공학석사학위논문

과거 가격 및 희소한 트윗을 이용한 주가 변동 예측

Stock Price Movement Prediction from Historical Prices and Sparse Tweets

서울대학교 대학원
컴퓨터공학부
조민용

과거 가격 및 희소한 트윗을 이용한 주가 변동 예측

Stock Price Movement Prediction from Historical Prices and Sparse Tweets

지도교수 강 유

이 논문을 공학석사 학위논문으로 제출함

2021년 6월

서울대학교 대학원
컴퓨터공학부
조민용

조민용의 석사 학위논문을 인준함

2021년 6월

위 원 장 김 형 주

부위원장 강 유

위 원 문 봉 기

Abstract

Stock Price Movement Prediction from Historical Prices and Sparse Tweets

Minyong Cho
Department of Computer Science & Engineering
The Graduate School
Seoul National University

Given historical stock prices and sparse tweets mentioning the stocks to predict, how can we precisely predict stock price movement? Many market analysts strive to use a large amount of information for prediction. However, they confront more noise when utilizing larger data for prediction. Thus, existing methods use only historical prices, or those along with a small amount of refined data such as news articles or tweets mentioning target stocks. However, they have the following limitations: 1) using only historical prices gives low performance since they have insufficient information, 2) news articles lack timeliness compared to social medias for predicting stock price movement, and 3) the previous methods using tweets do not handle stocks without tweets mentioning them.

In this paper, we propose GLT (Stock Price Movement Prediction using **G**lobal and **L**ocal **T**rends of Tweets), an accurate stock price movement prediction method that works without tweets mentioning target stocks. GLT pre-trains both of stock and tweet representations in a self-supervised way. Then, GLT generates global and local tweet trends which represent global public opinion and the local trends related to target stocks, respectively. The trend vectors are combined to accurately predict stock price movement. Experimental results show that GLT provides the state-of-the-art accuracy for stock price movement prediction.

Keywords : Stock Price Movement Prediction, Self-Supervised Learning, Recurrent Neural Network, Time Series Classification, Social Media Dataset
Student Number : 2019-26047

Table of Contents

I. Introduction **1**

II. Related Work **4**

2.1 Stock Price Movement Prediction 4

2.2 Attentive LSTM 4

III. Proposed Method **6**

3.1 Overview 6

3.2 Self-supervised Pre-training for Representing Tweets and Stocks 7

3.3 Global Tweet Trend 10

3.4 Local Tweet Trend 11

3.5 Stock Movement Prediction 11

IV. Experiment **13**

4.1 Experiment Setting 13

4.2 Classification Performance 15

4.3 Ablation Study 16

4.4 Hyperparameter Robustness 16

V. Conclusion **18**

References **19**

Abstract in Korean **23**

List of Figures

Figure 1. The proposed GLT consistently gives the best performance in all datasets and metrics, thanks to its consideration of both global and local tweet trends. 2
Figure 2. Self-supervised pre-training for learning tweet and stock representations. 8
Figure 3. The overall framework of stock price movement prediction model. 9
Figure 4. Robustness of GLT. Blue points show the performance of GLT, while the green dashed lines denote the best performance among baselines. GLT with any dimension of hidden units outperforms all baselines. 17

List of Tables

Table 1. Price features representing the daily trend of a stock. 10
Table 2. Summary of datasets. 14
Table 3. The classification performance of GLT and competitors, measured with the accuracy (Acc) and the Matthews Correlation Coefficient (MCC); higher values mean better performances. GLT shows the best performance for all the cases. 15
Table 4. Result of ablation study. The global tweet trend and the local tweet trend of GLT improve the classification performance of prediction. 16

