An application of graph neural networks for stock market data

Dragana Radojičić*, Nina Radojičić**
* University of Belgrade, Faculty of Economics, Serbia
** University of Belgrade, Faculty of Mathematics, Serbia
dragana.radojicic@ekof.bg.ac.rs, nina@matf.bg.ac.rs

***Abstract*—This research is developed in order to describe the behavior present in the market and Limit Order book dynamics, using the concepts of supervised and unsupervised learning. The main mathematical object of interest is the limit order book, whose job is to keep track of all incoming and outgoing orders. There is a wide variety of possibilities to be explored for how to use machine learning techniques to get insights into market behavior. More precisely, in order to develop a statistical arbitrage strategy, the leverage of machine learning techniques can be employed. Furthermore, the concept can be enhanced with the feature that interprets the relationship of different features previously extracted from the limit order book data. The main idea is to employ a Graph Neural Network in order to describe the relationship between different features, and that relationship can be seen as a new feature that is potentially informative and possesses the power to uncover hidden and unknown knowledge from the data set. This work studies the ability to use Graph Neural Networks in order to get more insights from the stock market data. More precisely, this work investigates the ability to use Graph Neural Networks to label the stock market data into one of the labels from the set S={sell, buy, idle}. The obtained results are examined by using the F-score measure and compared with the results obtained by using the recurrent neural networks. This study discusses the potential for using GNNs for stock market data.**

I. Introduction

The last decades brought a new way of trading and the automatization of the trading process permits users to set predetermined rules for the algorithm to automatically execute a trade. Nowadays, many trade events are occurring via the electronic stock exchanges (see [1]), and there is a possibility to apply machine learning in order to develop trading strategies.

A novel concept to abstract the knowledge base component from the fuzzy rule-based system in order to predict the stock market is obtained in [2]. On the other hand, in order to predict the stock trading signals, the authors in the paper [3] introduced an approach based on principal components accompanied by the weighted support vector machine method.

Whereas in [4], a method that extracts relevant information about financial risk from descriptive text data using a deep learning approach is presented. The authors in [5] have evaluated system risk using market data enhanced with the text data from financial tweets. The authors in [6] improve the time-varying risk-adjusted performance of trading systems of AI models.

Since historical stock market data is more available than before, there is an increasing interest in research of LOB dynamics modeling. Researchers both from academia and industry are interested to simulate LOB dynamics, in order to develop trading and execution strategies. It is known that the stock market is volatile, which is caused by various macro and micro factors, such as financial states, global economic events, unexpected events, politics, social media news, etc. The main tasks of algorithmic trading are exploring sophisticated techniques to find patterns in data and developing trading strategies to automatically execute orders. There is a variety of options to use traditional quantitative finance tools accompanied by sophisticated machine learning approaches to obtain profit.

Precisely in this research, the idea is to extract attributes from the raw stock market data and to calculate standardly known technical indicators to build up a system consisting of different features including technical indicators extracted from the data. Each feature derived within the data transformation part provides an additional piece of information. Note that the informativeness of the features used as an input of the chosen model has a strong influence on the level of the insights which will be gained during the learning process. The main idea of this research is to introduce the Graph Neural Network (GNN) model whose job is to classify each data vector into one of the labels from the set S = {buy, sell, idle}, i.e. the proposed GNN model outputs an action whether it is a good time to buy, sell, or idle.

A. The Limit Order Book

The Limit Order Book (LOB) is the dominant trading tool used to record all outstanding buy and sell orders. The LOB is a two-sided object defined on a discrete price grid. Each price recorded on the stock market is represented by a discrete point in the LOB. The minimum distance between two prices at the LOB is called tick. The first price below the best bid price is placed on the price level 2 on the bid side, while the lowest available price higher than the best ask price is placed on the price level 2 on the ask side, etc. The hidden orders are utilized in order to hide the true volume of the large orders by breaking up a large order into multiple small individual orders. The snapshots of the Apple and Microsoft stock market data are depicted in Fig. 1 and Fig.2, where the orders which need to be bought are colored yellow and placed on the *bid side* of the lob, while the orders which are recorded in order to be sold are colored blue and placed at the *ask side.*

