A New Model for Stock Price Movements Prediction Using Deep Neural Network

Huy D. Huynh*
University of IT - VNU HCMC
Ho Chi Minh City, Vietnam
huyhd@uit.edu.vn

L.Minh Dang†
University of IT - VNU HCMC
Ho Chi Minh City, Vietnam
danglienminh93@gmail.com

Duc Duong‡
University of IT - VNU HCMC
Ho Chi Minh City, Vietnam
ducdm@uit.edu.vn

ABSTRACT

In this paper, we introduce a new prediction model depend on Bidirectional Gated Recurrent Unit (BGRU). Our predictive model relies on both online financial news and historical stock prices data to predict the stock movements in the future. Experimental results show that our model accuracy achieves nearly 60% in S&P 500 index prediction whereas the individual stock prediction is over 65%.

CCS CONCEPTS

• **Information systems** → *Expert systems*;

KEYWORDS

Stock market prediction, GRU, BGRU, LSTM

ACM Reference Format:

Huy D. Huynh, L.Minh Dang, and Duc Duong. 2017. A New Model for Stock Price Movements Prediction Using Deep Neural Network. In *SoICT '17: Eighth International Symposium on Information and Communication Technology, December 7–8, 2017, Nha Trang City, Viet Nam*. ACM, New York, NY, USA, 6 pages. https://doi.org/10.1145/3155133.3155202

1 INTRODUCTION

Stock market prediction is always a challenging task because it is highly dynamic. Several methods have been deployed to forecast the future direction of the stock market. One of the most significant factors impacts human's reaction in the stock market derived from news articles. Recently, the number of online news have risen dramatically. As a result, investors find it difficult in updating the latest information. So automated systems should be developed and they will hopefully be useful for investors. For examples, if the direction of selected stock is predicted to be "up" in the next 24 hours, investors can buy stock or make a good trading action.

*Huy insisted his name be first.
†This author is the one who did all the really hard work.
‡This author is the one who did all the really hard work.

SoICT '17, December 7–8,2017, Nha Trang City, Viet Nam

ACM ISBN 978-1-4503-5328-1/17/12...$15.00
https://doi.org/10.1145/3155133.3155202

Sequence classification is a predictive modeling problem where you have some sequence of inputs over space or time and the task is to predict a category for the sequence. In this paper, we deal with the problem of classification each article into a set of predefined sentiment categories (i.e. stock price "up" or "down"). Facing with the variety of available data sources is difficult, the sequences can vary in length, be comprised of a very large vocabulary of input symbols and may require the model to learn the long term context or dependencies between symbols in the input sequence. The collection of data is huge, while the importance of the data and the potentially complex non-linear interactions in the data are not specified by the financial economic theory. In fact, this results in an excessive of predictive models, many with little theoretical justification and subject to over-fitting and poor predictive out-of-sample performance. In recent years, deep neural networks (DNNs) have achieved numerous successes in various domains such as speech recognition, computer vision and natural language processing. Thus, researchers have already applied some DNN models to train the features extracted from the news articles and historical stock prices such as in [9] and [20]. Previous work using DNNs have proven to be effective on forecasting the stock prices. However, these features do not capture the structural relation. For example, giving the event of "Microsoft sues Apple for violating its intellectual patent". If we use only terms "Microsoft", "sues", "Apple", it is difficult to accurately predict the price movement of Microsoft and Apple stocks because the system cannot differentiate between the accuser and the defender.

During the past few years, deep neural networks (DNNs) have achieved enormous successes in many data prediction models - speech recognition, computer vision, natural language processing to name but a few. In this paper, we are going to apply deep learning methods to financial data to predict the price movements.

For decades, the stock market prediction has merely focused on historical data. Researchers applied many algorithms such as Moving average, Multiple Kernel Learning, Support Vector Machines and other techniques to analyze the stock market behavior. Although they had a promising result, these approaches are difficult to predict the market accurately because the researchers tried to predict the stock market from historical prices.