Chapter 1

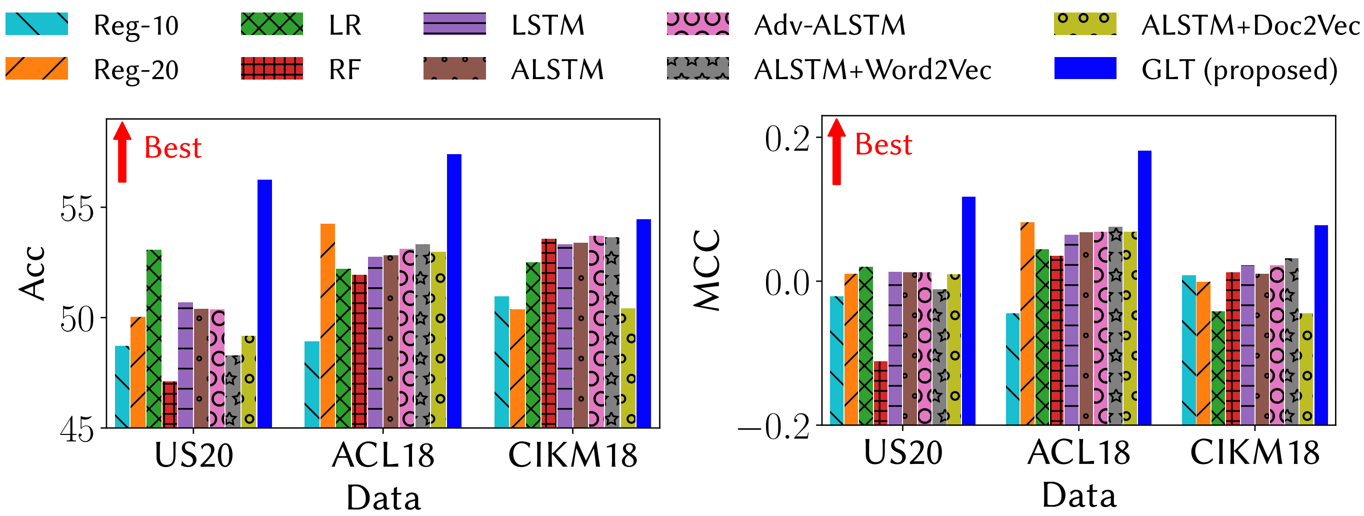
Introduction

Given historical stock prices and sparse tweets, how can we accurately predict stock price movement? Stock price prediction is an important task that has drawn much attention recently in data mining and machine learning communities [1, 2, 3, 4, 5, 6, 7].

Most recent studies use only historical stock prices; however, they give limited performance since they do not exploit the rich, fundamental information in other data including tweets [2, 8]. [9, 10] use news data for stock prediction; however, in many cases a news about an event is released after the event has affected the price of stocks. Although [4, 11] used tweets mentioning target stocks for prediction, they do not work for stocks without tweets mentioning them.

In this paper, we propose GLT (Stock Price Movement Prediction using **G**lobal and **L**ocal **T**rends of Tweets), an accurate approach for predicting stock price movement considering both historical stock prices and sparse tweets. GLT predicts stock price movement even when there are no or very few tweets mentioning the target stocks. GLT uses self-supervised pre-training for learning representations of tweets and stocks. GLT captures the global market trend by carefully exploiting the tweets and historical prices. Furthermore, GLT captures local tweet trend by exploiting other stocks' tweets to predict the movement of a target stock. Thanks to the exploitation of tweets and historical prices, GLT provides the state-of-the performance for stock movement prediction.

We summarize our main contributions as follows:



(a) Accuracy

(b) Matthews correlation coefficient

Figure 1: The proposed GLT consistently gives the best performance in all datasets and metrics, thanks to its consideration of both global and local tweet trends.

- **Self-supervised pre-training for representing tweets and stocks as latent vectors.** GLT learns both tweet and stock embeddings by coupling a masked language model and the embedding matrix of stocks in a self-supervised way, exploiting rich unlabeled tweet data.
- **Capturing global and local tweet trends.** GLT captures and exploits both global and local tweet trends, since both of them affect the prediction of stock price movement. GLT works even when we have no or very few tweets on target stocks.
- **Experiment.** Extensive experiments show that GLT provides the best accuracy, outperforming competitors by significant margins (see Figure 1).

In the rest of paper, we review related works in Section 2, introduce our proposed method GLT in Section 3, evaluate GLT and competitors in Section 4, and conclude in Section 5. The source code and datasets used in this paper are available at https://github.com/paper-submission-anony-21/glt.

Chapter 2

Related Work

2.1 Stock Price Movement Prediction

Stock price prediction has attracted growing research interests [1, 2, 3, 4, 5, 6, 7]. Most of the recent methods predict stock price with deep neural networks using historical stock prices [2, 12, 4, 11, 8, 13, 9, 10]. In addition to the historical prices, [9, 10] consider news events related to stocks, and [4, 11] use tweets mentioning target stocks to find useful patterns.

We note that none of the above methods make prediction for target stocks using tweets referring to the other stocks.

2.2 Attentive LSTM

The attention mechanism has been widely used to process sequential data [14, 2, 15]. The Attentive LSTM (ALSTM) is a variant of LSTM [16] that utilizes the attention mechanism. ALSTM encodes the sequence of hidden states $[\boldsymbol{h}_1^s, \boldsymbol{h}_2^s, \ldots, \boldsymbol{h}_T^s]$ for target stock s into one single *overall representation* $\boldsymbol{a}^s$ with adaptive weights:

$$\boldsymbol{a}^s = \sum_{t=1}^{T} \gamma_t^s \boldsymbol{h}_t^s,\ \gamma_k^s = \frac{\exp(\tilde{\gamma}_k^s)}{\sum_{t=1}^{T} \exp(\tilde{\gamma}_t^s)},\ \tilde{\gamma}_k^s = \boldsymbol{u}^\intercal \tanh(\boldsymbol{W}\boldsymbol{h}_k^s + \boldsymbol{b}) \tag{2.1}$$

where $\boldsymbol{W} \in \mathbb{R}^{E\times U}$, $\boldsymbol{u} \in \mathbb{R}^E$, and $\boldsymbol{b} \in \mathbb{R}^E$ are parameters to be learned; U and E are the dimension of hidden representations; T is the size of the lag window.

ALSTM concatenates the last hidden state $\boldsymbol{h}_T^s$ with $\boldsymbol{a}^s$ to make the *final representation* $\boldsymbol{e}^s \in \mathbb{R}^{2U}$. Then ALSTM uses a linear layer to predict the classification confidence $\hat{y}^s$ as follows:

$$\boldsymbol{e}^s = [\boldsymbol{h}_T^s; \boldsymbol{a}^s],\ \hat{y}^s = \boldsymbol{w}^\intercal \boldsymbol{e}^s + c \tag{2.2}$$

where $\boldsymbol{w} \in \mathbb{R}^{2U}$ and $c \in \mathbb{R}$ are parameters to be learned. ALSTM finally gets binary classification estimation as $sign(\hat{y}^s)$. Our method makes a stock price movement prediction based on the ALSTM.

