Sentiment Prediction of Chinese Stock Message Board Posts with Recurrent Neural Network

by

Guangyu Wu

A Thesis Submitted to the Department of Engineering and Computer Science

New York University Shanghai

In Partial Fulfilment of the Requirements

for the Degree of Bachelor of Science in Data Science

May 2020

Acknowledgement

I would like to express my sincere gratitude to my faculty advisor Professor Ratan Dey for his guidance and encouragement throughout the process of developing this project. Professor Dey helped me identified the research idea and made concrete research plans. We also explored the social network analysis of the dataset under research.

I also want to thank Professor Zhao Bin for teaching me the China financial market class, from which this paper originates. She inspired me to investigate the behavioral aspect of stock message board and gave me valuable suggestions on this project.

Sentiment Prediction of Chinese Stock Message Board Posts with Recurrent Neural Network

Guangyu Wu

ABSTRACT

Online stock message boards have gained popularity in recent years as they provide channels for individual investors to share their opinions on the financial market. Naturally, many fintech companies, market regulators, as well as researchers seek to analyze investor sentiments based on these sources. However, the extraction of sentiment from stock message board posts is not always straightforward, as the posts vary in content and often have a rich connotation. In this paper, a recurrent neural network-based post sentiment classifier is proposed to overcome the limitations of traditional dictionary-based and machine learning-based classifiers. Different sentiment classification models are compared and reasons for improvements are discussed. The proposed RNN classifier is then tested in an automatic trading strategy simulation, which demonstrates the information value of post sentiments in the financial market.

Keywords: Sentiment Analysis, Recurrent Neural Network, Stock Message Board

I. Introduction

With high speed, broad audience, and negligible costs, online stock message boards have become an important platform for individual investors to share their opinions on investments. While these online platforms facilitate the exchange of information, they also produce a large amount of textual data on investor sentiments. In contrast to the numerical trading data, the posts on stock message boards can capture the intricate thoughts of investors and represent investor sentiment more directly. Financial technology firms can build better trading strategies with a deeper understanding of investor sentiment, and market regulators can adjust their policies according to the public sentiment as well.

However, extracting sentiment from textual data is also more challenging. A post on the stock message board could be a piece of buying (or selling) suggestion, or a question on future stock moves, or even a satire on others' posts. The complex nature of post content means it is difficult to extract sentiment (positive, negative, neutral) from the posts automatically. Fortunately, recent years have witnessed the advent of powerful deep learning models that specialize in solving complex natural language processing problems. This paper sets out to apply the deep learning model, especially the Bi-directional Long-Short Term Memory model, to the task of predicting the sentiment of stock message board posts.

The rest of the paper is organized as follows: Part II will conduct a literature review on stock message board research with a focus on sentiment prediction, Part III explains the choice of data source and collection methods, Part IV lists the main models and prediction pipelines, Part V compares and discusses the performance of various models, Part VI concludes.

II. Literature Review

The research in online stock message boards originated in the late 1990s, shortly after the establishment of influential stock message boards such as Yahoo! message boards and RagingBull.com. One of the earliest studies found that the number of messages posted about a stock overnight can predict that stock's trading volume and returns on the next day (Wysocki 1998). Tumarkin and Whitelaw went one step further to incorporate the opinion tags[1] and construct the "Average Daily Weighted Opinion" factor (2001). They concluded that abnormal stock returns predict the volume and sentiment of posts but not vice versa.

The early studies analyzed posting volume and available sentiment labels but failed to utilize the textual content of posts. Antweiler and Frank popularized the use of computational linguistic methods to analyze stock discussion posts in their influential Journal of Finance paper (2004). The authors first manually labeled 1,000 posts and adopted Naive Bayes to learn the sentiment connotation of 1,000 most frequent words in their labeled data. Then, they constructed a post sentiment classifier and calculated "bullishness" measure and "disagreement" measure for 1.5 million posts. In the end, they discovered a positive relationship between the post sentiment and stock market volatility, and a statistically significant but economically weak relationship between post sentiment and next-day returns.

Following this study, more researchers start to apply text classifiers to financial message classification. Takahashi et al. used the Naive Bayes classifier to investigate the relationship between financial news headlines and stock price returns (2007). Schumaker and Chen chose the

[1] In the early days of RagingBull.com, users need to select a label for their posts ("Strong Buy, Buy, Hold, Sell, Strong Sell"), which will be displayed alongside the post content.

SVM text classifier to examine the relation between news articles and stock quotes (2008). Zhang and Swanson devised a Maximum Entropy text classifier, arguing that ME works better in discourse analysis (2010). The authors of these research shared the view that there was no standard text classifier for analyzing financial texts. This inconsistency motivated Zhang et al. to compare and contrast eight widely applied text classifiers on stock message board data in their study (2015). Their results reported out-of-sample prediction accuracies ranging from 48.1% to 61.3% for the eight classifiers, and the authors summarized common pitfalls including class imbalanced problem and noisy content in certain types of posts.

Around the time when Zhang et al.'s comparison study came out, two more papers brought emotion space model and topic modeling-based sentiment analysis to the field (Luo et al. 2015; Nguyen and Shirai 2015). Their approaches extend the pool of emotion words and improve the prediction of domain-specific posts. More recently, an increasing number of computer science researchers start to explore the application of deep learning in the stock market, and textual data have come into their horizon. For instance, Li et al. extracted investor sentiment from forum posts using Naive Bayes and feed the sentiment data into their LSTM-based stock prediction model (2016). The addition of sentiment measure boosted prediction performance and the authors believe the sentiment input helped capture the irrational component of stock prices. However, in many studies including Li et al., the deep learning model is applied only in stock prediction, not the extraction of sentiment labels from posts. Thus, this paper will fill in the gap by investigating the performance of deep learning models in sentiment classification of stock message board posts.

