Proceedings of the 3rd International Conference on Statistics: Theory and Applications (ICSTA'21)
~~Prague, Czech Republic~~ Virtual Conference – August, 2021
Paper No. XXX (The number assigned by the OpenConf System)
DOI: TBA

Combining Time Series and Sentiment Analysis for Stock Market Forecasting

Hsiao-Chuan Chou[1], Kandethody M. Ramachandran[2]
[1]Department of Mathematics and Statistics, College of Arts and Sciences, University of South Florida
4202 E Fowler Ave, Tampa, USA
hsiaochuan@usf.edu
[2] Department of Mathematics and Statistics, College of Arts and Sciences, University of South Florida
4202 E Fowler Ave, Tampa, USA
ram@usf.edu

Abstract – Objective of this research is to build a model to predict stock price using sentimental information from news headlines and historical prices, and the model is able to not only conclude better results but also minimize the difference between predicted values and actual values. News headlines show impact on stock price. Unlike previous approaches where the textual information were usually calculated into sentiment score, we apply various approaches to extract information from news headlines. On the other hand, price data through time series are also useful to predict stock prices. Hence, improvement is made with combination of sentiment analysis of news headlines and time series analysis and combination of news headline and historical prices is able to complement nonavailability of sentiment lexicon and lack of news. Compared to time series models and word embedding models, our combined model shows smaller or similarly error measures MAPE, MAE, and RMSE with time series models, and reduces lag in graphs.

Keywords: Long short-term memory, word embedding, time series, neural networks

1. Introduction

Financial asset forecasting is a charming and challenging problem at all times. There are many factors affecting trend direction. News and historical prices are commonly considered as two of those important factors affecting trend of financial asset. Hence, the idea of this work is to apply proper analytical approaches to them to reach better predictions of stock prices.

Sentiment is a personally subjective attitude toward a subject, which means an opinion to depict emotion. Sentiment analysis also known as text mining is capable to extract subjectivity from reviews, customer feedbacks, or texts. Sentiment analysis is being applied in many fields, especial business fields, to gather valuable information. Adjustment of business strategy, for instance, can be made from sentiments of customers' reviews, and that's why sentiment analysis is quite popular. Sentiment analysis can be performed on not only reviews of customers but also tweets, news and any personal text, and many researches related to stock market forecasting with sentiments have been done. With combination of moving average of financial indicator and sentence level sentiment score of Really Simple Syndication (RSS) news, trend of stock price for specific company is performed [1].

Since historical price of financial asset is time series data, many approaches for time series are applied to predict stock price. Artificial neural networks (ANN) as widely used forecasting model are also applied to solve non-linear problems such as stock price prediction [2]. Besides, recurrent neural networks (RNNs) as one of the popular neural network family constructs a model for trend of financial assets [3]. Volatility of financial asset market leads to a difficult situation to predict trend of the market. Combination of several machine learning approaches are being used in prediction [4].

In this paper, we compare pure time series models, word embedding models, and the new combined models for stock price data and news headline data in three error measures to find the most suitable model. The rest of the paper is as follows. Related works are given in Section 2 and data explanation would be introduced in Section 3. Data processing and evaluation are in Section 4 and 5 respectively. Models and procedures to build model are discussed in Section 6. Results would be presented in Section 7, and we finally would summarize and conclude this paper in Section 8.

2. Related Works

Over the past few decades, many researchers have used machine learning approaches to analyze financial information including financial time series data and textual data [5]. For time series data, there are various approaches such as artificial neural network (ANN), autoregressive integrated moving average (ARIMA), k nearest neighbor (KNN), recurrent neural network (RNN), support vector regression (SVR), and so on. ANN has been used for time series forecasting for over 20 years for its ability to tackle nonlinear patterns, but it is sensitive to parameter selection [6]. ARIMA which is introduced in 1970 has also been being used to analyze financial data due to its high ability for linear time series [2]. It usually combines machine learning approaches with better ability to handle nonlinear time series such as ANN [7]. Recurrent neural networks are well-suited to solve both classification and regression problems with series data. Long short-term memory (LSTM) is the most fashionable member of it because LSTM is capable to exploit the patterns in data and remember information for long time [8]. With memory property, it is wildly used for series data such as financial time series data [9], and text mining task. CNN can be applied to reduce dimension, and it also works for series data due to its sliding processing. A combined approach is proposed with combination of convolutional neural network (CNN) and LSTM for gold price time-series analysis [10]. Combined approach of support vector machine and bag of words is applied with textual information to predict price and direction [11]. Sentiment analysis is rarely applied to predict stock price alone because it extracts sentiment polarity from given text. Nevertheless, these approaches are applied to sentiment classification problems by labeling given texts based on stock price change. Recently, an increasing number of approaches based on two or more machine learning approaches is introduced into stock price prediction and direction. Price prediction of multiple companies based on neural network with concatenation of news and historical prices data of multiple companies is proposed [12]. Price prediction models based on dictionary-based text mining, time series analysis, and word embedding are performed with different combination of independent variables including news, polarity to news, historical prices [13]. SVR is also combined with sentiment analysis of tweets that work well on price forecasting tasks [14].