Chapter 3

Proposed Method

We propose GLT, an accurate method for stock movement prediction using historical prices and tweets.

3.1 Overview

We design GLT for capturing the relations between tweet trends and stocks to provide an accurate prediction. We concentrate on the following challenges to enhance the performance of prediction.

1. **Learning representations of tweets and stocks.** Finding the right representations for tweets and stocks are crucial for modeling. How can we learn the representations using a large amount of tweets?

2. **Capturing global trend.** Global market sentiment affects the price movement of each stock. How to capture the global trend, and use it for individual stock's prediction?

3. **Capturing local trend despite sparseness.** How can we capture local trend of a target stock, even when we have no or very few tweets for it?

We address the aforementioned challenges by the following ideas:

1. **Self-supervised pre-training.** We simultaneously learn both tweet and stock embeddings by coupling a masked language model and the embedding matrix

of stocks in a self-supervised way (Section 3.2).

2. **Global tweet trend.** We capture the global trend using tweets. All the tweets of each day and historical prices are combined to extract features representing the global trend (Section 3.3).

3. **Local tweet trend by exploiting other stocks' tweets.** We exploit tweets mentioning other stocks, in addition to those mentioning the target stock, for capturing the local trend (Section 3.4).

Figure 3 shows the overall architecture of GLT. We first pre-train the tweet and stock embeddings. Then, using the learned embeddings we extract the global and the local trends, and perform the prediction of stock price movement. Except the pre-training step, GLT is learned in an end-to-end way.

3.2 Self-supervised Pre-training for Representing Tweets and Stocks

Our first step is to learn tweet and stock embeddings. This step is crucial, since learning appropriate representations greatly helps improve the accuracy of prediction model. Our idea is to use self-supervised pre-training for such learning. Assume that we have a tweet "Thank you Apple for the new iPhone". We learn embeddings of the tweet and stocks by masking the word Apple, and predicting the true label of the mask as Apple. For the purpose, we replace the word Apple with [MASK] in the tweet. Then, we feed the replaced tweet into a bi-directional LSTM (BiLSTM) model which outputs the representation $\boldsymbol{h}_{\text{[MASK]}} \in \mathbb{R}^w$, as shown in Figure 2. The output vector $\boldsymbol{h}_{\text{[MASK]}}$ is multiplied to the stock embedding matrix $S \in \mathbb{R}^{n \times w}$, where n is