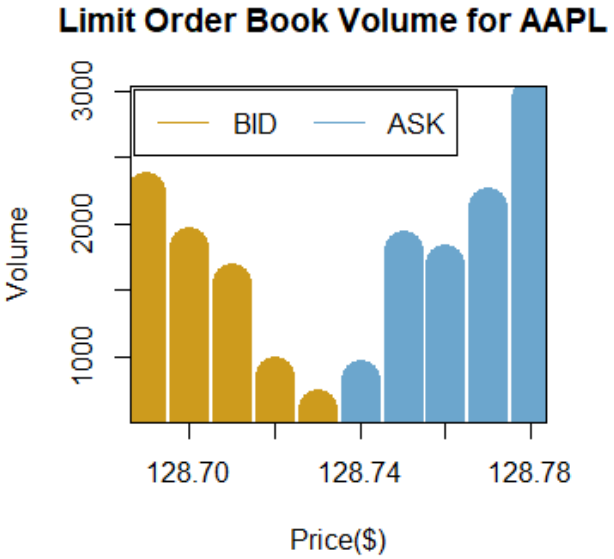

Figure 1. Snapshot of the NASDAQ limit order book for AAPL (Apple Inc. Company) stock symbol for 5 levels. On the bid/ask side are placed outstanding buy/sell orders (gold/blue), and the best bid price is $128.73 with a volume of 60 shares, while the best ask price is $128.73 with a volume of 100 shares.

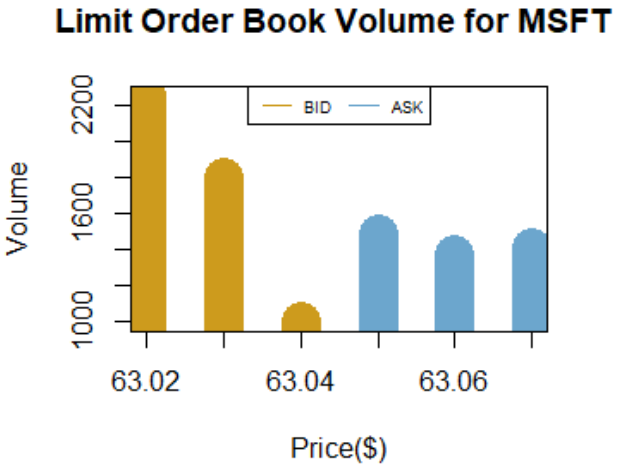

Figure 2. Snapshot of the NASDAQ limit order book for MSFT (Microsoft Company) stock symbol for 3 levels. On the bid/ask side are placed outstanding buy/sell orders (gold/blue), and the best bid price is $63.04 with a volume of 150 shares, while the best ask price is $63.05 with a volume of 1600 shares.

It is a challenging task to work with huge data sets such as stock market data. Thus, in order to proceed with research sophisticated data processing should be applied to transform the data set and extract relevant attributes from the data set. There is an increasing number of people interested in the research of modeling LOBs since stock market historical data is more easily accessible than before.

B. The data

In this research, a high-quality limit order book data set, namely, the LOBSTER (see [7]) data set is used. Note that LOBSTER replicates entire NASDAQ order book files, and therefore it is a massive data set for which researchers need to use sophisticated preprocessing techniques in order to be able to deal with massive data sets. For each trading day and for each company listed on NASDAQ, LOBSTER outputs a 'message' file and an 'order book' file. The 'message' file posses information about the event which has occurred and therefore has changed the shape of the limit order book. Precisely, the 'message' file provides us with the following information:

- *The time* when an event has occurred which is measured with a precision of at least milliseconds
- *The type* of the event that has occurred, i.e. execution of the order, submission of an order, deletion of order, etc.
- *Order ID* which is a reference number of a trader
- *The size* which models the volume or the number of shares for which an event has occurred
- *The Price* for which trade has occurred
- *The direction* tells us if the order has occurred on the ask or on the bid side, i.e. is it a buy or sell order.

The ‘order book’ file provides information about the prices and corresponding volumes up to the requested number of levels on the ask and bid side.

C. The data reconstruction

Since lob data is huge and therefore challenging for processing, in [8] the framework for processing the LOBSTER dataset is proposed. Once the preprocessing of the data set is enabled we can start transforming the data set and extract relevant features in order to grasp some knowledge about market behavior. Motivated by the system of data reconstruction of limit order book databases proposed in [9], we develop data transformation customized for particularly our research interest, during which characteristics of interest can be extracted. The data transformation part is conducted within the following three parts: data aggregation, data enrichment with technical indicators, and data labeling. The quantitative characteristics such as the number of executed orders, stock prices, historical returns, etc. are useful to reveal the dynamics of the order book data.

