Sentiment Analysis for Financial News

Aaron Tian
aarontian426@gmail.com
University of British Columbia

Ruby Nguyen
nrubyn22@gmail.com
University of British Columbia

Nicholas Jay Sanders
nicholasjsanders@outlook.com
University of British Columbia

***Abstract*--Stock prices are driven by many factors and one of them is market psychology which refers to the sentiment of financial market participants. Market sentiment can be largely influenced by news, therefore, having the ability to analyze breaking news in the matter of seconds can support investors tremendously in making decisions. In this paper, we applied variants of Bidirectional Encoder Representations from Transformers (BERT) and text-based Convolutional Neural Networks (CNN's) to perform sentiment analysis of financial news articles in order to provide valuable information for decision making in the stock market. BERT has been pre-trained on a large text corpus for a self-learning task. We use the dataset from Matheus Gomes de Sousa et. al [1] which contains 582 manually labelled news articles as positive and negative. We apply a BERT-Base model on this dataset and obtain a F-score of 82%.**

I. INTRODUCTION:

Extracting sentiment from financial news articles or social media posts can have great impacts not only on analyzing historical events within markets, but also could potentially be used for future speculation and prediction of the market behavior. News and public opinion can potentially consist of many relevant data for both analysis and prediction, which makes sentiment analysis of these texts an ideal starting task for future downstream tasks.

Sentiment analysis of news within the financial domain is not a novel idea. Our goal is to explore and improve methodologies in modeling and hopefully gain new understandings about this task. This means that we reviewed previous studies and then synthesized those findings with our own ideas into our final approaches. Our finished product is an improved, more robust model in predicting sentiment from financial textual data, achieving both of these goals.

II. RELATED WORK:

Our primary source was Matheus Gomes de Sousa et. al [1], where the authors explored several different models on classifying sentiment of news texts (using the same data set as us). The models attempted were BERT-base, Naive Bayes bow, SVM bow, Naive Bayes tfidf, SVM tfidf, and a text CNN. The authors chose to employ BERT-base instead of BERT-large due to lack of computational resources; which is an identical issue that we had in our approach that we overcame. All the models performed relatively subpar (roughly ranging in f1 macro scores of between 0.5 and 0.6), however the BERT-base model performed significantly better than all of the other models (achieving an f1 macro score of 0.73). BERT-base is pretrained on a massive amount of generalistic text and in this case fine tuned on the financial news data to fit the domain and task better.

The authors in Matheus Gomes de Sousa et. al [1] were able to associate historical increases and decreases in the DJI stock market value, and concluded "in the analyzed period, 69% of the periods between opening and closing the stock market, the sentiment of the news was consistent with the stock exchange variation. However, the collection period was short, and more extended periods must be

evaluated to verify if the observed behaviour is significant." These findings both support our chosen methodology, data set, and motivations.

In Shapiro, Adam Hale, Moritz Sudhof, Daniel Wilson [2] the authors explored and developed lexical-based sentiment classifiers, where the data consists of words with associated sentiment scores, rather than example texts. This lexical-based computational methodology was proposed by them as an alternative to traditional surveys that government agencies and financial institutes normally use to capture sentiment. The authors utilized a corpus that is quite large in comparison to other studies, consisting of 238,685 news papers from 16 different sources, between 1980 and 2015; within this dataset, 800 news articles were manually annotated for sentiment by research assistants. Several different lexicons were examined and analyzed, leaving the authors to conclude that it was best to actually just combine lexicons together and test the different combinations. Additionally, instead of classifying texts as negative, positive, or neutral, the authors used an ordinal classification between 1 and 5 (1 being very negative, 3 being neutral, and 5 being very positive). This ordinal multi-class classification task is much more difficult than a binary classification task, such as the one described previously in Matheus Gomes de Sousa et. al [1]. After determining their best lexical model, the authors compare their performance with off the shelf GloVe and BERT models, where GloVe performed worse but BERT performed similar to their most robust lexical model. The authors argued that lexical models should be preferable in this case because they are easier to interpret for their analysis. Relevance of these findings was further analyzed while echoing many cited papers with similar findings, that sentiment of news articles possesses correlation with consumer and market behavioral trends.