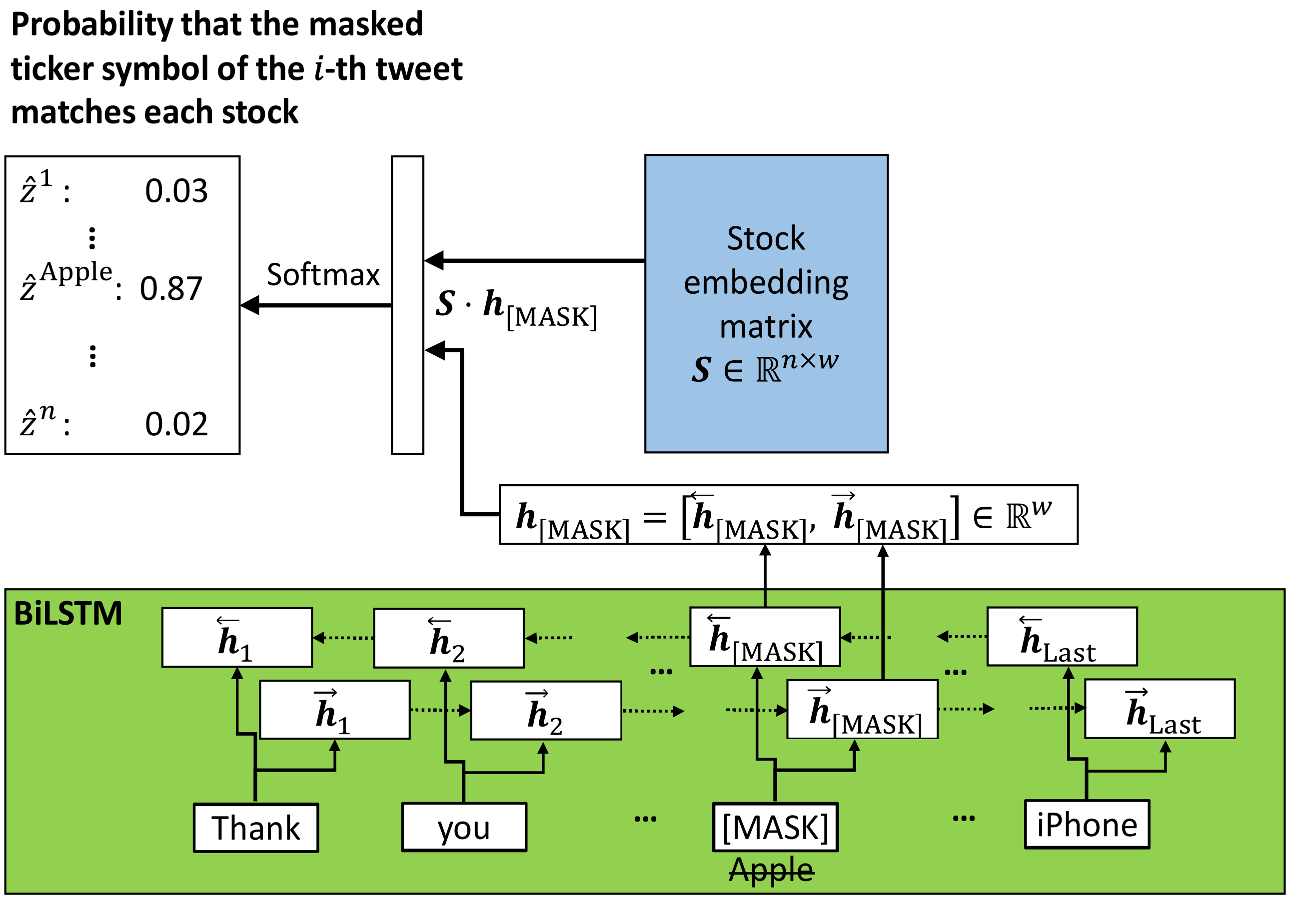


Figure 2: Self-supervised pre-training for learning tweet and stock representations.

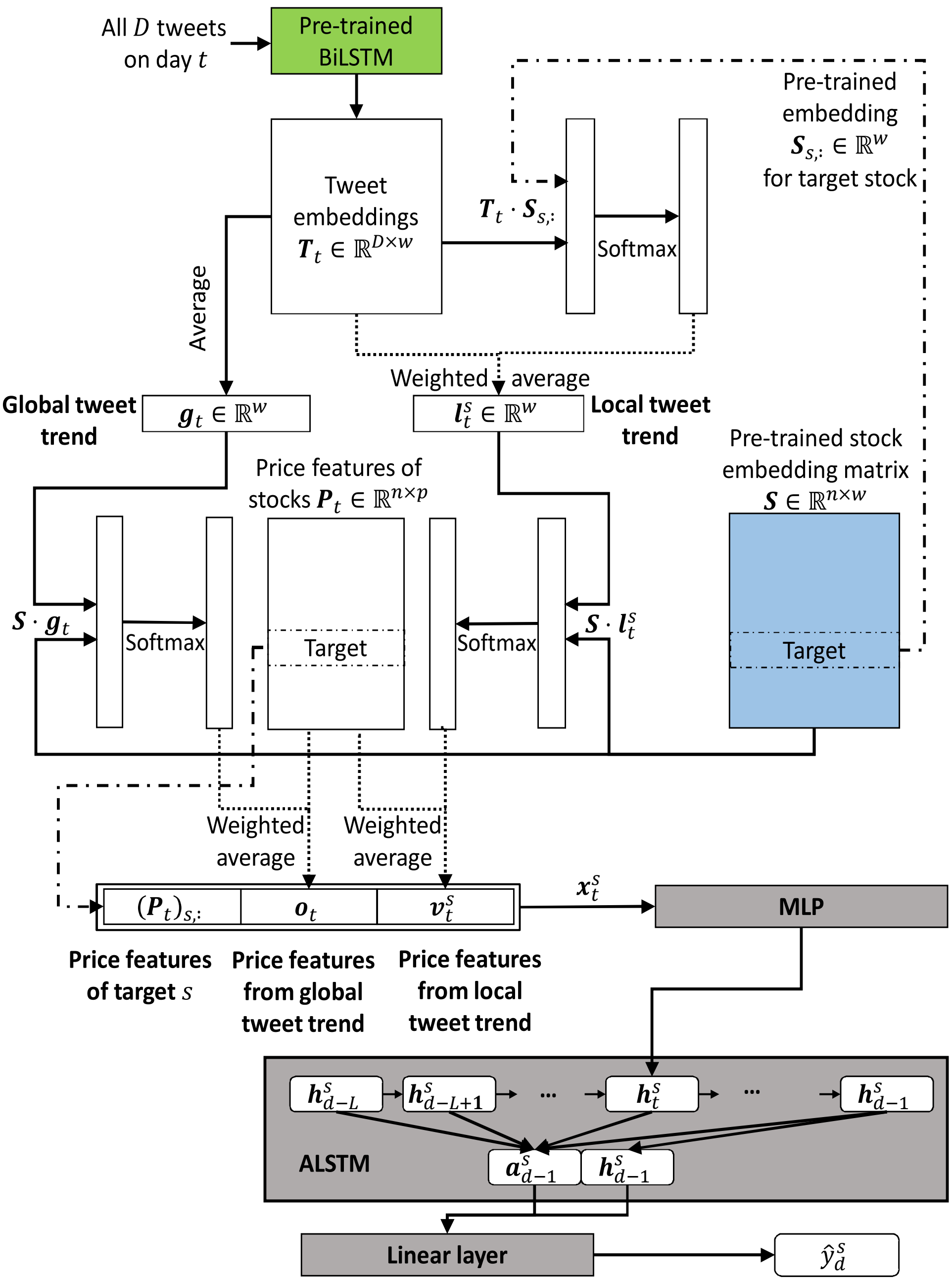


Figure 3: The overall framework of stock price movement prediction model.

Table 1: Price features representing the daily trend of a stock.