To address the above limitation, we introduce deep learning hierarchical decision Bidirectional Gated Recurrent Unit (BGRU) models for financial prediction and classification problems. The deep learning predictor has a number of advantages over traditional predictors, which include

- Input data can be extended more related items of possible relevance to the stock prediction,
- Non-linearity and complex interactions among input data are considered, which can help growing in-sample fit versus traditional models,
- More easily prevent over-fitting.

The experiments are conducted on a large-scale financial news dataset collected from Reuters and Bloomberg websites. Our experiments examine the influence of the news on predicting the polarity of the stock change in each time interval. In addition, our model had an improvement in the accuracy on both the Standard&Poor's 500 (S&P 500) stock index prediction and the individual stock prediction.

This article is arranged as follows. Section 2 discusses previous work. Section 3 explains the details of our approach and proposed method. Section 4 shows the experimental results and analysis. Section 5 delivers a brief discussion about conclusions in this research.

2 RELATED WORK

So far, there have been two different approaches in classifying documents. The first approach is to assign a class to the article manually by experts opinions in the content of the article. Although the successful rate is a bit higher by using this method, processing a large number of articles will be relatively hard by using only humans efforts. In the other hand, the second approach assigns labels to articles automatically according to their effects on the stock prices. The latter is less accurate than the former because sometimes, the stock prices change does not indicate the actual label of the article. For example, although the article implies that the stock price will increase, global finance crises may cause a drop in the stock price.

Machine learning is an active research area that attracts increasing interest. A large number of papers have shown that supervised machine learning models such as Genetic Algorithms [16], Random Forests [17],[25], Support Vector Machine [15], [8], [10] and Artificial Neural Network [11], [21] were effective in predicting the trend of the stock prices based on time-series price data, owing to their ability to handle non-linear systems. However, most of them still not had satisfactory results with a high accuracy and stable performance on stock prediction [1]. Traditionally, neural networks have mainly been used in time series data for the forecasting purpose, such as in [4, 24]. Due to the shortage in training dataset and computing power back then, shallow neural networks were implemented for various types of features such as historical prices, trading volumes in order to predict the future stock yields and market returns.

In recent years, we have seen a significant increase in the adoption of the data extracted from the websites and social networks in an attempt to create better stock predictive models. So far, there have been two different approaches in classifying documents. The first approach was to assign a class to the article manually by the experts opinions about the content of the article. Although the successful rate was higher by using this method, processing a large number of articles will be extremely hard by using only humans efforts. On the other hand, the second approach assigned labels to the articles automatically according to their effects on the stock prices. The latter was less accurate than the former because sometimes the stock prices movements did not indicate the actual label of the article. For example, although the article implies that the stock price will increase, global finance crises can cause a drop in the stock prices. Many new methods have been proposed to explore additional information (mainly online text data) for the stock forecasting such as financial news [9, 26], twitters sentiments [23], micro blogs [3]. [26] proposed semantic frame parser to convert from sentences to scenarios in order to detect the (positive or negative) roles of specific companies while support vector machines with the tree kernels was used as the predictive models. Moreover, [9] proposed the use of various lexical and syntactic constraints to extract event features for the stock prediction, they investigated both linear classifiers and deep neural networks as their predictive models. Most recently, [20] used DNNs in predicting the future stock movements based on the extracted features. Therefore, deep learning fits perfectly to the the challenge of stock market prediction, and provides a new valuable approach to this field.

3 PROPOSED METHOD

In this section, we wil provide a brief description of Recurrent neural network (RNN), Long-short-term-memory model (LSTM), Gated Recurrent Unit (GRU) and explains our proposed BGRU model.

3.1 Preliminary

RNN is in the family of neural network which operate on sequential data. They take sequence of vectors $(x_1, x_2, ..., x_n)$ as input and extract another sequence $(h_1, h_2, ..., h_n)$ that represent some information about the sequence at every step in the input. Especially RNN handles the variable-length sequence by having a recurrent hidden state whose activation at each time is dependent on that of the previous time.