Firstly, in order to process an immense amount of data such as the stock market data, we implement the data aggregation. Precisely, for each 60 seconds length non-overlapping interval the features that encapsulate dynamics of the order book during that time interval are extracted. Thus, from each 60 seconds time interval we obtain the market data vector consisting of features such as the number of executed orders during that interval, the average best ask price, the average price placed at the second level on the ask side, the number of hidden orders, the price at which the last trade occurred within that time

interval (called the closed price), the price at which the first trade within that time interval occurred (called the open price), etc.

There exists a wide range of technical indicators that are used in the trading industry (see [10], [11]). Those technical indicators possess the power to describe order book dynamics and could be useful to identify the future price trend. Thus, by using the algorithms' implementations available in the open library *ta-lib* (see [12]), we compute technical indicators for each vector of the aggregated order book data. Hence, at the end of this step, our data set consists of market data vectors, and each vector contains features extracted during time aggregation and also the calculated technical indicators.

The Algorithm, which is employed to assign the label 'buy', 'sell', or 'idle' to each vector in a training set, is inspired by the fact that we can obtain a certain profit by only being exposed to a certain risk. The idea of the labeling Algorithm is to examine at the time whether any subsequent price reaches the upper or lower bound, which are calculated from the current considered price and predefined upper threshold and lower threshold, which the algorithm takes as input. We examine if any subsequent price is higher than the target reward value, then the label is bought. On the other hand, if a subsequent price is lower than the current stop-loss, the label is sold. Finally, if the previous criteria are not met for any subsequent market data vector, the label is idle.

The paper [13] presents the model based on the Long short-term memory (LSTM) network developed for the LOBSTER data set, and the topology of the model is based on LSTM in order to capture the time dependency between rows in the market data. Furthermore in [14], by applying Fourier transforms new characteristics are extracted from the stock exchange database, and a model based on the Gated Recurrent Unit (GRU) is introduced to classify the market data vector of the LOBSTER data set. In [15], it is evaluated whether the performance of the network model based on the LSTM topology is improved when the features are selected with respect to the newly proposed methods.

II. Methodology

Deep Learning has been proven to successfully capture hidden patterns of Euclidean data (images, text, videos) [14]. The complexity of graph data created challenges for existing machine learning algorithms. Neural networks adapted to leverage the structure and properties of graphs are called Graph Neural Networks (GNNs) [17]. The most fundamental part of a GNN is a graph G=(V, E). where V is the set of nodes and E are the edges between nodes. The edges can be directed, if there are directional dependencies, and undirected otherwise. Real-world data is often represented as graphs, and thus, GNNs have been becoming more insightful. There have been successfully applied for text classification [18], traffic forecasting [19], etc.

The main idea is to employ a Graph Neural Network (GNN) in order to describe the relationship between different features, and that relationship can be seen as a new feature that is potentially informative and possesses the power to uncover hidden and unknown knowledge from the data set. Thus, the detection algorithm should be developed such that it captures relations between different features as a new dimension.

GNNs are able to capture the dependence of graphs via message passing between the nodes of graphs. Therefore, GNNs can be useful to model the interconnections between different market features that are previously extracted from the order book.

In this work, the market data vector is represented as a weighted graph G=(V, E) in order to express the complex interactions between features. Each node from set V corresponds to one feature, while the edges between each of them with weight equal to the correlation value between the nodes. These edges are represented with an adjacency matrix A

$$a_{ij} = corr(i, j),\ if\ i \neq j. \quad (1)$$

$$a_{ij} = autocorr(i, j),\ if\ i = j. \quad (2)$$

III. Experimental results

All algorithms were implemented in Python using the machine learning framework PyTorch. The question for the GNN is if it is a good time to sell, buy, or idle.

Thus, we performed a supervised learning approach on previously labeled data, as explained above. During the training phase of the GNN, back-propagation of the loss function is performed. Finally, testing is done on not previously seen data. In order to measure the success of the newly proposed GNN approach, precision, recall, and F1 score were calculated for both GNN and the Gated recurrent unit (GRU) approach from [9]. The presented results demonstrate that the newly proposed GNN approach overperformed the state-of-the-art on the three measures and used data set.

Table 1. A comparison of the presented GNN approach with the state-of-the-art GRU

	GRU	GNN
Precision	0.69	0.72
Recall	0.33	0.41
F1	0.45	0.52

IV. Conclusion

In this paper, we investigated the ability to use Graph Neural Networks to label the stock market data into one of the labels from the set S={sell, buy, idle}. The obtained results are examined by using the F-score measure and compared with the results obtained by using the GRU-based neural network, which was shown to

work well on the stock market data. The newly proposed GNN showed promising results, overperforming the GRU approach in the presented experimental study.