Features	**Calculation**
$c\_open,\ c\_high,\ c\_low$	$e.g.,\ c\_open = open_t/close_t - 1$
$n\_close,\ n\_adj\_close$	$e.g.,\ n\_close = close_t/close_{t-1} - 1$
$5\_day,\ 10\_day,\ 15\_day$ $20\_day,\ 25\_day,\ 30\_day$	$e.g.,\ 5\_day = \frac{\sum_{i=0}^{4} adj\_close_{t-i}/5}{adj\_close_t} - 1$

the number of all stocks, to output the probability of each stock representing the [MASK] via softmax. After the training, we get the matrix S which contains stock embeddings, and the BiLSTM which generates the embedding of a given tweet.

3.3 Global Tweet Trend

We use the global tweet trend for predicting individual stock's movement, since the movement is affected by the global market sentiment. Our idea is to add the weighted average of historical prices of all stocks, as a feature of the prediction module for price movement. The weights reflect the global tweet trend and its relation to individual stock. Given D tweets on day t, we generate their embeddings $\boldsymbol{T}_t \in \mathbb{R}^{D\times w}$ using the BiLSTM (Section 3.2), and average it to make a global tweet trend vector $\boldsymbol{g}_t \in \mathbb{R}^w$ on day t, as shown in Figure 3. The vector $\boldsymbol{g}_t$ is multiplied to the pre-trained embedding matrix S to generate the weight for each stock, and the weights are used to compute the weighted average $\boldsymbol{o}_t$ of stock price features wrt the global tweet trend. We use the price features listed in Table 1.

3.4 Local Tweet Trend

In addition to the global tweet trend, we use the local tweet trend of a target stock s. A major challenge is the sparseness of tweets mentioning the target stock; if there are no or very few tweets for the target stock, then it is not clear how to perform training or inference. Our idea is to use tweets mentioning other stocks, in addition to those mentioning the target stock, for the prediction. We generate the local tweet trend vector $\boldsymbol{l}_t^s \in \mathbb{R}^w$ of stock s by the weighted average of tweet embeddings where the weight is determined by the relevance of each tweet and the target's stock embedding, as shown in Figure 3. The vector $\boldsymbol{l}_t^s$ is multiplied to the pre-trained embedding matrix S to generate the weight for each stock, and the weights are used to compute the weighted average $\boldsymbol{v}_t^s$ of stock price features wrt the local tweet trend.

3.5 Stock Movement Prediction

The final step is to predict the stock movement using the features computed in the previous step. Given a target stock s, we concatenate its original price features $(\boldsymbol{P}_t)_{s,:}$ with the features $\boldsymbol{o}_t$ from the global tweet trend (Section 3.3) and the features $\boldsymbol{v}_t^s$ from the local tweet trend (Section 3.4). The concatenated features are then fed into an MLP layer, an Attention LSTM, and a linear layer to predict the movement $\hat{y}_d^s$ of the target stock s, as shown in Figure 3. We train our model to minimize the following objective function:

$$\sum_{d \in Q} \sum_{s \in R} l(y_d^s, \hat{y}_d^s) + \frac{\lambda}{2} ||\Theta||_F^2, \; l(y_d^s, \hat{y}_d^s) = max(0, 1 - y_d^s \hat{y}_d^s) \tag{3.1}$$

where Q is a set of trading days in the training dataset, R is a set of stocks to predict, and Θ denotes the parameter of the model. Following [2], we select the hinge loss [17] to focus more on the examples close to the decision boundary. To prevent overfitting, we add a regularizer term to the objective function where λ is a hyperparameter.

Chapter 4

Experiment

We conduct experiments to answer the following questions about the performance of GLT.

- **Q1. Classification performance (Section 4.2).** Does GLT outperform baselines?
- **Q2. Ablation study (Section 4.3).** Do the global tweet trend and the local tweet trend improve the performance of GLT?
- **Q3. Hyperparameter robustness (Section 4.4).** Is GLT robust to the dimension of hidden units?

4.1 Experiment Setting

Dataset. We use three benchmarks to evaluate our proposed method, US20, ACL18 [4], and CIKM18 [11] in Table 2. All datasets consist of high-trade-volume stocks in US stock markets, and US20 is a new dataset that we collect and publicly release. We label the instances according to the increase rate of adjusted closing prices. The increase rate is calculated as $p_d^s/p_{d-1}^s - 1$, where p_d^s is the adjusted closing price of stock s on target day d. Instances with the rate $\geq 0.55\%$ and $\leq -0.5\%$ are labeled as 1 and -1, respectively.

Parameter settings. We optimize GLT using the mini-batch Adam [18] with a batch size of 1,024. We select hyperparameters λ in Eq. (3.1), the learning rate, the dimension of hidden units of ALSTM, and the lag window size via grid-search within

Table 2: Summary of datasets.

Dataset	# Stocks	# Tweets	Train	Valid	Test
US20	50	272,762	07/2019-	03/2020-	05/2020-07/2020
ACL18	87	106,271	01/2014-	08/2015-	10/2015-12/2015
CIKM18	38	955,788	01/2017-	09/2017-	11/2017-12/2017

the [0.01, 0.1, 1], [0.001, 0.005, 0.01], [4, 8, 16, 32], and [2, 3, 4, 5, 10, 15], respectively. We report the mean performance on the test set over five different runs.

Competitors. We compare the performance of our proposed GLT to the following competitors.