3. Data Description

We choose to obtain news and opinion headlines of six stocks from different sectors from Investing.com, which are Bank of America Corporation (BAC) in financial services, The Boeing Company (BA) in industrials, Exxon Mobil Corporation (XOM) in energy, Uber Technologies, Inc. (UBER) in technology, Johnson & Johnson (JNJ) in healthcare, and Apple Inc. (AAPL) in technology. The textual dataset we use covers the various period depending on stock. Reason to choose this dataset is because there are trading days without news for each stock (See Table 1).

Table 1: Numbers of news, days, and days without news.

Ticker	Period		# of news	# of days without news	# of total days
BA	2008-10-30	2020-01-30	5773	1460	2831
BAC	2008-10-08	2020-01-28	7234	825	2845
XOM	2009-05-21	2020-02-12	2763	1740	2701
UBER	2019-05-14	2020-01-21	1195	2	174
JNJ	2012-07-23	2020-02-10	750	1456	1900
AAPL	2012-0717	2020-01-27	19971	158	1894

Lack of information for days practically happens so that we choose historical prices as additional information. Historical prices are time series data and easy to obtain, and combination of two types of data should be helpful to show the individual influence of news and opinions. We obtain historical prices of the six stocks on Yahoo! Finance website over the same period in order to complement news and opinions data.

4. Data Processing

In this paper, we apply word tokenization, stop-words removal, stemming, and part-of-speech tagger for textual dataset, and data transformation for numerical dataset prior to further modelings. In this research, word tokenization also includes normalization such as lowercasing and abbreviation replacing. Our stop-words list is originally from Natural Language Toolkit (NLTK) which is a platform for building Python programs to work with human language data, which contains 127 stop-words. We added new stop-words based on word frequency and removed stop-words related to direction. For price data, we applied transformation logarithms such as MinMaxScaler or log transformation depending on error measures.

5. Evaluation Methodology

In this research, three error measures are applied, which are root mean squared error (RMSE), mean absolute error (MAE), and mean absolute percentage error (MAPE) as shown:

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i' - y_i)^2} \tag{1}$$

$$MAE = \frac{1}{n}\sum_{i=1}^{n}\left|y_i - y_i'\right| \tag{2}$$

$$MAPE = \frac{1}{n}\sum_{i=1}^{n}\left|\frac{y_i - y_i'}{y_i'}\right| \tag{3}$$

where y_i is the actual value and y_i' is the prediction for the corresponding actual value. MAE and RMSE measure difference between predicted prices and actual prices. MAPE shows the difference between predicted and actual prices in average percentage so it can compare one stock with other five stocks. For easier interpretability, MAPE would be multiplied by 100 in the result.

6. Experiment

Preprocessing for experimental setting, numerical dataset and textual dataset would be joint based on date so there are days without news. Afterwards, joint dataset would split into training and test dataset by 80/20 without shuffle because we prefer to keep it a time series of prices and news headlines. We focus on the combination of time series analysis and sentiment analysis to better forecast stock price, and our combination approach includes four parts:

1. Find a time series model which lead smallest error. Predicted prices by the model are considered as new input data.
2. Use word embedding approach to obtain representation matrix of daily news headlines for each stock as new input data.
3. Apply lexicon-based sentiment approach to obtain the polarity of each daily news headline in score as new input data.
4. Perform neural networks or support vector regression with these new inputs for stock price forecasting.

For time series analysis, window size as a key factor is number of prices of the past days for current price prediction. We experiment different window sizes (3, 4, 5, 10, 15, 20, 30) and window size is decided by the model with smallest error (MAPE, MAE, and RMSE). Secondly, we would perform all models from 6.1 to 6.5, and the one with smallest error is used to further predict stock prices.

Time series models includes Model_1, Model_2a, Model_2b, Model_3, Model_4a, and Model_4b which are introduced from Section 6.1 to 6.5. Word embedding models include Model_5a, Model_5b, and

Model_5c which are introduced from Section 6.6 to 6.8. Combined models include Model_6, Model_7, Model_8, and Model_9 which are introduced in Section 6.9 to 6.12.

6.1. Support Vector Regression Model (Denoted Model_1)

Experiment based on Support vector regression (SVR) used as baseline models from previous researches [15]. The input data for model_1 is a time series of closing prices, P_1 $P_2 ... P_t ... P_n$, for n trading days, and they perform SVR to predict price at time t by considering $P_{t-4}, P_{t-3}, P_{t-2}, P_{t-1}$ which are price at time *t-4*, *t-3*, *t-2*, and *t-1* respectively. Kernel function and hyperparameters are selected by grid search to reach the smallest as possible.