Finally, it is worth noting that the research on stock message board and financial text sentiment classification in China started later than in the English-language world. The few studies on

Chinese stock message boards are published mostly in the last seven years. Jin et al. collected 1 million posts from the stock message board Eastmoney Guba and built a KNN classifier to label sentiments (2013). In contrast, Yi et al. extracted sentiment from Eastmoney Guba posts by counting sentiment keyword frequency guided by an expert-created dictionary (2015). The usage of an expert dictionary is also adopted by Zhang and Yuan when they analyze finance-related microblogs on the Sina Weibo, for its robustness against noise (2017). The popularity of expert dictionary in sentiment analysis also reflects the lack of reliable machine learning or deep learning-based sentiment classifier on Chinese financial texts.

As discussed above, this paper aims at applying popular deep learning models to predict the sentiment of Chinese stock message board posts. The advantages and drawbacks of adopting recurrent neural networks in this task will be discussed and compared against those of expert dictionary-based sentiment extraction and other machine learning-based methods such as SVM. The goal of this paper is also to take the first step towards integrating deep learning methods in analyzing financial texts in Chinese. Much efforts have been devoted to the direct prediction of stock market movements using deep learning, but the sentiment information of individual posts on stock message boards or stock related social media are also valuable. Reliable sentiment classifiers can benefit the study of investor behaviors, opening up new research opportunities in behavioral finance; or provide the basis for analyzing emotional contagion and communication dynamics in online space.

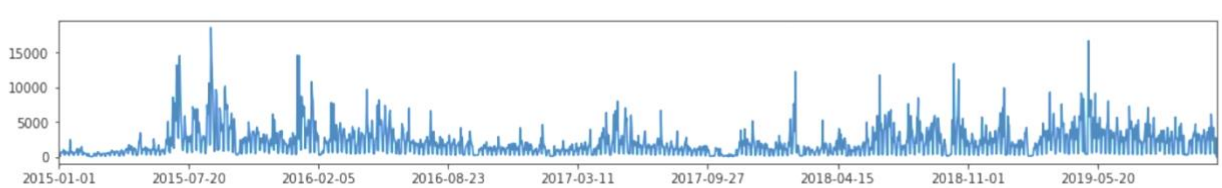
III. Data

The stock message board posts used in this research were collected from an online forum dedicated to the discussion of the Shanghai Stock Exchange Composite Index on the Eastmoney

Guba website. Eastmoney Guba (literary translation as “Oriental Money Stock Forum”) is one of the most influential stock message boards in China. Compared to other stock message boards, Eastmoney Guba has the most extensive user base and daily posting volume.

The stock message board Eastmoney Guba has many sub-forums. The forum dedicated to discussing the SSE Composite Index is chosen for two reasons. First, compared to the discussions in company-specific forums, discussions on stock index tend to have fewer firm-specific or domain-specific keywords that complicate the sentiment classifier. Second, multiple research on Chinese stock message boards selected it as the financial time series to study (Huang et al. 2015; Yi et al. 2015; Chen and Yan 2017; Zhang and Yuan 2017). Having the same underlying as these research makes it possible to borrow the expert-picked emotion keywords from previous studies.

The data were downloaded using a Python scraper program. The resulting raw dataset contains more than 4 million posts with title, body, and rich metadata. The metadata include second-level timestamps, number of views and comments, etc. Empty posts, duplicate posts, and posts whose author has unregistered are removed. And posts before January 2015 were dropped since there are too few posts per day during that period, making it less representative of general investor sentiment. The cleaned dataset contains 3,602,693 posts from January 1st, 2015, to November 14th, 2019.



Graph 1. Number of posts each day

5000 posts are randomly drawn from the time period from January 2015 to January 2016 for manual sentiment labeling. Each post is assigned a sentiment label (positive, neutral, negative) by two human labelers who have knowledge of the China financial market. The labeled dataset is further divided into a training set covering the first ten months and a test set covering the last two months. The dataset is split into two mutually exclusive time periods instead of a random split to ensure no information from the period of test set leaks into the training set.

In addition, the historical prices of the SSE index are collected from the WIND terminal, which will be used in the empirical tests of sentiment prediction performance.

IV. Models and Methodology

Seven different sentiment prediction models are implemented and compared in this paper. Two of them are based on sentiment keyword dictionary and manually encoded rules. Another two models use the bag-of-words representation of the posts and apply traditional machine learning methods. The last three models use word embedding representation of the text and deep learning methods. One preprocessing procedure common to all models is Chinese text segmentation, implemented with the Jieba python module[2].