- **Reg-10.** A linear regression using recent 10 trading days' closing prices.
- **Reg-20.** A linear regression using recent 20 trading days' closing prices.
- **LR.** A logistic regression using recent N trading days' closing prices, where N is a hyperparameter that performs best on the validation set.
- **RF.** A discriminative random forest classifier using historical stock price features.
- **LSTM.** A vanilla LSTM model using historical stock price features [19].
- **ASLTM.** LSTM with attention layer [8].
- **Adv-ALSTM.** ALSTM with adversarial training [2].
- **ASLTM + Word2Vec.** ALSTM using Word2Vec [20] of recent tweets mentioning the target.
- **ASLTM + Doc2Vec.** ALSTM using Doc2Vec [21] for representing recent tweets mentioning the target.

Evaluation metrics. We use two metrics to evaluate the performance: accuracy (Acc) and Matthews Correlation Coefficient (MCC) [22] which considers the imbalance of classes.

4.2 Classification Performance

We compare the classification performance of GLT and competitors. The result is summarized in Table 3 and Figure 1. Note that GLT consistently provides the best performance on all datasets and metrics; GLT shows up to 3.17%p higher accuracy and 0.0995 higher MCC, compared to second-best methods.

Table 3: The classification performance of GLT and competitors, measured with the accuracy (Acc) and the Matthews Correlation Coefficient (MCC); higher values mean better performances. GLT shows the best performance for all the cases.

	US20		ACL18		CIKM18	
Method	Acc	MCC	Acc	MCC	Acc	MCC
Reg-10	48.71	-0.0212	48.92	-0.0449	50.95	0.0076
Reg-20	50.03	0.0097	54.25	0.0818	50.36	-0.0013
LR	53.07	0.0200	52.20	0.0442	52.50	-0.0425
RF	47.10	-0.1114	51.94	0.0348	53.57	0.0119
LSTM	50.69	0.0127	52.75	0.0639	53.31	0.0216
ALSTM	50.38	0.0123	52.81	0.0677	53.38	0.0100
Adv-ALSTM	50.36	0.0120	53.11	0.0685	53.69	0.0217
ASLTM + Word2Vec	48.28	-0.0116	53.32	0.0754	53.64	0.0315
ASLTM + Doc2Vec	49.16	0.0090	52.98	0.0681	50.40	-0.0449
GLT (proposed)	**56.24**	**0.1172**	**57.40**	**0.1813**	**54.45**	**0.0774**

Table 4: Result of ablation study. The global tweet trend and the local tweet trend of GLT improve the classification performance of prediction.

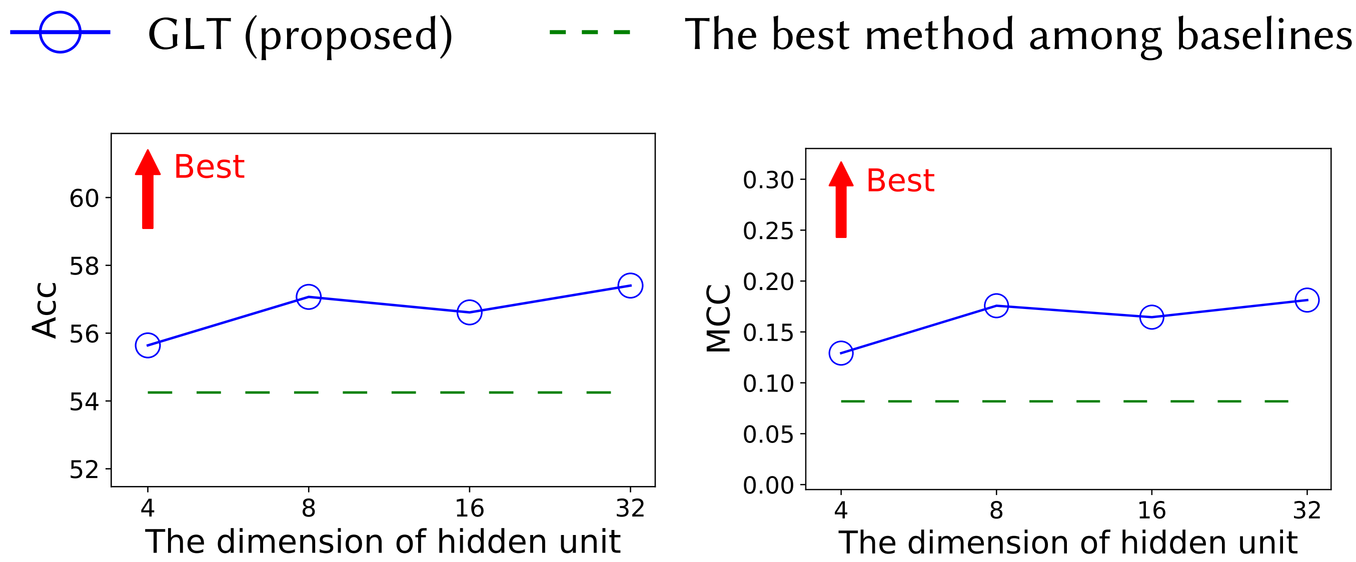
	US20		ACL18		CIKM18	
Method	Acc	MCC	Acc	MCC	Acc	MCC
ALSTM	50.38	0.0123	52.81	0.0677	53.38	0.0100
GLT-G	51.85	0.0355	55.64	0.1432	53.00	-0.0009
GLT-L	55.22	0.0950	53.58	0.0983	**54.55**	0.0480
GLT (proposed)	**56.24**	**0.1172**	**57.40**	**0.1813**	54.45	**0.0774**

4.3 Ablation Study

We compare the performance of GLT with its variants and summarize the result in Table 4. GLT-G is the GLT without the global tweet trend (Section 3.3), and GLT-L is the one without the local tweet trend (Section 3.4). Note that GLT outperforms GLT-G and GLT-L in terms of MCC for all the cases. In terms of Acc, GLT outperforms all of its variants in all but the CIKM18 case; however, even in that case GLT shows the second-best performance.