In the classic paper, Huina Mao, Scott Counts, Johan Bollen [3], the authors here explore several data sets from various sources (Twitter feeds, news headlines, financial data, and Google search queries) as well as various methods and modeling techniques. However, the purpose of this paper is mainly to analyze and discover whether sentiment analysis is a good tool for predicting market behavior. Additionally, just like the paper above, the authors were seeking to find better alternatives to slow and expensive surveying methods for capturing sentiment. The news data used was from 8 different sources (Wall Street Journal, Bloomberg, Forbes.com, ReutersBusiness & Finance, BusinessWeek, Financial Times, CNN-Money and CNBC), and scraped news headlines from the sources' twitter pages. For the search engine queries, the authors obtained weekly search volume data from Google Insights for specific seed queries (dow jones, stock market, stock to buy, stock, bullish, bearish, financial news and wall street). For social media data, the authors randomly sample public Twitter posts and categorize them based on the appearance of words within the tweet (i.e. if bullish is in the text, the text is bullish). To track the relevant market state, the authors downloaded daily and weekly data of DJI average, trading volume, and volatility from Yahoo Finance. Correlation analysis, Granger Causality Analysis, Multiple Regression Analysis, and Forecasting analysis were all used to examine the relationships between temporal market behavior and temporal sentiment. The findings and conclusion implicate that sentiment analysis is still the best discovered

methodology for predicting market behavior, as well as that not just surveys and news articles are good data sources for sentiment analysis.

III. DATASET:

We adopted the dataset used by Matheus Gomes de Sousa et. al [1]. This dataset is composed of financial news texts and headlines collected from CNBC, Bloomberg, Business Insider, New York Times, and Forbes. The corpus contains only English. The genre is news media, which includes both formal and informal language. The datasets consist of 582 financial news examples, each of the examples includes a headline of the news (headlineTitle) and a short summary of that piece of news (headlineText). All of the examples in the datasets are already annotated for sentiment orientation (-1 for negative, 1 for positive) manually by researchers. 62% of the examples in the corpus are of negative sentiment orientation and 38% of them are of positive sentiment orientation. The length of headlineTitle is in the range of 3 and 19 tokens. The average length for headlineTitle is 10 tokens. The length of headlineText is in the range of 10 and 52 tokens. And the average length of headlineText is 20 tokens. We divided the dataset into train, development and test sets by 70%, 15%, and 15%.

IV. METHODS:

The methods we used for this project includes BERT (Bidirectional Encoder Representation from Transformers) from J.Devlin et. al [4], and CNN (Convolutional Neural Network).

The hyperparameters setup for the both BERT-Large and BERT-Base are as follows: batch size is set to be 32, learning rate is set to be 2e-5, max_grad_norm is set to be 1.0, the epoch number is set to be 3 for BERT-Large and 10 for BERT-Base, warmup_proportion is set to be 0.1. For optimizer we use AdamW and for loss function we use Cross Entropy.

The hyperparameters for the CNN model were listed below: batch size is set to be 32, embedding dimension is set to be 300, number of kernels is set to be 32, dropout ratio is set to be 0.5, learning rate to be 0.01, the number of epoch is set to be 15, and the region sizes are set to be 2,3,4. For optimizer we use Adam and for loss function we use Cross Entropy.

V. EXPERIMENTS:

We built a baseline model using BERT and logistic regression first. We didn't fine-tune the BERT part of the model at this stage, instead, we used the pre-trained BERT to process the input sentences and pass along the last hidden state of the first token ([CLS]) of each sentence onto the logistic regression model. Using the output of BERT, the logistic regression model classifies each sentence as either positive or negative. The f-score on the validation set was around 0.63 and accuracy is around 0.73. Since Matheus Gomes de Sousa et. al [1] achieved an f-score of 0.73, we decided to fine-tune the BERT model to achieve better classification accuracy.

We used a fine-tuned BERT model to run our data with hyperparameters described in the method section. Since our dataset is quite small, we decided to combine the headline text with headline title to double the data size. And because the average length of headline text is 28 tokens and the average length of headline title is 10 tokens, we restrained sequence max length to 32. This could help us get most of the information we need while avoiding too

many paddings, which could negatively influence the classification accuracy. We divided the dataset into train, development, and test sets, and ran both BERT-base and BERT-Large. The largest macro f1 score we got on the development set using BERT-Large is around 0.81, and the largest macro f1 score using BERT-base is around 0.83. Both versions of the fine-tuned BERT achieved higher scores compared to the model using the combination of BERT without fine-tuning and logistic regression. In addition, we also built a CNN model for this task using the hyperparameters described in the methods section. The largest f1 score we got on development set from this CNN model is 0.65. This result proved our assumption that BERT could be more suitable for this relatively small dataset since CNN will need more data to deliver decent classification accuracy.