6.2 Unidirectional Long Short-term Memory (Denoted Model_2a)

Kumar and Ningombam perform long short-term memory (LSTM) to implement technical analysis for stock price forecasting of APPL in 2018 [16]. Their training details include dropout of 0.25, learning rate is 0.001, adam optimizer, epochs of 50, and linear activation for output layer activation. We also applied multilayer LSTM which stacks several LSTM layers. The design is able to capture more complicated patterns of data, but it also increases overfitting risk. The number of layers is from one layer to five layers, and the model with the smallest error is denoted model_2a.

6.3 Bidirectional LSTM (Denoted Model_2b)

According to an analysis of forecasting financial time series by Siami-Namini et al, Bidirectional LSTM results in smaller error than LSTM and ARIMA on NASDAQ index, Nikki 225 index, S&P 500 commodity price index, Dow Jones industrial average index, and IBM stock [17]. Bidirectional LSTM firstly is applied on the input sequence (from past to present) and then on the reverse of input sequence (from present to past) so that it improves the shortcoming of unidirectional LSTM which is moving information from the past. The architecture of our bidirectional LSTM is single layer.

6.4 Convolutional Neural Network (Denoted Model_3)

The Convolutional neural network (CNN) model is based on the idea that price at a certain time is less affected by the prices a long time ago. Recurrent neural networks must go through prices day by day, but on the contrast, CNN model goes through prices subseries days by subseries days. The baseline CNN model consists of one input layer, two convolutional layers, one pooling layer, and a hidden layer. The size of convolutional filter is 1 × 7, and pooling size is 1 × 2, and there are 6 and 12 filters in two convolutional layers respectively [18].

6.5 CNN-LSTM (Denoted Model_4a and Model_4b)

Proposed LSTMs with convolutional layer (CNN-LSTM) provide a boost in prediction performance for gold price in 2020 [10]. Raw data is converted into new feature values by element-wise multiplication with kernel in convolutional layers. These feature values are subsampled in pooling layer for lower dimension. Their first architecture of CNN-LSTM consists of two convolutional layers of 32 and 64 filters of size (2,), respectively, followed by a max-pooling layer with size (2,), a LSTM layer of 100 units, and a output layer of one neuron,, denoted Model_4a. The second architecture consists of two convolutional layers of 64 and 128 filters of size (2,), respectively, followed by a max-pooling layer with size (2,), a LSTM layer of 200 units, a hidden layer of 32 neurons and a output layer of one neuron, denoted Model_4b.

6.6 Pretrained Word2vec Model (Denoted Model_5a)

Inspired by prediction based on sentiment score of news headlines and historical prices, polarity of news headlines is informative so that new headlines vector might also provide useful information. We convert news headlines into vector by pretrained word embedding model and LSTM, and then use news headline vectors as input and actual price on the same day as target. We apply package gensim, pre-trained word2vec model to convert word into vector, and afterward, the series of word vectors are fed into recurrent neural network to extract information [19].

6.7 Word Embedding Model Based on Our Textual Dataset (Denoted Model_5b)

Unlike usage of pre-training language models, we train our word embedding model based on our textual data, and word vectors are produced by our embedding model.

6.8 Bidirectional Encoder Representations from Transformers (Denoted Model_5c)

Bidirectional encoder representations from transformers (BERT) are language representation models from unlabeled text [20], and it is a pre-training transformer-based machine learning approach for NLP task by Google. Every news headline is converted into headline embedding with 1×768 of dimension, and then support vector regression with grid search is applied to forecast stock price on next day.

6.9 Model Based on Historical Prices and Sentiment Score (Denoted Model_6)

Mohan et al combined lexicon-based sentiment analysis and LSTM for stock price prediction [13]. Words of news headline is converted into sentiment score, and then pair of historical price and sentiment score is used to forecast price. They perform LSTM to predict price at time t by considering pairs ($Price_{(t-1)}, Score_{(t)}$),…,($Price_{(t-m)}, Score_{(t-m-1)}$), where window size m is decided by error, $Price_{(t-m)}$ is price at time *t-m*, $Score_{(t-m-1)}$ is score at time *t-m-1.*

6.10 Model Based on News Headlines and Predicted Prices (Model_7)

Instead of identifying the polarity of news headlines, news headline at time t is processed by word embedding model, and then concatenated with predicted prices at time t by time series model for further prediction on stock price at time t. There are two situations for word2vec and BERT respectively. In the first situation, headline is converted into headline vector by BERT, concatenated with predicted prices by timeseries model, and then support vector regression is applied for prediction when Model_5c is better than other embedding models. In second situation, headline is converted into a sequence of word vectors, and then converted into headline vector by LSTM. Afterwards, the headline vector is concatenated with predicted prices by time series model, and the neural network is applied for prediction, when Model_5a and Model_5b are better than Model_5c.