Traditionally, the update of the recurrent hidden state h_t is implemented as

$$h_t = g(Wx_t + Uh_{t-1} + b) \tag{1}$$

Where g is a smooth, bounded function such as a logistic function or a hyperbolic tangent function. At each time t step, the hidden state h_t is a function of the input vector x_t that the network receives at time t with its previous hidden state h_{t-1} and bias b.

Though RNN have been proved successful on speech recognition and text generation tasks [18], a problem with RNN has been observed by [13], they stated that it was difficult to train RNN model to capture the long-term dependencies because the gradients tend to either vanish or explode.

LSTM which was proposed by [14] is a particular form of recurrent network which provides a solution by incorporating memory units. This allows the network to learn when to forget the previous hidden states and when to update the hidden states with new information. Models with hidden units with varying connections within the memory unit have been proposed in the literature with great empirical success.

The LSTM unit at each time t step is a collection of vectors in R^{d}: an input gate i_{t}, a forget gate f_{t}, an output gate o_{t}, a memory cell c_{t}, and a hidden state h_{t}. The LSTM transition equations are as following:

$$i_{\mathrm{t}} = \sigma(W^{(\mathrm{i})}x_{\mathrm{t}} + U^{(\mathrm{i})}h_{\mathrm{t\text{-}1}} + b^{(\mathrm{i})}) \tag{2}$$

$$f_{\mathrm{t}} = \sigma(W^{(\mathrm{f})}x_{\mathrm{t}} + U^{(\mathrm{f})}h_{\mathrm{t\text{-}1}} + b^{(\mathrm{f})}) \tag{3}$$

$$o_{\mathrm{t}} = \sigma(W^{(\mathrm{o})}x_{\mathrm{t}} + U^{(\mathrm{o})}h_{\mathrm{t\text{-}1}} + b^{(\mathrm{o})}) \tag{4}$$

$$u_{\mathrm{t}} = \tanh(W^{(\mathrm{u})}x_{\mathrm{t}} + U^{(\mathrm{u})}h_{\mathrm{t\text{-}1}} + b^{(\mathrm{u})}) \tag{5}$$

$$c_{\mathrm{t}} = i_{\mathrm{t}} \odot u_{\mathrm{t}} + f_{\mathrm{t}} \odot c_{\mathrm{t\text{-}1}} \tag{6}$$

$$h_{\mathrm{t}} = o_{\mathrm{t}} \odot \tanh(c_{\mathrm{t}}) \tag{7}$$

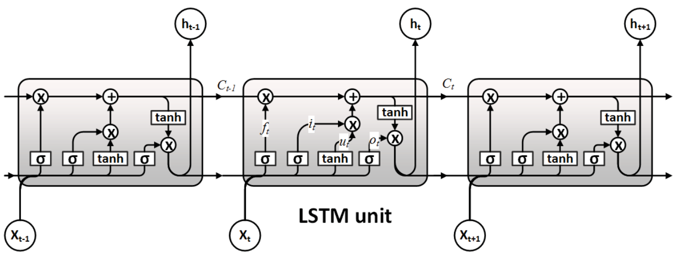

Figure 1: A LSTM network.

Where x_{t} is the input at the current time step, σ denotes the logistic sigmoid function while $\odot$ denotes element wise multiplication. Unlike to the traditional recurrent unit which overwrites its content at each time step, the LSTM unit is able to decide whether to keep the existing memory via the introduced gates. Intuitively, the forget gate controls the extent to which the previous memory cell is forgotten, the input gate decides what new information is going to update, and the output gate controls the exposure of the internal memory state. Figure 1 illustrates the LSTM network [1].