Considering the way we doubled the dataset, we realized that there could be some similarities between the texts and headlines within each example. We ran some analysis and found that the average token overlapping ratio is around 20% for all the headlines. This could possibly make our model overfit since it could just memorize a pattern from an input and give the same sentiment polarity when it is fed a similar sentence, and more importantly, the actual performance of the model may not be as good as it showed. Therefore, we ran a separate experiment solely on headline text and implemented both Bert-Large and Bert-Base models with the same set of hyperparameters except setting batch size to be 16 for Bert-Large to prevent CUDA out-of-memory error. Bert-Large generated a macro f1 score of 0.81, and Bert-Base generated 0.75 for the development set. We also ran the CNN model with only headline text, this time the largest f1 score is around 0.66, which is the same as the result of the combined dataset. This showed that splitting and combining data only gives a small improvement compared with non-splitting data for both models.

We applied the best performing model on the development set, the BERT-base model, on the test set, using the same hyperparameters during the training phase, the f1 score we got is 0.82.

VI. RESULTS:

Baseline models built initially

Metric	Scikit Learn Dummy Classifier	BERT + Logistic Regression
Test f1-score	0.56	0.63

The test results on our initial models were less desirable because they did not improve on our primary source, Matheus Gomes de Sousa et. al [1], which achieved a macro f1 score of 0.73. This gave us an understanding of how much we would need to improve our performance with regard to this initial set of results. The dummy classifier was expected to do about as well as guessing, which implies that BERT + logistic regression does not actually fit the data very well in this case.

Models trained on headline text only

Metric	BERT-base	BERT-large	Text CNN
Validation f1-score	0.75	0.81	0.66
Test f1-score	0.81	0.78	0.58

The model trained on headline text only achieves a significant improvement in comparison to our original scores. Only the Text CNN did not do better than the best model achieved in Matheus Gomes de Sousa et. al [1]. Surprisingly, BERT-base does better than BERT-large here as we can train on 15 epochs instead of 3 epochs.

Models trained on combined data

Metric	BERT-base	BERT-large	Text CNN
Validation f1-score	0.83	0.81	0.65
Test f1-score	0.82	0.77	0.62

The model trained on the combined dataset improved the macro f1 score significantly from our initial model from 0.63 to 0.82. Essentially doubling our data set by splitting up the data proves to only improve performance minimally which could perhaps be a result of the previously mentioned token overlap between text and headlines. Even though the data set is doubled here, it is still possible that we could get further performance increases with many more added examples.

VII. CONCLUSION:

We were able to achieve better performances in comparison to our initial models and Matheus Gomes de Sousa et. al [1] because we maximized our models' potentials with the available computational resources and doubled the data by splitting up the data by treating headlines and texts as separate examples. Despite the challenges of dealing with similar computational constraints as Matheus Gomes de Sousa et. al [1] and strenuous time constraints, we demonstrate that there are further improvements in methodology that can be pursued.

In comparison to Shapiro, Adam Hale, Moritz Sudhof, Daniel Wilson [2], we find that neural models can reasonably be improved on this task and explored further. Although we made an improvement on the best model established by Matheus Gomes de Sousa et. al [1], we suggest that our current best model can potentially be further improved with the addition of more training data, a further pretrained language model specific to the financial news domain, multi-task learning (such as models used for opinion detection), generating augmented data from out smaller data set, or perhaps pre-training on silver data that's been automatically generated on unlabeled data. Lexical-based models and other deep learning models, such as XLNet, could be attempted as well, however other deep learning models may have similar difficulties that we encountered.

REFERENCES:

[1] M. Sousa, K. Sakiyama, L. Rodigues, P. Moraes, E. Fernandes, and E. Matsubara"BERT for Stock Market Sentiment Analysis", 2019.

[2] Shapiro, A. Hale, M. Sudhof, and D. Wilson "Measuring News Sentiment", 2017.

[3] H. Mao, S. Counts, Microsoft Research, and J. Bollen "Predicting Financial Markets: Comparing Survey, News, Twitter and Search Engine Data", 2011.

[4] J. Devlin, M. Chang, K. Lee, and K. Toutanova, Google AI Language "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding", 2018