6.11 Model Based on Historical Prices, Sentiment Scores, and Predicted Prices

This model, denoted Model_8, is constructed on the idea of Model_6, which bases on pairs of sentiment score and historical prices. For Model_6, stock price at time *t* is predicted by pairs of historical price at time *t-1* and sentiment score at time *t*. For Model_8, stock price at time *t* is predicted by pairs of historical price at time *t-1*, sentiment score at time *t*, and predicted price at time *t*. The predicted prices base on the BERT language model which is Model_5c. Afterward, CNN-LSTM is applied for price forecasting.

6.12 Model Based on News Headlines Vector, Sentiment Score, and Predicted Prices

The last model is extension of model_7 which combines sentiment score at time *t*, headline vector and predicted price at time *t*, and this model is denoted Model_9. We assume that sentiment scores by lexicon and headline vector should be complementarily informative. When Model_5c is better than Model_5a and Model_5b, headline vector by BERT is concatenated with sentiment score by lexicon and predicted price by time series model, and support vector regression is applied for price forecasting. When Model_5a and Model_5b are better than Model_5c, headline is converted into headline vector by word embedding model and LSTM, the headline vector is further concatenated with sentiment score by lexicon and predicted price by time series model, and then neural network is applied for price prediction.

7. Result and Performance

In this section, proper window sizes by experiments are shown in Table 2 at first, and then we show the results using all models for chosen six stocks as shown from Section 7.1 to 7.6. Since error measures sometimes does not improve, we also use graphs for our decision-making. In each section, we would present one table by best models among time series models, word embedding models, and combined models, and followed by its prediction graph and prediction graph of the model with smallest error.

Table 2: Better window size with errors for six stocks.

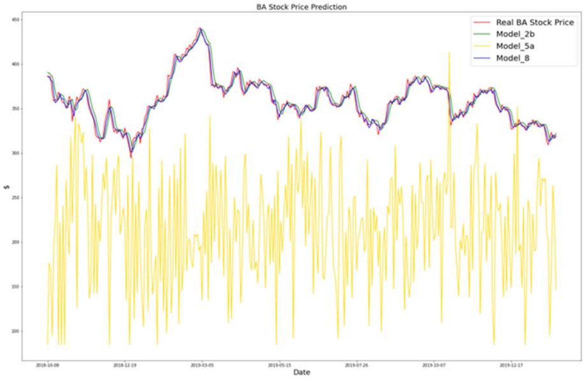
Stock	Window size	MAPE	MAE	RMSE
BA	3	1.9164	6.7913	9.1339
BAC	10	1.8153	0.5271	0.6446
XOM	3	1.3685	1.0141	1.3214
UBER	3	2.7847	0.8476	1.0531
JNJ	3	1.2789	1.7449	2.3005
AAPL	4	2.0712	1.1166	1.3669

7.1 The Boeing Company (BA)

Among the time series models, bidirectional LSTM (Model_2b) produces smallest error which are MAPE of 1.7766, MAE of 6.2972, and RMSE of 8.4879. Among word embedding models, Model_5a produces smallest errors, and none of embedding models is able to catch pattern, which means predictions based on embedding model are unreliable. For combined models, compared with Model_6 which consists of CNN-LSTM with sentiment score of news headline, and historical prices, Model_8 shows an improvement in 1.4269 of decrease in MAPE, 5.2884 of decrease in MAE, and 5.9117 of decrease in RMSE, which means that predicted prices by BERT are informative. From Fig. 1: Left, regression line by Model_2b resembles moving average line which shows lag between regression line by Model_2b and actual prices. Model_8 as combined model with smallest errors shows less lag than Model_2b with similarly small errors. With sentiment scores as new input feature, error decreases from Model_7 to Model_9, but combined model_7 and Model_9 are still less reliable than Model_6 and Model_8.

Table 3: The best model in time series and word embedding models, and combined models for BA.

Ticker	**Model**	**MAPE**	**MAE**	**RMSE**
BA	Model_2b	1.7766	6.2972	8.4879
	Model_5a	40.2427	145.4781	160.0848
	Model_6	2.3790	8.6836	10.4685
	Model_7	52.1324	187.4539	128.1867
	Model_8	0.9521	3.3952	4.5568
	Model_9	42.9119	154.4238	156.1739



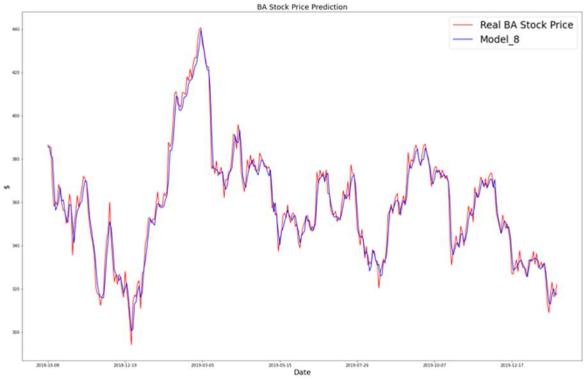


Fig. 1: Left: Stock prices plotted on zero axis. Right: Predicted prices by the best model plotted.