A) Dictionary based-models

A1. Expert Dictionary

[2] https://github.com/fxsjy/jieba

The first model is a simple yet time-proven count-based sentiment classifier. The sentiment keywords[3] from Chen and Yan's study are used for their clear indication of positive or negative sentiment. The sentiment score of a post is calculated with the formula below and a label is assigned based on the sign of this score.

$$Sentiment\ score = \frac{\#\ positive\ keyword\ -\ \#\ negative\ keyword}{\#\ of\ tokens\ in\ the\ post}$$

For illustration, consider the following hypothetical post:

$$The\ stock\ index\ will\ \textcolor{green}{\boldsymbol{rise}},\ and\ it\ won't\ \textcolor{red}{\boldsymbol{drop}}\ until\ tomorrow!\ Who\ said\ the\ stock\ market\ will\ \textcolor{red}{\boldsymbol{crash}}??$$

$$Sentiment\ score = \frac{1\ -\ 2}{21} = -0.047$$

$$Sentiment\ label = -1$$

The example above also shows the limitation of the expert dictionary-based classifier. Chen and Yan also pointed out that one might use sentiment keywords with a negation word around, e.g. “will not bump up” and “cannot crash.” Such scenarios will invert the emotion of a sentence or even the entire post, thus interfering with the overall sentiment calculation.

A2. Expanded Dictionary

To address such cases, this paper includes a rule-based sentiment classifier that inverts the emotion of a clause when it detects the presence of negation words around sentiment keywords.

[3] Positive words: 救市,抄底,牛市,涨停,多头,利多,反弹,回档,增仓,护盘,开户,改革牛,慢牛

Negative words: 股灾,爆仓,跌停,暴跌,停牌,熊市,空头,利空,割肉,逼空,抛售,离场,崩盘,跳水,打压,洗盘,阴跌,套牢

In the meantime, the sentiment keyword dictionary is expanded with text mining techniques to capture sentiment-bearing colloquial terms.

$$Clause\ sentiment$$
$$= (\#\ positive\ keyword\ -\ \#\ negative\ keyword) * (-1)^{\#\ of\ negation\ words}$$

$$Sentiment\ score\ = \frac{1}{\#\ of\ clauses}\sum clause\ sentiment$$

Illustration with the previous example:

$$The\ stock\ index\ will\ \boldsymbol{rise}, and\ it\ \boldsymbol{won't}\ \boldsymbol{drop}\ until\ tomorrow!\ Who\ said\ the\ stock\ market\ will\ \boldsymbol{crash}??$$

$$Sentiment\ score\ = \frac{1}{3}[1 * (-1)^0\ +\ (-1) * (-1)^1\ +\ -1\ *\ (-1)^0]\ =\ 0.33$$

$$Sentiment\ label\ =\ 1$$

With manually encoded rules, the expanded dictionary model can now capture basic sentiment inversions in the posts. However, this method requires careful selection of the negation words, and it still fails to pick up rhetorical questions or other forms of sentiment signals such as punctuations and emoticons. Such limitations motivate the use of machine learning models that learn the connotations of words and sentence structures automatically.

B) Machine learning-based models

B1. Logistic Regression

The logistic regression classifier is adopted because it is a generative-discriminative pair with the Naive Bayes classifier, which is adopted by multiple studies before. Since we are only interested in sentiment classification but not the class conditional probability of words, logistic regression is used for its robustness to dependence between word features.

$$Count\ \ Vectorizer\ \rightarrow\ Tfidf\ \ Transformer\ \rightarrow\ Random\ Over\ Sampler$$
$$\rightarrow\ Logistic\ Regression$$

More specifically, the posts are first turned into their bag-of-words representation. Then, the TF-IDF measure is calculated to evaluate the importance of words. TF-IDF is particularly useful in text classification since it evaluates the importance of certain words in a document versus the whole corpus.

$$Term\ frequency\ =\ \frac{\#\ of\ occurences\ of\ the\ word\ in\ a\ document}{\#\ of\ words\ in\ a\ document}$$

$$Inverse\ document\ frequency\ =\ log(\frac{\#\ of\ documents}{\#\ of\ documents\ containing\ the\ word})$$

Similar to what Zhang et al. observed, the sentiment classes for the training set are imbalanced. A random over sampler is applied to make the numbers of samples in each class equal. Finally, the data is used to train a three-class logistic regression classifier.

B2. Support Vector Machines

The pipeline for support vector machines classifier is similar to that of the logistic regression classifier, except for the last step, an SVM model is trained. In this paper, the SGD classifier from Scikit-learn is used with the standard Hinge loss function and L2 penalty term.

$$Count\ \ Vectorizer\ \rightarrow\ Tfidf\ \ Transformer\ \rightarrow\ Random\ Over\ Sampler \rightarrow\ SGD\ Classifier$$