Future work can be continued in several directions. We will try to introduce some new features in order to explore how that would affect the proposed model, as well as to extract features using wallet transformations (see [20]). Furthermore, the use of social networks as a source of relevant information can be powerful and can extend our set of extracted features.

Acknowledgment

The authors are thankful to Thorsten Rheinlander for his help in doing the LOBSTER data analysis and for valuable and insightful suggestions.

References

[1] le Calvez, Arthur, and Dave Cliff. "Deep learning can replicate adaptive traders in a limit-order-book financial market." In 2018 IEEE Symposium Series on Computational Intelligence (SSCI), pp. 1876-1883. IEEE, 2018.

[2] Asadi, Shahrokh."Evolutionary fuzzification of RIPPER for regression: Case study of stock prediction." Neurocomputing 331 (2019): 121-137.

[3] Chen, Yingjun, and Yijie Hao. "Integrating principle component analysis and weighted support vector machine for stock trading signals prediction." Neurocomputing 321 (2018): 381-402.

[4] Rönnqvist, Samuel, and Peter Sarlin. "Bank distress in the news: Describing events through deep learning." Neurocomputing 264 (2017): 57-70.

[5] Cerchiello, Paola, Paolo Giudici, and Giancarlo Nicola. "Twitter data models for bank risk contagion." Neurocomputing 264 (2017): 50-56.

[6] Vella, Vince, and Wing Lon Ng. "Enhancing risk-adjusted performance of stock market intraday trading with neuro-fuzzy systems." Neurocomputing 141 (2014): 170-187.

[7] https://lobsterdata.com. Accessed: 2022-03-05

[8] Dragana Radojičić and Simeon Kredatus; An approach for processing data from NASDAQ stock exchange database, Proceedings of the 10th International Conference on Information Society and Technology (ICIST 2020), Serbia Proceedings Vol.2, pp.256-259, 2020

[9] Radojičić, Dragana, Simeon Kredatus, and Thorsten Rheinländer. "An approach to reconstruction of data set via supervised and unsupervised learning." In 2018 IEEE 18th International Symposium on Computational Intelligence and Informatics (CINTI), pp. 000053-000058. IEEE, 2018.

[10] Murphy, John J. Technical analysis of the financial markets: A comprehensive guide to trading methods and applications. Penguin, 1999.

[11] Colby, Robert W., and Thomas A. Meyers. "The Encyclopedia of Technical Stock Market Indicators." Dow Jones-Irwin 11 (1988): 270.

[12] Ta-lib: Technical analysis library. https://www.ta-lib.org/. Accessed: 2022-05-03.

[13] Dragana Radojičić; An LSTM neural network model for stock market data, Proceedings of the 11th International Conference on Information Society and Technology (ICIST 2021), Serbia Proceedings, pp.173-177, 2021, ISSN 2738-1447

[14] Radojičić, Dragana, and Simeon Kredatus. "The impact of stock market price Fourier transform analysis on the gated recurrent unit classifier model." Expert Systems with Applications 159 (2020): 113565.

[15] Radojičić, Dragana, Nina Radojičić, and Simeon Kredatus. "A multicriteria optimization approach for the stock market feature selection." Computer Science and Information Systems 18, no. 3 (2021): 749-769.

[16] Goodfellow, Ian, Yoshua Bengio, and Aaron Courville. *Deep learning*. MIT press, 2016.

[17] Scarselli, F., Gori, M., Tsoi, A.C., Hagenbuchner, M. and Monfardini, G., 2008. The graph neural network model. IEEE transactions on neural networks, 20(1), pp.61-80.

[18] Malekzadeh, M., Hajibabaee, P., Heidari, M., Zad, S., Uzuner, O. and Jones, J.H., 2021, December. Review of graph neural network in text classification. In 2021 IEEE 12th Annual Ubiquitous Computing, Electronics & Mobile Communication Conference (UEMCON) (pp. 0084-0091). IEEE.

[19] Bui, K.H.N., Cho, J. and Yi, H., 2021. Spatial-temporal graph neural network for traffic forecasting: An overview and open research issues. Applied Intelligence, pp.1-12.

[20] Meyer, Yves. Wavelets and Operators: Volume 1. No. 37. Cambridge university press, 1992.