A slightly more dramatic variation on the LSTM is the GRU proposed by [5]. It combines the forget and the input gates into a single "update gate". It also merges the cell state and the hidden state and makes some other changes. GRU model is simpler than the standard LSTM models but it performs similarly and is often faster [7]. The internal mechanics of the GRU are defined as:

$$z_{\mathrm{t}} = \sigma(W^{(\mathrm{z})}x_{\mathrm{t}} + U^{(\mathrm{z})}h_{\mathrm{t\text{-}1}} + b^{(\mathrm{z})}) \tag{8}$$

$$r_{\mathrm{t}} = \sigma(W^{(\mathrm{r})}x_{\mathrm{t}} + U^{(\mathrm{r})}h_{\mathrm{t\text{-}1}} + b^{(\mathrm{r})}) \tag{9}$$

$$\tilde{h}_{\mathrm{t}} = \tanh(Wx_{\mathrm{t}} + r_{\mathrm{t}} \odot Uh_{\mathrm{t\text{-}1}} + b^{(\mathrm{h})}) \tag{10}$$

$$h_{\mathrm{t}} = z_{\mathrm{t}} \odot h_{\mathrm{t\text{-}1}} + (1 - z_{\mathrm{t}}) \odot \tilde{h}_{\mathrm{t}} \tag{11}$$

[1]source: colah.github.io/posts/2015-08-Understanding-LSTMs

Where an update gate z_{t}, r_{t} is a reset gate and the candidate activation $\tilde{h}_{\mathrm{t}}$. Figure 2 illustrates the GRU network.

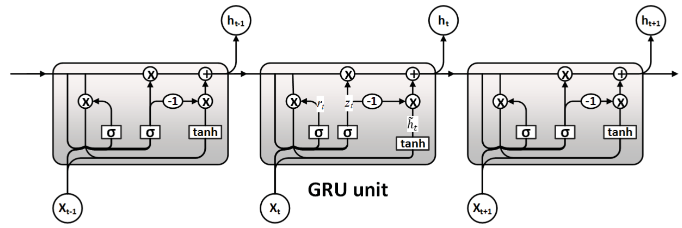

Figure 2: A GRU network.

3.2 Proposed BGRU model

In this part, we introduce the BGRU networks. The idea behind BGRU is the idea of bidirectional recurrent neural network in [2, 22]. The BGRU presents each training sequence forwards and backwards to two separate recurrent nets, both of which are connected to the same output layer. The equations for this network is as follows:

Forward:

$$\vec{z}_{\mathrm{t}} = \sigma(\vec{W}^{(\mathrm{z})}x_{\mathrm{t}} + \vec{U}^{(\mathrm{z})}h_{\mathrm{t\text{-}1}} + \vec{b}^{(\mathrm{z})}) \tag{12}$$

$$\vec{r}_{\mathrm{t}} = \sigma(\vec{W}^{(\mathrm{r})}x_{\mathrm{t}} + \vec{U}^{(\mathrm{r})}h_{\mathrm{t\text{-}1}} + \vec{b}^{(\mathrm{r})}) \tag{13}$$

$$\tilde{\vec{h}}_{\mathrm{t}} = \tanh(\vec{W}x_{\mathrm{t}} + r_{\mathrm{t}} \odot \vec{U}h_{\mathrm{t\text{-}1}} + \vec{b}^{(\mathrm{h})}) \tag{14}$$

$$\vec{h}_{\mathrm{t}} = \vec{z}_{\mathrm{t}} \odot h_{\mathrm{t\text{-}1}} + (1 - \vec{z}_{\mathrm{t}}) \odot \tilde{\vec{h}}_{\mathrm{t}} \tag{15}$$

Backward:

$$\overleftarrow{z}_{\mathrm{t}} = \sigma(\overleftarrow{W}^{(\mathrm{z})}x_{\mathrm{t}} + \overleftarrow{U}^{(\mathrm{z})}h_{\mathrm{t\text{-}1}} + \overleftarrow{b}^{(\mathrm{z})}) \tag{16}$$

$$\overleftarrow{r}_{\mathrm{t}} = \sigma(\overleftarrow{W}^{(\mathrm{r})}x_{\mathrm{t}} + \overleftarrow{U}^{(\mathrm{r})}h_{\mathrm{t\text{-}1}} + \overleftarrow{b}^{(\mathrm{r})}) \tag{17}$$

$$\tilde{\overleftarrow{h}}_{\mathrm{t}} = \tanh(\overleftarrow{W}x_{\mathrm{t}} + r_{\mathrm{t}} \odot \overleftarrow{U}h_{\mathrm{t\text{-}1}} + \overleftarrow{b}^{(\mathrm{h})}) \tag{18}$$

$$\overleftarrow{h}_{\mathrm{t}} = \overleftarrow{z}_{\mathrm{t}} \odot h_{\mathrm{t\text{-}1}} + (1 - \overleftarrow{z}_{\mathrm{t}}) \odot \tilde{\overleftarrow{h}}_{\mathrm{t}} \tag{19}$$