7.2 Bank of America Corporation (BAC)

Among the time series models, bidirectional LSTM gives smallest errors which are MAPE of 1.1405, MAE of 0.3282, and RMSE of 0.4383 than others. None of embedding models seem functional because of smallest MAPE by embedding models is 29.8403. Besides, combined models, Model_6 and Model_8, produce similarly small errors as time series model. Especially, errors by Model_8 are smaller than by bidirectional LSTM, which also shows more reliability and less lag in Fig. 2: Left. Despite smaller error by Model_9 than by Model_7, Model_5a outperforms Model_7 and Model_9, which is not reasonable. The possible reason could be the way to combine headline vector and predicted prices.

Table 4: The best model in time series and word embedding models, and combined models for BAC

Ticker	Model	MAPE	MAE	RMSE
BAC	Model_2b	1.1405	0.3282	0.4383
	Model_5a	29.8403	8.9151	10.2604
	Model_6	1.2356	0.3558	0.4437
	Model_7	36.1119	10.6882	11.0415
	Model_8	0.5406	0.1556	0.1955
	Model_9	28.6205	8.4836	8.7312

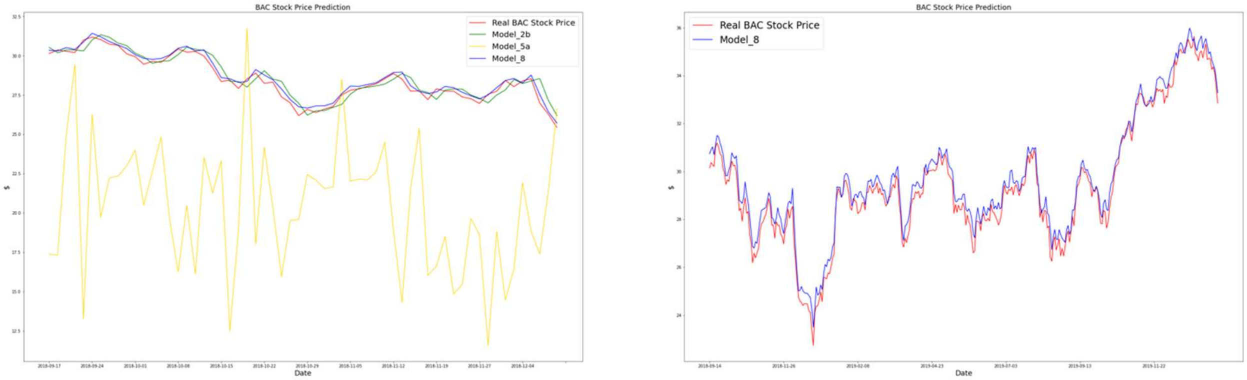


Fig. 2: Left: Stock prices plotted on zero axis. Right: Predicted prices by the best model plotted.

7.3 Exxon Mobile Corporation (XOM)

Overall, time series models give great performance excluding support vector regression model, and bidirectional LSTM leads smallest error than other time series models, which is MAPE of 1.0858, MAE of 0.8098, and RMSE of 1.0289. Word embedding model looks much more functional than in BA, BAC cases, which give MAPE of 11.1683 under conditions without news during 1740 out of 2701 days. With such informatively numerical and textual data, there is no doubt that Model_6 and Model_8 show small errors. Besides, Model_7 and Model_9 seem no improvement to embedding model, and reason could be word2vec might not extract important information.

Table 5: The best model in time series and word embedding models, and combined models for XOM.

Ticker	Model	MAPE	MAE	RMSE
XOM	Model_2b	1.0858	0.8098	1.0289
	Model_5a	11.1683	7.9629	9.8078
	Model_6	0.9327	0.6789	0.8942
	Model_7	12.9858	9.3518	10.7769
	Model_8	0.6887	0.5212	0.6407
	Model_9	11.6688	8.4274	9.5500

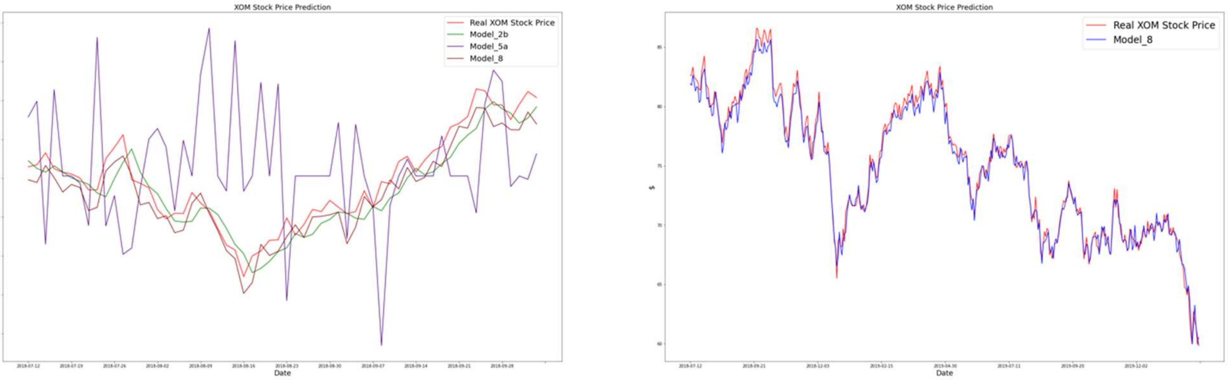


Fig. 3: Left: Stock prices plotted on zero axis. Right: Predicted prices by the best model plotted.

