01	77.08	77.01	77.03
DeepFolio	79.08	79.49	79.08	79.18
OCET	91.4	91.67	91.4	91.53
Transformer-6	73.22	73.48	73.22	73.35
Transformer-LOB-4	**92.11**	92.14	92.11	**92.12**
Transformer-LOB-12	92.04	92.03	92.04	92.03

When the labelling k increase (indicating that the time relationship that the model needs to learn from the labeled data increase), CNN+RNN models performances also increase, since the time features inside the data increase. When the sequence length expands, OCET does not outperform the CNN+RNN models, since the time feature extract module inside OCET is too weak to handle it when the importance of time feature increase. Deep Transformer-LOB outperforms all these models, indicating that deep model can learn more features than these basic models.

In this experiment, RNN model performs its advantage in sequential feature extraction. The performance of DeepLOB and DeepFolio obviously increase compared to FI-2010, since more time features and less price volume features are in the new dataset. OCET does not show any advantages compared to CNN+RNN model, since it strongly based on price volume features, but pay less attention to the time features. Transformer-LOB, however, contains learnable time embedding layer in each time encoder, which fully embed the time features. Also, deeper model with larger parameters allows it to learn more information from the dataset, which allows Transformer-LOB to learn more information from the dataset. However, the model did not perform better as the depth increase.

5. Conclusion

This work had verified by experiments that Transformer can be directly used into STP task and performs well in both two LOB datasets. Transformer-LOB, a model based on Transformer structure, is modified specifically for the STP task on LOB datasets. The main modifications are made to the time embedding part, corresponding to the positional encoding part in Transformer, and the stacking approach of

Transformer blocks, which enables the model to better incorporate both time features and price-volume features. These modifications were made to adapt the model to the task's specific requirements, resulting in significant improvements in performance.

The results show that increasing the model depth to a certain extend can improve its performance. However, further increasing the depth does not lead to more improvements. This is caused by the nature of data, whose limited features and restricted quantity lead to the large-scale models cannot showcase any advantages compared to small models.

Further works will focus on the emergence of large models in LOB data, which will involve the utilization of larger datasets and model parameters. Moreover, in DeepLOB's work, the feature extraction module used CNN, which aligns more with the data structure of LOB data, DeepFolio, OCET and some other works are all following this feature extraction method, while Transformer-LOB simply uses linear layer to extract the features, which may lead to being trapped into local optima during the optimization process. Corresponding architecture will be modified to achieve better feature extraction performance in the future. Another direction that worth to explore is the zero-shot capability of the model on LOB datasets, including testing on different stocks, different industry domains, and different trading markets with different frequencies. Testing results in this work indicate that the model performances on LOB with longer time spans, i.e., lower frequency, is not as good as on dataset with higher frequency. This is partially caused the characteristics of the data itself, while it is also due to the weaker time extraction capability of the model. This will be addressed and improved in future work.

References

[1] Ahn, H.-J., Cai, J., Hamao, Y., Ho, R.Y.K.: The components of the bid–ask spread in a limit-order market: Evidence from the Tokyo Stock Exchange. Journal of Empirical Finance. 9, 399–430 (2002).

[2] Aitken, M.J., Berkman, H., Mak, D.: The use of undisclosed limit orders on the Australian Stock Exchange. Journal of Banking &amp; Finance. 25, 1589–1603 (2001).

[3] Anagnostidis, P., Papachristou, G., Thomaidis, N.S.: Liquidity commonality in order-driven trading: Evidence from the Athens Stock Exchange. Applied Economics. 48, 2007–2021 (2015).

[4] Thakor, A.V., A., B.A.W., Parlour , C.A., Seppi, D.J.: Chapter 3 - limit order markets: A survey. In: Handbook of Financial Intermediation and banking. pp. 63–96. North-Holland/Elsevier, Amsterdam, San Diego (2008).

[5] Carrie, C.: The new electronic trading regime of dark books, mashups and algorithmic trading. The Journal of Trading. 1, 14-20 (2006)

[6] Bollerslev, T., Marrone, J., Xu, L., Zhou, H.: Stock return predictability and variance risk premia:

Statistical Inference and International evidence. SSRN Electronic Journal. (2012).