4.4 Hyperparameter Robustness

We report the robustness of GLT against the dimension of hidden units in Figure 4, for ACL18 dataset. Note that the performance increases when the dimension is increased from 4 to 8, and then it becomes stable. Even when the dimension is 4, GLT outperforms all the baselines.



(a) Accuracy

(b) Matthews correlation coefficient

Figure 4: Robustness of GLT. Blue points show the performance of GLT, while the green dashed lines denote the best performance among baselines. GLT with any dimension of hidden units outperforms all baselines.

Chapter 5

Conclusion

We propose GLT, a prediction model for stock price movement which considers both historical stock prices and sparse tweets. We represent tweets and stocks as embeddings via self-supervised pre-training. GLT then generates the global and local tweet trends to reflect public opinion, and predicts the movement of stocks using the trends and historical stock prices. Experimental results show that GLT provides the state-of-the-art prediction result, by considering both of global and local tweet trends. Future works include extending the method to predict stock movement only with tweets, without using price data. Another direction is to combine news and tweets to understand a more comprehensive context of markets.

References

[1] J. Yoo, Y. Soun, Y. chan Park, and U. Kang, "Accurate multivariate stock movement prediction via data-axis transformer with multi-level contexts," in *KDD*, 2021.

[2] F. Feng, H. Chen, X. He, J. Ding, M. Sun, and T. Chua, "Enhancing stock movement prediction with adversarial training," in *Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence, IJCAI 2019, Macao, China, August 10-16, 2019* (S. Kraus, ed.), pp. 5843–5849, ijcai.org, 2019.

[3] Y. Chen, Z. Wei, and X. Huang, "Incorporating corporation relationship via graph convolutional neural networks for stock price prediction," in *Proceedings of the 27th ACM International Conference on Information and Knowledge Management, CIKM 2018, Torino, Italy, October 22-26, 2018* (A. Cuzzocrea, J. Allan, N. W. Paton, D. Srivastava, R. Agrawal, A. Z. Broder, M. J. Zaki, K. S. Candan, A. Labrinidis, A. Schuster, and H. Wang, eds.), pp. 1655–1658, ACM, 2018.

[4] Y. Xu and S. B. Cohen, "Stock movement prediction from tweets and historical prices," in *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics, ACL 2018, Melbourne, Australia, July 15-20, 2018, Volume 1: Long Papers*, pp. 1970–1979, 2018.

[5] X. Li, H. Xie, R. Wang, Y. Cai, J. Cao, F. Wang, H. Min, and X. Deng, "Empirical analysis: stock market prediction via extreme learning machine," *Neural Comput. Appl.*, vol. 27, no. 1, pp. 67–78, 2016.

[6] Y. Kara, M. A. Boyacioglu, and Ö. K. Baykan, "Predicting direction of stock price index movement using artificial neural networks and support vector machines: The sample of the istanbul stock exchange," *Expert Syst. Appl.*, vol. 38, no. 5, pp. 5311–5319, 2011.

[7] C. Tsai and Y. Hsiao, "Combining multiple feature selection methods for stock prediction: Union, intersection, and multi-intersection approaches," *Decis. Support Syst.*, vol. 50, no. 1, pp. 258–269, 2010.

[8] Y. Qin, D. Song, H. Chen, W. Cheng, G. Jiang, and G. W. Cottrell, “A dual-stage attention-based recurrent neural network for time series prediction,” in *Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence, IJCAI 2017, Melbourne, Australia, August 19-25, 2017* (C. Sierra, ed.), pp. 2627–2633, ijcai.org, 2017.

[9] Q. Liu, X. Cheng, S. Su, and S. Zhu, “Hierarchical complementary attention network for predicting stock price movements with news,” in *Proceedings of the 27th ACM International Conference on Information and Knowledge Management, CIKM 2018, Torino, Italy, October 22-26, 2018* (A. Cuzzocrea, J. Allan, N. W. Paton, D. Srivastava, R. Agrawal, A. Z. Broder, M. J. Zaki, K. S. Candan, A. Labrinidis, A. Schuster, and H. Wang, eds.), pp. 1603–1606, ACM, 2018.

[10] X. Ding, Y. Zhang, T. Liu, and J. Duan, “Deep learning for event-driven stock prediction,” in *Proceedings of the Twenty-Fourth International Joint Conference on Artificial Intelligence, IJCAI 2015, Buenos Aires, Argentina, July 25-31, 2015* (Q. Yang and M. J. Wooldridge, eds.), pp. 2327–2333, AAAI Press, 2015.