C) Deep learning-based models

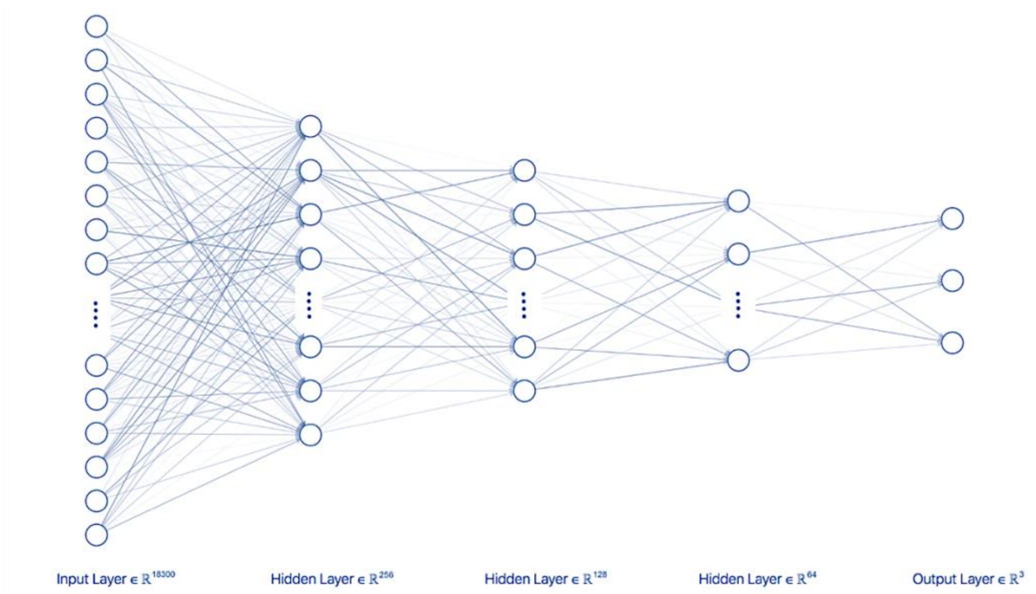
Before diving into the models, it is worth noting that word embedding representation is employed for the three deep learning models. The switch to word embeddings is driven by two main reasons. First, bag-of-words often result in huge but sparse vectors, with the dimension equal to the size of vocabulary in the corpus. Such sparse vectors create stress on memory and make training less efficient. Word embedding provides a stable dimension reduction without losing much information. Second, word embedding is not only a denser representation but also a more expressive one, as it contains the contextual similarity information. Word embeddings trained on a specific type of corpus can provide a better semantic representation of texts in that field. In this paper, word embeddings trained from Chinese finance news[4] are used to transform the posts into the numerical format. More specifically, a post will be normalized to 61 words (covering the length of 95% of the labeled set). Longer posts will be truncated while shorter posts are padded with zeros. The resulting representation of a post is a 61 by 300 matrix (61 words, 300 dimension word vector).

C1. Multilayer Perceptron (MLP)

Multilayer Perceptron is a class of feed-forward neural networks that are popular for supervised learning problems. With several layers of densely connected neurons, an MLP model adjusts the weight and bias parameters to represent the correlations and dependencies between inputs and outputs, guided by the backpropagated losses. In contrast to recurrent neural networks, MLP is a feed-forward neural network because the connections between neurons always point forward and do not form cycles. In this study, MLP is included as a benchmark for the performance of a

[4] https://github.com/Embedding/Chinese-Word-Vectors

vanilla artificial neural network. Note that the word embedding representation of a post is first flattened to an 18,300 dimension vector. To handle the intricacy in the posts, the MLP model has three hidden layers with 256, 128, 64 hidden units respectively. Three random dropout layers are appended after each hidden layer to prevent overfitting. Finally, a softmax layer is added to output categorical predictions.

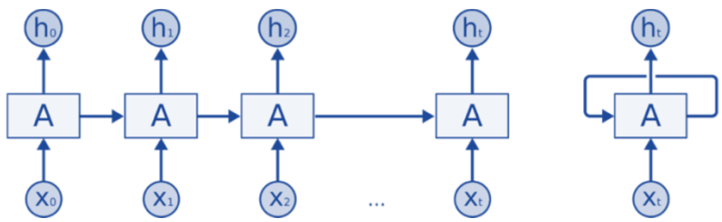


Graph 2. Multilayer Perceptron Network Structure

C2. Stacked Long Short-Term Memory (LSTM)

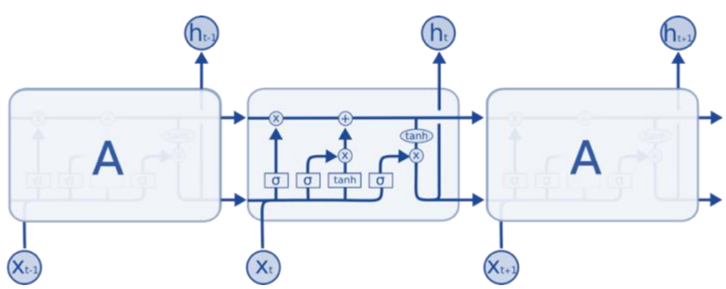
Recurrent neural networks are known for modeling sequential or time-series data well, common applications include machine translation and stock quotes prediction. Its hidden layers have the same weights and bias throughout the prediction process, which gives the RNN ability to memorize information from earlier steps. In the natural language processing context, an analogy might be made that the RNN model remembers the first words in the sentence when analyzing the later part of a sentence.

The special architecture of RNN, however, is a blessing and a curse. Since the same weights and bias are used repeatedly in the model, RNNs often suffer from vanishing or exploding gradient problems, and the memory about earlier data will also gradually get lost.



Graph 3. General Recurrent Network Structure

Hochreiter and Schmidhuber proposed the long short-term memory (LSTM) model to overcome these weaknesses. It uses four control gates and cell states to manage the flows of information. Input and forget gates decide whether new information will be added or whether old information should be forgotten. The candidate gate decides how much to update the cell state, while the output gate decides whether to add the current information to the output and the next update.