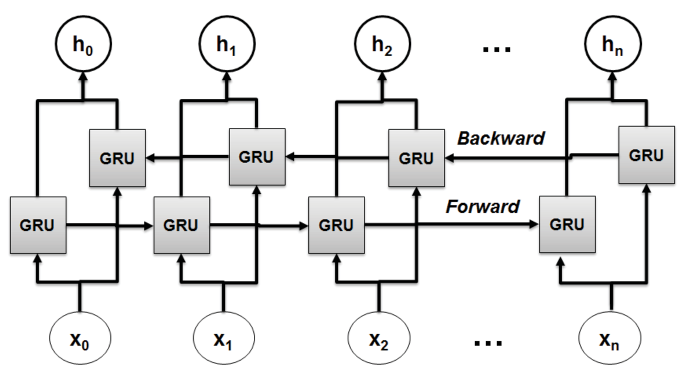

Figure 3: A BGRU network.

The representation of a word: $h_{\mathrm{t}} = [\overleftarrow{h_{\mathrm{t}}}, \overrightarrow{h_{\mathrm{t}}}]$ For a given sequence $(x_1, x_2, \ldots, x_n)$ containing n words, every word at time t is represented as a d-dimensional vector and a forwards

GRU computes a representation $\tilde{h}_t$ of the left context of the sentence. We add more backwards GRU to compute a representation of the right context $\tilde{h}_t$. The representation of a word using this model is obtained by concatenating its left and right context representations, $h_t = [\overleftarrow{h_t}, \overrightarrow{h_t}]$. These representations effectively include a representation of a word in context, which is useful for numerous tagging applications. This could update more useful information for a specific time frame. Figure 3 shows the BGRU models.

3.3 Word embedding

In this paper, we mapped each article(news) into a real vector domain, a popular technique when working with text called word embedding. We used the recent word embedding methods [19] to choose the effective features from the online financial news dataset. This is a technique where words are encoded as real-valued vectors in a high dimensional space, the similarity between words in terms of meaning translates to closeness in the vector space. The results shown that the features derived from financial news were effective and they significantly improved the prediction accuracy compared to the system that only depends on the historical prices.

3.4 Dropout training

We applied a dropout mask [12] in our models. The key idea was to randomly drop the units (along with their connections) from the neural network during the training. This helped to prevent the units from co-adapting too much. During the training, dropout sampled from an exponential number of different "thinned" networks. We observed a significant improvement in our model's performance by reducing the over-fitting.

4 EXPERIMENTS

Our experiments examined the influence of the news on predicting the polarity of stock change for each time interval, and then compared it with the two state-of-the-art financial-new-based stock market prediction systems. We used the Keras 1.1.0 [6] deep learning library in Python to implement this experimentation. The first layer is the Embedded layer that used 32 length vectors to represent each word. We were also limit the total number of words that we are interested in modeling to the 20,000 most frequent words, and remove the rest. The sequence length (number of words) in each review varies, so we constrained each review to be 2,000 words, truncating long reviews and pad the shorter reviews with zero values.The next layer is the LSTM layer or BGRU layer with 128 memory units (smart neurons). Keras provides this capability with parameters on the LSTM layer and GRU layer the dropout-W for configuring 20 percent the input dropout and dropout-U for configuring 20 percen the recurrent dropout. Finally, because this is a classification problem we used a Softmax output layer with a single neuron and a sigmoid activation function to make 0 or 1 predictions for the two classes (up and down) in the problem. The illustration our prediction model architecture was shown in Figure 4.