7.4 Uber Technologies, Inc. (UBER)

Prediction for Uber price is much worse than for other stocks, all models for UBER give much larger errors than for other stocks. Small sample might cause the result. There are 140 days for training and 34 for test. Bidirectional LSTM still leads to smallest errors which is MAPE of 2.3639 among time series models. Surprisingly, self-trained embedding model produces smallest errors among embedding models. The reason might be quality and quantity of news headlines. Besides, Model_8 is able to give similar small error with Model_2b. Despite larger error, Model_8 shows less lag than Model_2b in Fig. 4: Left. Model_9 shows no improvement after adding sentiment scores as new feature, and reason could be small sample size and stochastic characteristic of neural network.

Table 6: The best model in time series and word embedding models, and combined models for UBER.

Ticker	Model	MAPE	MAE	RMSE
UBER	Model_2b	2.3639	0.7551	0.9751
	Model_5b	12.9310	3.9197	4.6411
	Model_6	4.1442	1.3402	1.7054
	Model_7	10.6323	3.2549	4.0323
	Model_8	3.0765	0.9387	1.0554
	Model_9	15.1156	4.8640	5.4916

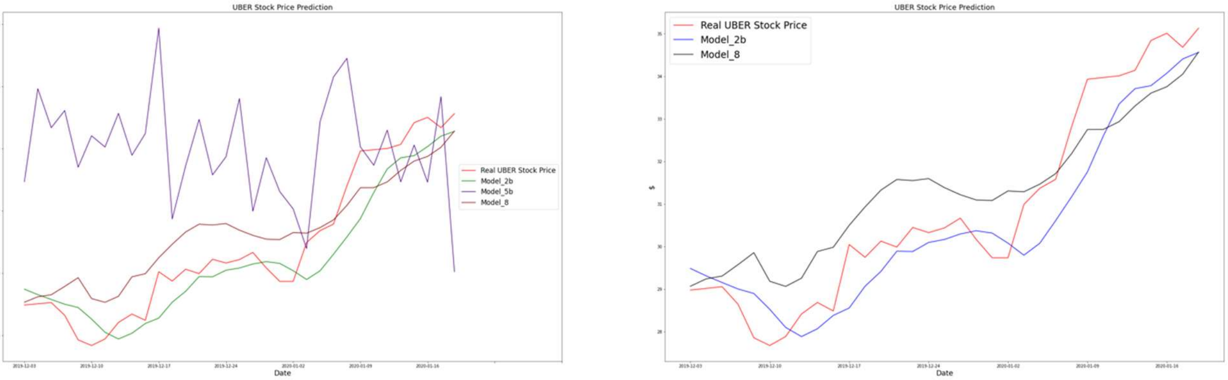


Fig. 4: Left: Stock prices plotted on zero axis. Right: Predicted prices by the best model plotted.

7.5 Johnson & Johnson (JNJ)

Model_2b based on bidirectional LSTM shows better performance among time series models again, and unidirectional LSTM gives similar errors. Since there are days without news, embedding models produce large errors and are less reliable than time series models and combined models. Model_6 as combined model with sentiment score and historical price leads to similar performance with time series models based on neural networks, and Model_8 produces smallest errors among all models. Since Model_5c outperforms Model_5a and Model_5b, support vector regression is applied for prediction. Comparison of Model_7 and Model_9, it seems that headline vector by BERT is more informative than by word2vec and LSTM.

Table 7: The best model in time series and word embedding models, and combined models for JNJ.

Ticker	Model	MAPE	MAE	RMSE
JNJ	Model_2b	0.9566	1.2965	1.8897
	Model_5c	18.9405	26.0472	28.2778
	Model_6	1.0636	1.4393	2.0482
	Model_7	3.6074	4.9771	5.3382
	Model_8	0.7580	1.0371	1.3188
	Model_9	3.6029	4.9709	5.3326