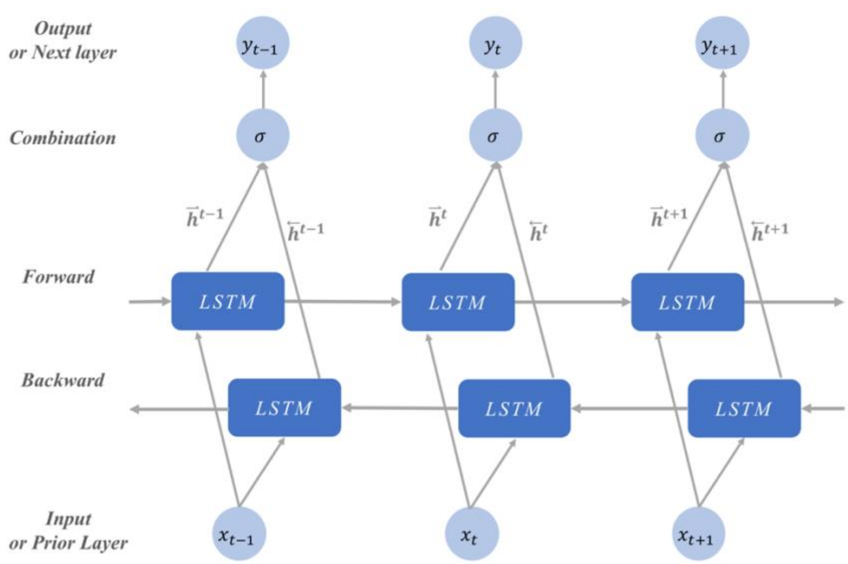


Graph 4. Long Short-Term Memory Network Structure

In this paper, the LSTM models have two layers with 128 and 64 units and a densely connected layer with softmax activation for output. Stacked LSTM is implemented here to capture the rich information in the posts because RNN is deep in time by its nature but not deep in terms of abstraction level. An additional layer in an LSTM model will utilize the representation from the previous layer to create a representation of higher abstraction levels.

C3. Bi-directional Long Short Term Memory (BiLSTM)

Long Short-Term Memory model can preserve information in earlier positions in a sequence and use it in the processing of later positions. However, some sentence structures, such as rhetorical questions, can only be properly understood once the reader reaches the end of a sentence. When reading posts on the stock message board, one often needs to read the whole post before understanding the author's opinion. This fact motivates the use of the Bi-directional Long Short-Term Memory model. In a Bi-LSTM layer, each sequence from the training set will be fed into two sub-layers, one in the forward order, and the other in backward order. This ensures that for every position in the sequence, the model has full information of all the positions before and after it.



Graph 5. Bi-directional LSTM Network Structure

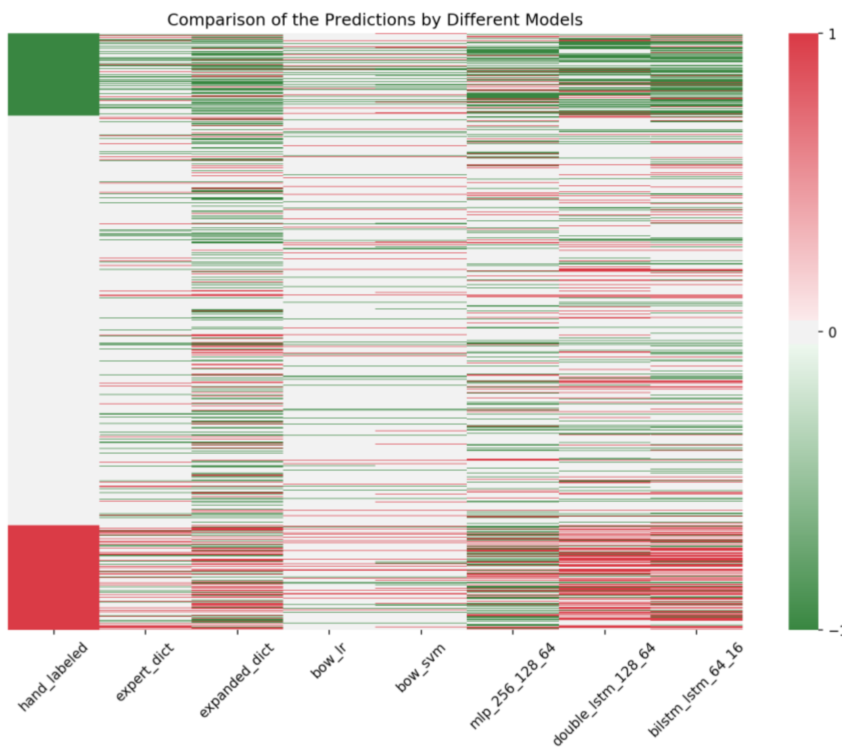
V. Results and Discussions

The models listed above are trained on the over-sampled, class-balanced training set. The trained models are evaluated with the out-of-sample test set. The prediction accuracies are reported in the table below. It is worth noting that the dictionary-based models already reached an accuracy of 60%, which is a decent level for three-class classification problems. The traditional machine learning based-models consistently outperformed the dictionary-based models; the deep learning models, except for Multilayer Perceptron, produced the best predictions. The Bi-directional LSTM model had a slight edge over the stacked LSTM model and achieved the highest accuracy of 72%.