[11] H. Wu, W. Zhang, W. Shen, and J. Wang, “Hybrid deep sequential modeling for social text-driven stock prediction,” in *Proceedings of the 27th ACM International Conference on Information and Knowledge Management, CIKM 2018, Torino, Italy, October 22-26, 2018* (A. Cuzzocrea, J. Allan, N. W. Paton, D. Srivastava, R. Agrawal, A. Z. Broder, M. J. Zaki, K. S. Candan, A. Labrinidis, A. Schuster, and H. Wang, eds.), pp. 1627–1630, ACM, 2018.

[12] D. Ding, M. Zhang, X. Pan, M. Yang, and X. He, “Modeling extreme events in time series prediction,” in *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, KDD 2019, Anchorage, AK, USA, August 4-8, 2019* (A. Teredesai, V. Kumar, Y. Li, R. Rosales, E. Terzi, and G. Karypis, eds.), pp. 1114–1122, ACM, 2019.

[13] L. Zhang, C. C. Aggarwal, and G. Qi, “Stock price prediction via discovering multi-frequency trading patterns,” in *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Halifax, NS, Canada, August 13 - 17, 2017*, pp. 2141–2149, ACM, 2017.

[14] J. Yoo and U. Kang, "Attention-based autoregression for accurate and efficient multivariate time series forecasting," in *Proceedings of the 2021 SIAM International Conference on Data Mining (SDM)*, pp. 531–539, 2021.

[15] K. Cho, B. van Merrienboer, Ç. Gülçehre, D. Bahdanau, F. Bougares, H. Schwenk, and Y. Bengio, "Learning phrase representations using RNN encoder-decoder for statistical machine translation," in *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing, EMNLP 2014, October 25-29, 2014, Doha, Qatar, A meeting of SIGDAT, a Special Interest Group of the ACL* (A. Moschitti, B. Pang, and W. Daelemans, eds.), pp. 1724–1734, ACL, 2014.

[16] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[17] L. Rosasco, E. D. Vito, A. Caponnetto, M. Piana, and A. Verri, "Are loss functions all the same?," *Neural Comput.*, vol. 16, no. 5, pp. 1063–107, 2004.

[18] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings* (Y. Bengio and Y. LeCun, eds.), 2015.

[19] D. M. Q. Nelson, A. C. M. Pereira, and R. A. de Oliveira, "Stock market's price movement prediction with LSTM neural networks," in *2017 International Joint Conference on Neural Networks, IJCNN 2017, Anchorage, AK, USA, May 14-19, 2017*, pp. 1419–1426, IEEE, 2017.

[20] T. Mikolov, K. Chen, G. Corrado, and J. Dean, "Efficient estimation of word representations in vector space," in *1st International Conference on Learning Representations, ICLR 2013, Scottsdale, Arizona, USA, May 2-4, 2013, Workshop Track Proceedings* (Y. Bengio and Y. LeCun, eds.), 2013.

[21] Q. V. Le and T. Mikolov, "Distributed representations of sentences and documents," in *Proceedings of the 31th International Conference on Machine Learning, ICML 2014, Beijing, China, 21-26 June 2014*, vol. 32 of *JMLR Workshop and Conference Proceedings*, pp. 1188–1196, JMLR.org, 2014.

[22] B. W. Matthews, "Comparison of the predicted and observed secondary structure of t4 phage lysozyme," *Biochimica et Biophysica Acta (BBA)-Protein Structure*, vol. 405, no. 2, pp. 442–451, 1975.

요 약

과거 주가와 예측할 주식을 언급하는 희소한 트윗이 주어졌을 때 주가 변동을 어떻게 정확하게 예측할 수 있을까? 많은 시장 분석가들은 예측을 위해 많은 양의 정보를 사용하려고 노력합니다. 하지만 예측을 위해 더 많은 양의 데이터를 사용할수록 더 많은 노이즈에 직면합니다. 따라서 기존 방법은 과거 주식 가격만 사용하거나 뉴스 기사 혹은 대상 주식을 언급하는 트윗과 같은 소량의 정제된 데이터를 사용합니다. 그러나 기존 방법들은 다음과 같은 한계가 있습니다: 1) 과거 주식 가격만 사용하면 정보가 부족하여 성능이 저하되고, 2) 뉴스 기사는 주가 변동을 예측하는데 소셜 미디어에 비해 적시성이 부족하며, 3) 트윗을 사용하는 이전 방법들은 트윗이 언급하지 않은 주식들을 처리하지 못합니다.

본 논문에서는 목표 주식을 언급하는 트윗 없이도 작동하는 정확한 주가 변동 예측 방법인 GLT (Stock Price Movement Prediction using **G**lobal and **L**ocal **T**rends of Tweets)를 제안합니다. GLT는 자가 감독 방식을 활용하여 주식 및 트윗 임베딩을 사전 학습합니다. 그런 다음 GLT는 각각 글로벌 여론과 목표 주식과 관련된 트렌드를 나타내는 글로벌 및 로컬 트윗 트렌드를 생성합니다. 이러한 추세 벡터들은 주가 변동을 정확하게 예측하는데 기여합니다. 실험 결과에 따르면 GLT는 주가 변동 예측에서 최고 수준의 정확도를 제공합니다.

주요어 : 주가 예측, 자가 지도 학습, 순환 신경망, 시계열 분류, 소셜 미디어 데이터
학번 : 2019-26047