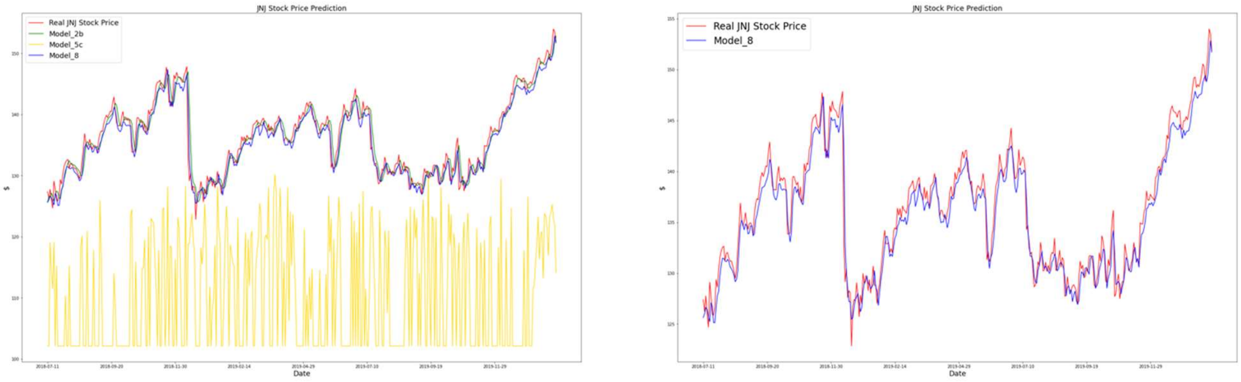


Fig. 5: Left: Stock prices plotted on zero axis. Right: Predicted prices by the best model plotted.

7.6 AAPL Inc. (AAPL)

Bidirectional LSTM (Model_2b) outperforms other time series models for every stock. Quality of news headlines might play an important role in analysis because there are only 158 days without news out of 1894 days and errors for embedding models are larger than 30 which means there is more than 30% difference between predictions and real prices. Comparison between Model_7 and Model_9, sentiment scores are not informative with headline vector by BERT. Model_8 again gives smallest MAPE, MAE and RMSE.

Table 8: The best model in time series and word embedding models, and combined models for AAPL.

Ticker	Model	MAPE	MAE	RMSE
AAPL	Model_2b	1.6852	0.8788	1.1741
	Model_5c	33.1868	18.4446	20.8859
	Model_6	1.8301	0.9519	1.1928
	Model_7	20.0284	11.2110	12.8617
	Model_8	1.0405	0.5222	0.6774
	Model_9	20.0526	11.2261	12.8853

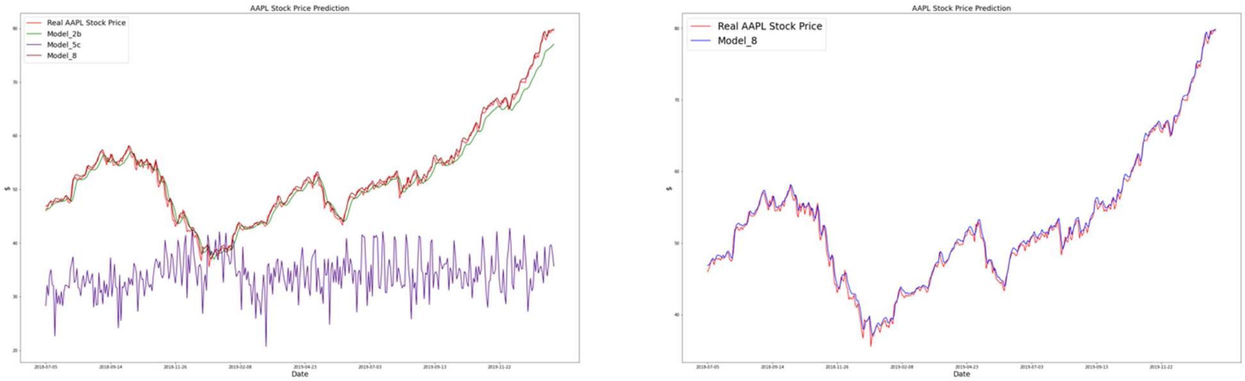


Fig. 6: Left: Stock prices plotted on zero axis. Right: Predicted prices by the best model plotted.

8. Conclusion and Future Work

In this work, we focus on stock price prediction by combination of different models including sentiment scaling system, word embedding model, and time series model to better analyze time series data in financial field. However, it is questionable how to combine two types of information due to dimension. Numerical information is usually one dimension but it is more complicated to textual information. Especially for word embedding models, BERT produces 768 dimensions of headline vector and 300 dimensions by word2vec. Concatenation of one dimension and high dimension of matrix might cause problem and this might be the reason why Model_7 and Model_9 does not reach our expectation.

Time series models are able to grab periodic status, which performance prediction with small error. Practically, it is unhelpful in finance field because predictions by time series models are likely either higher than actual prices or lower than actual prices. On the graphs, regression line by time series data resembles a smooth move average line, which comes lagging.

Sentiment scaling system is popular method to extract polarity of given text but nonavailability of sentiment lexicon happens. One solution is application of word embedding model which is able to convert given text into informative matrix, and the matrix somehow complements nonavailability of sentiment lexicon. They can be used for sudden and short-term influence.