Table 1. Accuracy of Different Models on Test Set

Model	**Accuracy**
Expert dictionary	0.607
Expanded dictionary	0.541
Logistic Regression	0.671
Support Vector Machines	0.669
Multilayer Perceptron	0.63
Stacked LSTM	0.71
Bi-directional LSTM	**0.72 ***

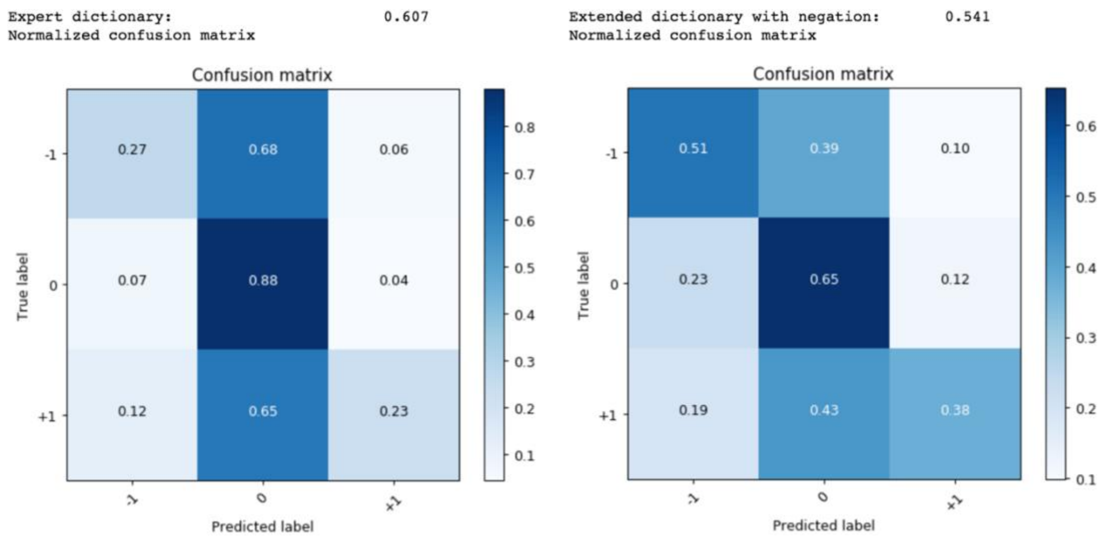
What the accuracies could not show, however, is the prediction preference of each model. Note that, similar to many posts on stock message boards, the posts in the test set are imbalanced in terms of classes. The majority of the posts are neutral in sentiment, which means they are irrelevant or contain positive and negative views that cancel out. A hypothetical classifier that predicts every post to be neutral can already have high accuracy. This special feature of the problem forces us to examine the confusion matrices of different models and compare them in more detail.



Graph 6. A visual representation of prediction results by different models

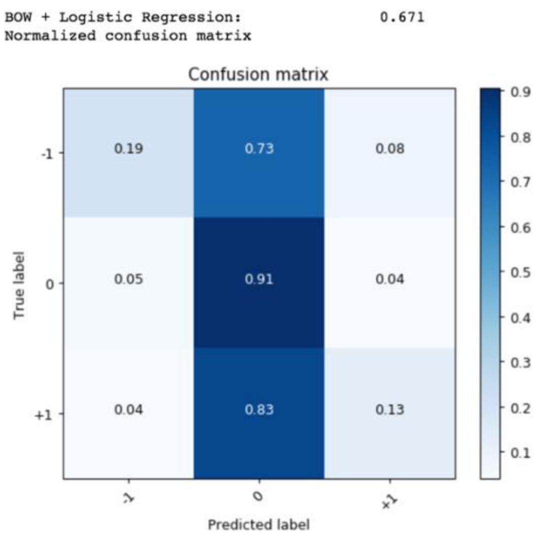
A) Dictionary based models

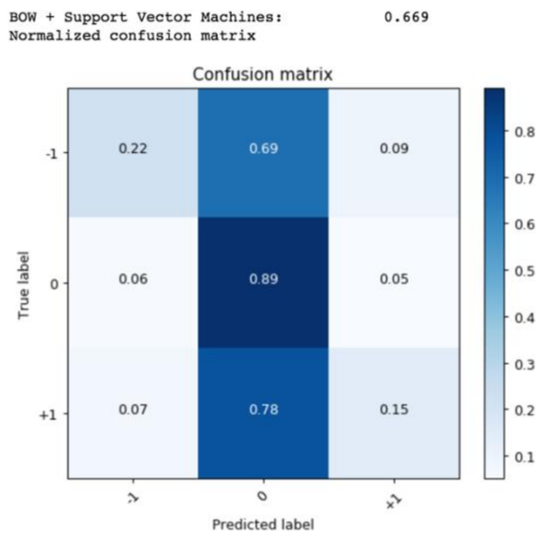
The expert dictionary model classified most neutral cases correctly. It is also more conservative in that it made few directional predictions. In contrast, the expanded dictionary model is more aggressive in giving directional prediction and it was able to predict 51% of the negative cases and 38% of the positive cases correctly. This pattern might be ascribed to the negation mechanism encoded in its algorithm. However, the overall accuracy of the expanded dictionary model is not satisfactory.



B) Machine learning-based models

Despite the high accuracy, the two machine learning-based models are even more conservative than dictionary-based ones. They can predict neutral cases with high accuracy and rarely make wrong directional predictions (e.g. predicting positive when the post is actually negative). However, such classifiers might be useful in practical applications as they make so few directional sentiment predictions. Systematic trading firms cannot generate enough signals for building effective strategies. Regulators could not perceive the changes in public sentiment as most posts are labeled as neutral. The fact that these two models mostly make neutral predictions is surprising at first, but there might be a methodological explanation. Since the posts are transformed into numerical format using a bag-of-words approach, all positional information in the posts is lost in the first step. Thus, the classifiers cannot capture sentiment inversion with negation words, let alone rhetorical questions and connotations. Only posts whose author clearly indicate their attitudes will be picked up by the classifiers, resulting in conservative predictions.