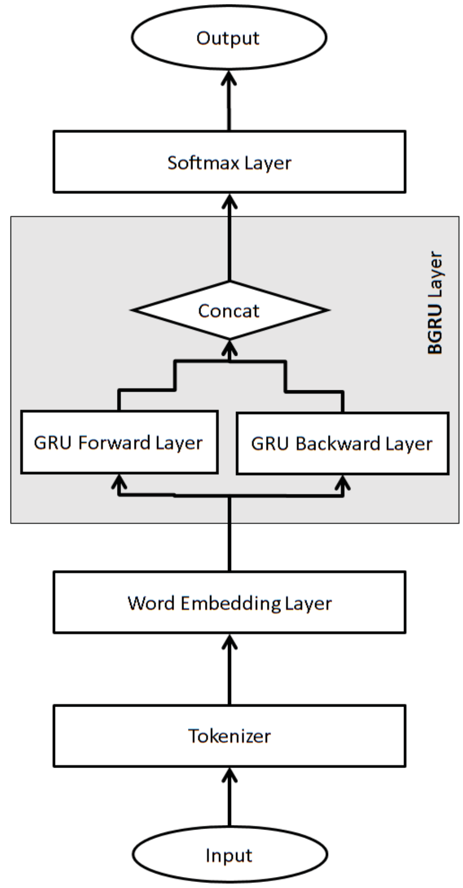

Figure 4: Illustration on the architecture our model.

4.1 Data description

Our experiment used financial news from Reuters and Bloomberg between October 2006 and November 2013. Reuters and Bloomberg dataset contained approximately 106,521 news and 447,145 news respectively. We also used the public price data from Yahoo Finance from 2006 to 2013 which match the time period of the financial news to conduct our experiment on forecasting (S&P500) index and its individual stocks. We split the dataset into three sections which are similar to [20], the news between 2006-10-01 and 2012-12-31 for training, had news from 2013-01-01 to 2013-06-15 for validating and testing contained news from 16-06-2013 to 31-12-2013.

4.2 Evaluate the impact of time

To assess the impact of the financial news on the price of the stock over time, we examine our prediction method on many time intervals (i.e. 1 day, 2 days, 5 days, 7 days and 10 days) on the Reuters News dataset. In one-day period case, it means that the news affects stock prices within 24 hours. Similarly, for the remaining periods. We compared the stock price open and close to labeled "up" or "down"

Table 1: Final results on the test dataset

Author	Accuracy
Ding et al. [2014]	55.21%
Peng and Hui Jiang [2016]	56.87%
LSTM	58.64%
GRU	58.59%
BGRU	**59.98%**

for each article. Experimental results are shown in Figure 5, we applied LSTM and BGRU model. We got the highest accuracy in the first 24 hours and the accuracy decrease over time. It also demonstrated the impact of financial news and the rapid reflect of the stock market. Through all the experiments above, it is clear that news quickly impacts the stock price within 24 hours. In fact, we are able predict stock movements over a period of more than a day. However, the influence of the news is being reduced over time.

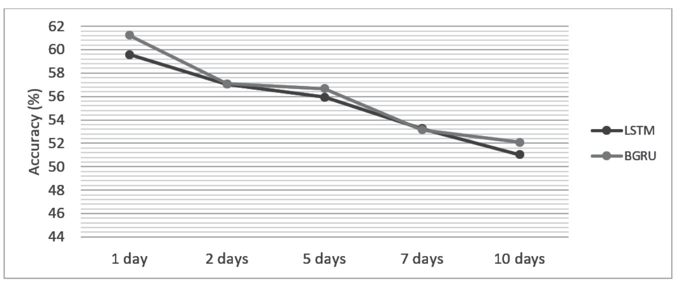

Figure 5: Result of different time intervals.