Despite stochastic characteristic of neural networks, bidirectional LSTM (Model_2b) outperforms the other time series models in 6 out of 6 stocks, but on the figures, it does not look like models with low error. Prediction based on word embedding models is unreliable because of high error. The high error might be caused by the days without news. Our proposed model (Model_8) is combination of these models, and result shows the potential not only to keep as lower error as time series models and but also to extract information from words. In 5 out of 6 chosen stocks, Model_8 shows beneficial performance, and also show sensitivity on graph. Regression line by Model_8 shows less lag than other models. There are still various ways to concatenate two type of information and expanded error measures, which we leave as future work.

References

[1] S. Bharathi and A. Geetha, “Sentiment Analysis for Effective Stock Market Prediction,” *International Journal of Intelligent Engineering and System*, vol. 10, no. 3, 2017.

[2] A. A. Adebiyi and O. A. Adewumi, “Stock price prediction using the ARIMA model,” in *UKSim-AMSS 16th International Conference on Computer Modelling and Simulation*, 2014, pp. 106-112.

[3] H.D. Huynh, L.M. Dang and D. Duong, “A new model for stock price movements prediction using deep neural network,” in *Proceedings of the Eighth International Symposium on Information and Communication Technology*, 2017, pp. 57-62.

[4] W. Khan, M. A. Ghazanfar, M. A. Azam, A. Karami, K. H. Alyoubi and A. S. Alfakeeh, “Stock market prediction using machine learning classifiers and social media, news,” *Journal of Ambient Intelligence and Humanized Computing*, pp. 1-24, 2020.

[5] C. C. Emioma and S. O. Edeki, “Stock price prediction using machine learning on least-squares linear regression basis” in *Journal of Physics: Conference Series*, vol. 1734, no. 1, 2021.

[6] J. T. Yao and C. L. Tan, “Guidelines for financial forecasting with neural networks,” in *International Conference on Neural Information Processing*, pp. 757-761, 2001.

[7] Y. Musa and S. Joshua, “Analysis of ARIMA-artificial neural network hybrid model in forecasting of stock market returns,” *Asian Journal of Probability and Statistics*. vol. 6, no. 2, pp. 42-53, 2020.

[8] S. Hochreiter and J. Schmidhuber, “Long short-term memory,” *Neural Computation*, vol. 9,no. 8,pp. 1735-1780, 1997.

[9] A. Yadav, C. K. Jha and Sharan, “Optimizing LSTM for time series prediction in Indian stock market,” in *International Conference on Computational Intelligence and Data Science (ICCIDS)*, pp. 2091-2100, 2020.

[10] I. E. Liveris, E. Pintelas and P. Pintelas, “A CNN-LSTM model for gold price time-series forecasting,” *Neural Computing and Applications*, vol. 32, no. 23, pp. 17351-17360, 2020.

[11] R. P. Schumaker and H. Chen, “Textual analysis of stock market prediction using breaking financial news: The AZFin text system,” *ACM Transactions on Information Systems (TOIS-09)*. vol. 27, no. 2, pp. 12:1-12:19, 2009.

[12] R. Akita, A.Yoshihara, T. Matsubara and K. Uehara, “Deep learning for stock pre-diction using numerical and textual information,” in *2016 IEEE/ACIS 15th International Conference on Computer and Information Science (ICIS)*, IEEE, 2016, pp. 1-6.

[13] S. Mohan, S. Mullapudi, S. Sammeta, P. Vijayvergia and D. C. Anastasiu, “Stock price prediction using news sentiment analysis” in *2019 IEEE Fifth International Conference on Big Data Computing Service and Applications*, IEEE, 2019, pp.205-208.

[14] H. Maqsood, I. Mehmood, M. Maqsood, M., Yasir, S. Afzal, F. Aadil, M. M. Selim and K. Muhammad, “A local and global event sentiment based efficient stock exchange forecasting using deep learning,” *International Journal of Information Management*, pp. 432-451, 2020.

[15] H. Yang, L. Chan and I. King, “Support vector machine regression for volatile stock market prediction,” in *International Conference on Intelligent Data Engineering and Automated Learning*, Springer, Berlin, Heidelberg, 2002, pp. 391-396.

[16] S. Kumar and D. Ningombam, “Short-term forecasting of stock prices using long short term memory,” in *International Conference on Information Technology (ICIT),* 2018.

[17] S. Siami-Namini, N. Tavakoli and A. S. Namin, “A comparative analysis of forecasting financial time series using arima, lstm, and bilstm,” 2019, arXiv preprint arXiv:1911.09512.

[18] J. Cao and J. Wang, “Stock price forecasting model based on modified convolution neural network and financial time series analysis,” *International Journal of Communication Systems*, vol. 32, no. 12, 2019.

[19] Google code archive - long-term storage for Google code project hosting. (n.d.). Retrieved April 05, 2021, from https://code.google.com/archive/p/word2vec/

[20] J. Devlin, M. W. Chang, K. Lee and K. Toutanova, “Bert: Pre-training of deep bidirectional transformers for language understanding,” 2018, arXiv preprint arXiv:1810.04805.