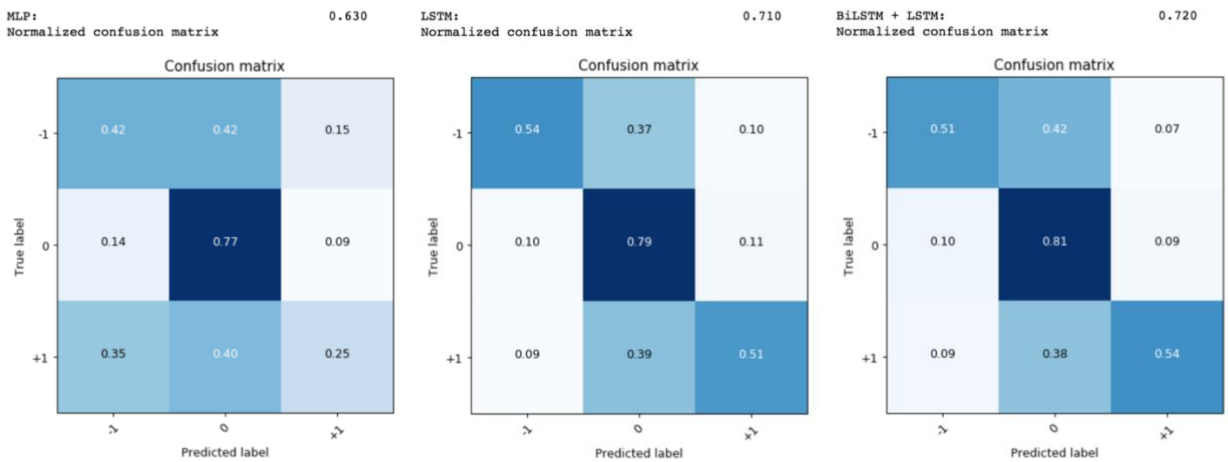




C) Deep learning-based models

Among the three deep learning models, the multilayer perceptron model is inferior to the other two. However, it still beats the dictionary-based models in terms of accuracy, which, to an extent, verifies the idea that a feed-forward neural network that is deep and wide enough can approximate almost any functions. Nonetheless, the difference between MLP and RNN models demonstrate the strength of RNN, especially LSTM, in modeling sequential data.

The confusion matrices of stacked LSTM and Bi-directional LSTM are similar, with the latter showing slightly higher accuracy. They consistently predict neutral posts correctly and are flexible enough to predict directional sentiments. Note that the shape of the confusion matrix for the LSTM models resembles that of the expanded dictionary model. One interpretation is that the LSTM models automatically learned the sentence structures and word connotations that were previously hand-coded, and went on to learn more rules that are not obvious to humans.

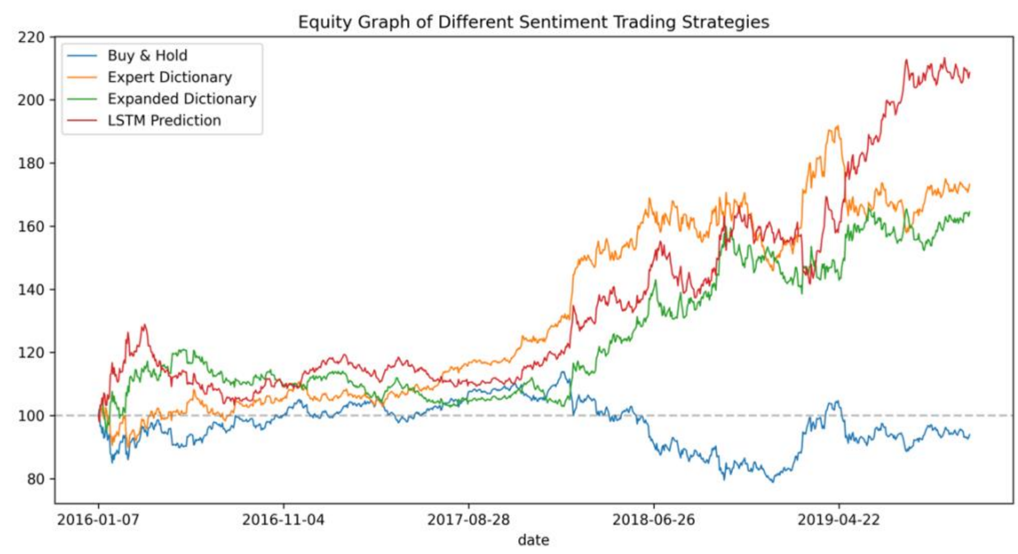


Finally, to evaluate the practical usefulness of the proposed deep learning-based model, the best-performing Bi-directional LSTM model is applied to build a systematic trading strategy. More specifically, the trained deep learning sentiment classifier is used to predict the sentiment labels of 3 million posts posted after the period of the training set. Then, an average daily sentiment of the posts is calculated for all trading days and used as trading signals: buy, sell, or hold the SSE index.

Several baseline strategies are created for comparison. One strategy is to buy and hold the SSE index throughout the period. Two more strategies are based on the sentiment labels created by the expert dictionary model and the expanded dictionary model. The simulation shows that the strategy based on Bi-directional LSTM has the highest Sharpe Ratio, a risk-adjusted measure of investment returns, followed by the expanded dictionary strategy. The equity graphs of strategies are plotted together.