4.3 S&P500 stock prediction

In the first part of our experiments, we compared the result of BGRU Models, LSTM Models with the two state-of-the-art financial-news-based stock market prediction systems. [9] reports a system that uses structured event tuples (O_i, P, T), where $O_i \subseteq O$ is a set of objects, P is a relation over the objects and T is a time interval. They propose the representation that further structures the event to represent news documents, and investigates the complex hidden relationships between events and stock price movements by using a standard feed-forward neural network. [20] used word embedding method to select features from news corpora, and employ DNNs to predict the future stock movements based on the extracted features. Following [9] and [20], the standard measure of accuracy (Acc) is used to evaluate S&P 500 index prediction and individual stock prediction. Experimental results are shown in Table 1. We find that our models (i.e. LSTM, GRU, BGRU) achieve consistently better performance compared to the baseline methods. The BGRU method have obtained the best performance with an accuracy of 59.98%.

Table 2: Training and testing dataset for the individual stock

Company	Training set	Testing set
Google Inc.	2,252	1,124
Wal-Mart	1,484	741
Boeing Company	2,080	1,039

4.4 Individual stock prediction

We used the three companies selected by [9] in different financial sectors for evaluating the effectiveness of our approach on the aspect of individual stock prediction. We chose Google Inc. in Information Technology, Wal-Mart Stores in Consumer Staples and Boeing Company in Industrial (classified by the Global Industry Classification Standard). We extracted all the news, regard to the three mentioned companies in Reuters News. Detailed statistics about the number of news for training and test set are shown in Table 2. We compared our BGRU model with the standard LSTM and GRU. The results are shown in Figure 6. Our model achieved the best performance compared to the standard LSTM model and GRU model.

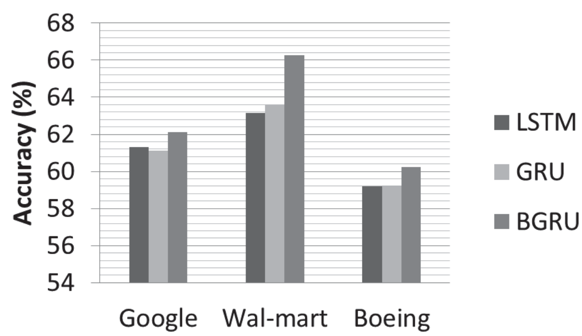

Figure 6: Result of individual stock prediction.

5 CONCLUSION

The DNN is a powerful framework for the large dataset and optimize the performance. In this paper, we apply the extended model of RNN such as LSTM, GRU and introduce a new BGRU model for the stock price movement prediction and classification. Experimental results have shown that our proposed method was simple but very effective, which could significantly improve the stock prediction accuracy on a standard financial database over the other systems which only used the historical price information.

REFERENCES

[1] JG Agrawal, VS Chourasia, and AK Mittra. 2013. State-of-the-art in stock prediction techniques. *International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering* 2, 4 (2013), 1360–1366.

[2] Pierre Baldi, Søren Brunak, Paolo Frasconi, Giovanni Soda, and Gianluca Pollastri. 1999. Exploiting the past and the future in protein secondary structure prediction. *Bioinformatics* 15, 11 (1999), 937–946.

[3] Roy Bar-Haim, Elad Dinur, Ronen Feldman, Moshe Fresko, and Guy Goldstein. 2011. Identifying and following expert investors in stock microblogs. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, 1310–1319.

[4] Man-Chung Chan, Chi-Cheong Wong, and Chi-Chung Lam. 2000. Financial time series forecasting by neural network using conjugate gradient learning algorithm and multiple linear regression weight initialization. In *Computing in Economics and Finance*, Vol. 61.

[5] Kyunghyun Cho, Bart Van Merriënboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. 2014. Learning phrase representations using RNN encoder-decoder for statistical machine translation. *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)* (2014).

[6] Francois Chollet. 2015. Keras deeplearning library www.keras.io. (2015).

[7] Junyoung Chung, Caglar Gulcehre, KyungHyun Cho, and Yoshua Bengio. 2014. Empirical evaluation of gated recurrent neural networks on sequence modeling. *NIPS Deep Learning Workshop* (2014).