[7] Ferreira, M.A., Santa-Clara, P.: Forecasting stock market returns: The sum of the parts is more than the whole. Journal of Financial Economics. 100, 514–537 (2011).

[8] Sirignano, J., Cont, R.: Universal features of Price Formation in financial markets: Perspectives from Deep Learning. SSRN Electronic Journal. (2018).

[9] Atsalakis, G.S., Valavanis, K.P.: Surveying stock market forecasting techniques – part II: Soft computing methods. Expert Systems with Applications. 36, 5932–5941 (2009).

[10] Kercheval, A.N., Zhang, Y.: Modelling high-frequency limit order book dynamics with support Vector Machines. Quantitative Finance. 15, 1315–1329 (2015).

[11] Passalis, N., Tefas, A., Kanniainen, J., Gabbouj, M., Iosifidis, A.: Temporal bag-of-features learning for predicting mid price movements using High Frequency Limit Order Book Data. IEEE Transactions on Emerging Topics in Computational Intelligence. 4, 774–785 (2020).

[12] Tsantekidis, A., Passalis, N., Tefas, A., Kanniainen, J., Gabbouj, M., Iosifidis, A.: Forecasting stock prices from the limit order book using Convolutional Neural Networks. 2017 IEEE 19th Conference on Business Informatics (CBI). (2017).

[13] Tsantekidis, A., Passalis, N., Tefas, A., Kanniainen, J., Gabbouj, M., Iosifidis, A.: Using deep learning to detect price change indications in financial markets. 2017 25th European Signal Processing Conference (EUSIPCO). (2017).

[14] Zhang, Z., Zohren, S., Roberts, S.: DeepLOB: Deep convolutional neural networks for limit order books. IEEE Transactions on Signal Processing. 67, 3001–3012 (2019).

[15] Sangadiev, A., Rivera-Castro, R., Stepanov, K., Poddubny, A., Bubenchikov, K., Bekezin, N., Pilyugina, P., Burnaev, E.: DeepFolio: Convolutional Neural Networks for portfolios with Limit Order Book Data, https://arxiv.org/abs/2008.12152.

[16] Yang, P., Fu, L., Zhang, J., Li, G.: OCET: One-dimensional convolution embedding transformer for&nbsp;stock trend prediction. Communications in Computer and Information Science. 370–384 (2023).

[17] Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, L., Polosukhin, I.: Attention is all you need, https://arxiv.org/abs/1706.03762.

[18] Devlin, J., Chang, M.-W., Lee, K., Toutanova, K.: Bert: Pre-training of deep bidirectional Transformers for language understanding, https://arxiv.org/abs/1810.04805.

[19] Radford, A., Narasimhan, K.: https://openai.com/research/language-unsupervised.

[20] Radford, A., Wu, J., Child, R., Luan, D., Amodei, D., Sutskever, I.: https://cdn.openai.com/better-language-models/language_models_are_unsupervised_multitask_learners.pdf.

[21] Brown, T.B., Mann, B., Ryder, N., Subbiah, M., Kaplan, J., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., Agarwal, S., Herbert-Voss, A., Krueger, G., Henighan, T., Child, R., Ramesh, A., Ziegler, D.M., Wu, J., Winter, C., Hesse, C., Chen, M., Sigler, E., Litwin, M., Gray, S., Chess, B., Clark, J., Berner, C., McCandlish, S., Radford, A., Sutskever, I., Amodei, D.: Language models are few-shot learners, https://arxiv.org/abs/2005.14165.

[22] Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., Uszkoreit, J., Houlsby, N.: An image is worth 16x16 words: Transformers for image recognition at scale, https://arxiv.org/abs/2010.11929.

[23] Ntakaris, A., Magris, M., Kanniainen, J., Gabbouj, M., Iosifidis, A.: Benchmark dataset for mid-price forecasting of limit order book data with Machine Learning Methods. Journal of Forecasting. 37, 852–866 (2018).

[24] Gould, M.D., Porter, M.A., Williams, S., McDonald, M., Fenn, D.J., Howison, S.D.: Limit order books, https://arxiv.org/abs/1012.0349.