Table 2. Sharpe Ratio of Different Sentiment Trading Strategies

Strategy	**Sharpe Ratio**
Buy and Hold	-0.305
Expert Dictionary Sentiment	0.607
Expanded dictionary Sentiment	0.737
Bi-directional LSTM Sentiment	**1.102 ***



VI. Conclusion

Extracting sentiment from stock message board posts is a challenging task. This paper reviews the history of stock message board research with a focus on sentiment extraction techniques. Seven post sentiment prediction models are then carefully constructed based on common practices in prior studies and lastest development in computer science literature. Specifically, dictionary-based models, traditional machine learning-based models, and deep learning models are compared and contrasted using prediction accuracies and their respective confusion matrix. The characteristics of each model are discussed with respect to their fitness in predicting sentiment in Chinese financial texts.

The Bi-directional LSTM model turns to be the best performer among the models tested, both in terms of accuracy and performance in trading simulations. The superior performance could be ascribed to the adoption of finance domain-specific word embeddings and the structure of Long Short Term Memory RNN, which specializes in extracting information from long complex texts. This paper also reflects on the drawbacks of applying traditional NLP and machine learning techniques on this sentiment prediction task, especially the fact classifiers can be content with predicting neutral cases and be too conservative about making directional predictions. In contrast, the RNN models are less prone to such issues.

This study also has several limitations. First, a larger dataset covering a longer period will improve the accuracy and robustness of most models tested in this paper, especially for the deep learning models. Human-generated sentiment labels are hard to procure and inherently noisy, crowdsourcing the task on a larger scale will be preferable in future studies. Second, the models examined in this study are standard and simplistic in their structures. More customizations and modifications can be made to the models to boost prediction performance, and there is a large room for hyperparameter tuning for the deep learning models. Thirdly, given the vast quantity of posts available, unsupervised or semi-supervised methods should be explored and compared with the supervised approaches. The metadata and social network statistics associated with the posts can also serve as inputs for sentiment prediction algorithms.

Finally, by applying deep learning to the automatic sentiment labeling of Chinese stock message board posts, this paper hopes to open up new research opportunities in social network analysis and behavioral finance, as well as advancing the relevant literature in the context of China.

References

Antweiler, Werner and Murray Z. Frank. "Is All That Talk Just Noise? The Information Content of Internet Stock Message Boards." *The Journal of Finance*, vol. 59, no. 3, 2004, pp. 1259–1294. *JSTOR*, www.jstor.org/stable/3694736.

Banghui, Luo, et al. "Emotion space model for classifying opinions in stock message board." *Expert Systems with Applications*, vol 44, 2016, pp. 138-146, ISSN 0957-4174, https://doi.org/10.1016/j.eswa.2015.08.023.

Huang, Runpeng, et al. "Stock Market Prediction Based on Microblog Sentiment Informatoin." *Journal of Industrial Engineering/Engineering Management*. vol. 29, no. 1. DOI: 10.13587/j.cnki.jieem.2015.01.006.

Li, Jiahong, et al. "Sentiment-aware stock market prediction: A deep learning method." International Conference on Service Systems and Service Management, Dalian, 2017, pp. 1-6.

Jin, Xuejun, et al. "The Impact of Internet Media on Stock Market: Empirical Study on Eastmoney Guba." *Research on News and Media*. vol. 12, 2013.

Nguyen, Thien Hai and Shirai, Kiyoaki. "Topic Modeling based Sentiment Analysis on Social Media for Stock Market Prediction." Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing 2015. July 2015, http://dx.doi.org/10.3115/v1/P15-113.

---, Velcin, Julien. "Sentiment analysis on social media for stock movement prediction." *Expert Systems with Applications*, vol. 42, issue 24, November 2015, pp. 9603-9611, ISSN 0957-4174, https://doi.org/10.1016/j.eswa.2015.07.052.

Sabherwal, Sanjiv, et al. "Do Internet Stock Message Boards Influence Trading? Evidence from Heavily Discussed Stocks with No Fundamental News." *Journal of Business Finance & Accounting*, vol. 38, 2011, pp. 1209-1237. doi:10.1111/j.1468-5957.2011.02258.x.

Takahashi S., et al. "Analysis of the Relation Between Stock Price Returns and Headline News Using Text Categorization." *Knowledge-Based Intelligent Information and Engineering Systems*, vol. 4693, 2007. Springer, Berlin, Heidelberg.

Tumarkin, R. and Whitelaw, F. "News or Noise? Internet Postings and Stock Prices." *Financial Analysts Journal*, vol. 57, issue 3, 2001, pp. 41-51, DOI: 10.2469/faj.v57.n3.2449.

Wysocki, Peter D. "Cheap Talk on the Web: The Determinants of Postings on Stock Message Boards." University of Michigan Business School Working Paper No. 98025, November 1998, SSRN, DOI: 10.2139/ssrn.160170.

Yi, Hongbo, et al. "The Impact of the Different Investor Sentiment in Internet Forums on Stock Market." *Collected Essays on Finance and Economics*, no. 1, Jan. 2015, DOI:10.13762/j.cnki.cjlc.2015.01.007.

Zhang, Xindong and Yuan, Dongliang. "Research on the Impact of Investor Sentiment on Stock Market Based on Micro-blog. " *Journal of Intelligence*, vol. 36, no. 8, 2017, DOI: 10.3969/j.issn.1002-1965.2017.08.015.

Zhang, Y., et al. "Measuring effects on stock returns of sentiment indexes created from sock message boards." *Journal of Financial Research*, vol 35, 2012, pp. 79-114. doi:10.1111/j.1475-6803.2011.01310.x.