[8] Minh Dang and Duc Duong. 2016. Improvement methods for stock market prediction using financial news articles. In *Information and Computer Science (NICS), 2016 3rd National Foundation for Science and Technology Development Conference on*. IEEE, 125–129.

[9] Xiao Ding, Yue Zhang, Ting Liu, and Junwen Duan. 2014. Using Structured Events to Predict Stock Price Movement: An Empirical Investigation.. In *EMNLP*. 1415–1425.

[10] Duc Duong, Toan Nguyen, and Minh Dang. 2016. Stock Market Prediction using Financial News Articles on Ho Chi Minh Stock Exchange. In *Proceedings of the 10th International Conference on Ubiquitous Information Management and Communication*. ACM, 71.

[11] Erkam Guresen, Gulgun Kayakutlu, and Tugrul U Daim. 2011. Using artificial neural network models in stock market index prediction. *Expert Systems with Applications* 38, 8 (2011), 10389–10397.

[12] Geoffrey E Hinton, Nitish Srivastava, Alex Krizhevsky, Ilya Sutskever, and Ruslan R Salakhutdinov. 2012. Improving neural networks by preventing co-adaptation of feature detectors. *arXiv preprint arXiv:1207.0580* (2012).

[13] Sepp Hochreiter. 1998. The vanishing gradient problem during learning recurrent neural nets and problem solutions. *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems* 6, 02 (1998), 107–116.

[14] Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural computation* 9, 8 (1997), 1735–1780.

[15] Yakup Kara, Melek Acar Boyacioglu, and Ömer Kaan Baykan. 2011. Predicting direction of stock price index movement using artificial neural networks and support vector machines: The sample of the Istanbul Stock Exchange. *Expert systems with Applications* 38, 5 (2011), 5311–5319.

[16] Kyoung-jae Kim and Ingoo Han. 2000. Genetic algorithms approach to feature discretization in artificial neural networks for the prediction of stock price index. *Expert systems with Applications* 19, 2 (2000), 125–132.

[17] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. 2012. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*. 1097–1105.

[18] Brenden M Lake, Ruslan Salakhutdinov, and Joshua B Tenenbaum. 2015. Human-level concept learning through probabilistic program induction. *Science* 350, 6266 (2015), 1332–1338.

[19] Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. 2013. Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems*. 3111–3119.

[20] Yangtuo Peng and Hui Jiang. 2016. Leverage Financial News to Predict Stock Price Movements Using Word Embeddings and Deep Neural Networks. *Proceedings of NAACL-HLT 2016, San Diego, California* (2016).

[21] Karsten Schierholt and Cihan H Dagli. 1996. Stock market prediction using different neural network classification architectures. In *Computational Intelligence for Financial Engineering, 1996., Proceedings of the IEEE/IAFE 1996 Conference on*. IEEE, 72–78.

[22] Mike Schuster and Kuldip K Paliwal. 1997. Bidirectional recurrent neural networks. *IEEE Transactions on Signal Processing* 45, 11 (1997), 2673–2681.

[23] Jianfeng Si, Arjun Mukherjee, Bing Liu, Sinno Jialin Pan, Qing Li, and Huayi Li. 2014. Exploiting Social Relations and Sentiment for Stock Prediction.. In *EMNLP*, Vol. 14. 1139–1145.

[24] Andrew Skabar and Ian Cloete. 2002. Neural networks, financial trading and the efficient markets hypothesis. *Australian Computer Science Communications* 24, 1 (2002), 241–249.

[25] Martin Wiesmeier, Frauke Barthold, Benjamin Blank, and Ingrid Kögel-Knabner. 2011. Digital mapping of soil organic matter stocks using Random Forest modeling in a semi-arid steppe ecosystem. *Plant and soil* 340, 1-2 (2011), 7–24.

[26] Boyi Xie, Rebecca J Passonneau, Leon Wu, and Germán G Creamer. 2013. Semantic frames to predict stock price movement. In *Proceedings of the 51st annual meeting of the association for computational linguistics*. 873–883.