STOCK PREDICTION USING SENTIMENT ANALYSIS

Efrain Galarza **Vijay Gandhi** **Durga Kasireddy**

INTRODUCTION:

The stock market investment plays a key role in the finance sector. In this paper, we consider sentiment to be an important factor on stock price movements. Sentiment of investors and traders can help us determine whether stock prices will go up or down. Therefore, analyzing information in real time can help us predict these price variations. To achieve this, we go through news articles that are relevant to the economy and to the stock market in order to determine the sentiment and to predict the stock price movements with our model. The Bidirectional Encoder Representations from Transformers (BERT) deep learning model is used to predict the sentiment of finance news articles.

LITERATURE SURVEY:

In the paper *"On the Importance of Text Analysis for Stock Price Prediction",* they generated 21 numeric features based on four non-linguistic feature types which were Earnings Surprise, Recent Movements, Volatility S&P 500 Index and Event Category. These features captured changes between stock moving averages over multiple time intervals: 1 month, using the 5 days moving average; 1 quarter, using the 10 days moving average; and 1 year, using the 20 days moving average. The stock price movements for a company were normalized with the change of the S&P 500 index in the same period within the same moving average windows. For the linguistic features, they used unigram features and they applied non-negative matrix factorization (NMF). The resulting vector, along with the baseline numeric features were used in a random forest classifier. The accuracy obtained with this set of features got up to around 55% and they mentioned that adding sentiment features did not improve their results.

In *"Real-Time Sentiment Analysis of Twitter Streaming data for Stock Prediction",* they performed Twitter sentiment analysis by using RNN components in Stanford Core NLP, which is a natural language processing software that allows extraction of sentiment from text classifying text into three classes (positive, negative or neutral). They tried to predict the future stock prices of Google, Microsoft and Apple after collecting actual stock prices for these companies and 56,000 tweets that involved the ticker keyword for each company. High correlations were found between the predicted and real prices for these stocks.

In "*A Novel Twitter Sentiment Analysis Model with Baseline Correlation for Financial Market Prediction with Improved Efficiency*", polynomial regression, classification modelling and lexicon-based sentiment analysis were performed. A lexicon-based sentiment analysis method was used as the sentiment classification model as it saves time that one would normally take labelling training sets and it also prevents overfitting, which is usually present in machine learning sentiment analysis. However, it is mentioned in the paper that it is still difficult to define a "language sentiment" as words or phrases can be interpreted as both positive and negative in different contexts. Their model predicts the future stock market trend with an accuracy of 67.22%.

DATASET DESCRIPTIONS:

For training of the model:

The BERT model is pre-trained on a huge amount of information, but we fine-tune it using a dataset with finance news articles and their sentiment labels, which can be positive, neutral, or negative. Fine tuning the model is important because some words in finance may have different meanings to other contexts. For example, the terms bull or bear markets refer to rising and declining markets, respectively. This dataset was obtained from Kaggle[1] and it was originally used in the research paper "Good Debt or Bad Debt: Detecting Semantic Orientations in Economic Texts" . It has 4837 finance news articles labeled with their sentiment. These news articles were obtained from LexisNexis using an automated web scraper. Unfortunately, the date range for this dataset is not mentioned in the original publication.

For testing of the model:

After our BERT model has been trained with the stock news articles dataset, it is time to test it. Originally, we intended to use tweets and news articles to build our test dataset. However, we could not find a tweets dataset with enough data to be used in our project. For this reason, we just used news articles.

For testing, we used a different dataset, which included news articles for a specific time period. The news articles came from another dataset from Kaggle[2] and it included summaries, abstracts and snippets from news related to Apple from 2006 to 2016. The news information was obtained from The New York Times API. We fed this dataset into our BERT model to get their sentiment. This sentiment will be compared later against whether Apple's stock price went up, remained the same, or went down for the same day, to see if we can use the sentiment as a good predictor for

[1] https://www.kaggle.com/ankurzing/sentiment-analysis-for-financial-news
[2] https://www.kaggle.com/BidecInnovations/stock-price-and-news-realted-to-it

stock price movements. The labels, to which the sentiment will be compared to, were obtained from a different dataset, which is a stock dataset that we built from information obtained out of Yahoo Finance. This dataset was within the same date range as the previous dataset, 2006 to 2016, and it included daily stock information of Apple and the S&P 500 index. From here, we took Apple's close and open prices and subtracted them, to calculate the daily percent change in price. After that, we normalized this value by subtracting the same percent change computed for the entire S&P 500 index. For example, if Apple's stock price went up 4% and the S&P 500 went up 1% that day, then the normalized value would be 3%. The normalized change is then binned into one of two labels: positive, if it is greater or equal to 0%, or negative, if it is less than 0. These labels will be compared to the sentiment labels to evaluate our model's prediction accuracy.

Database and dashboard:

The news articles and stock information are stored in the ElasticSearch database. The Kibana dashboard is integrated with the database to visualize the results from the information stored in the database. A pie-chart can be plotted for the sentiment of the news articles for a specific date range. Final sentiment of the stock is predicted as the majority sentiment in the pie-chart for the same time period.

Searched 6 of 6 shards. 1641 hits. 0.007 seconds

_index	_type	_id	_score ▲	author	location	language	friends	followers	statuses	date	message
stocksight	tweet	grfx2HUByfQTur8ZWlSk	1	HeathMontgomery	Dixie County, FL	None	699	1324	30993	2020-11-18T01:21:19	@banny_e: I am certain that this is an artificial intelligence robot that Elon
stocksight	tweet	g7fy2HUByfQTur8Z1lTw	1	iam_bolajistar	Minna, Nigeria	None	1639	461	491	2020-11-18T01:23:08	@DaddyFRZ: Elon musk does not even believe God exists yet he is on th
stocksight	tweet	hLfy2HUByfQTur8Z4lSC	1	AK9dj	None	None	632	79	2981	2020-11-18T01:23:10	If I got a chance to Elon Musk's AMA, I will definitely ask him.
stocksight	tweet	hbfy2HUByfQTur8Z61QU	1	Cosmic_Penguin	Hong Kong	None	4025	5293	29761	2020-11-18T01:23:10	@nasahqphoto: Check out all of the pictures made by the NASA HQ photo
stocksight	tweet	hrfy2HUByfQTur8Z-1RF	1	1965Samran	None	None	4330	269	9775	2020-11-18T01:23:14	@Miketrillaa: This why I'm glad I got a Tesla. No hands in the wheel I can
stocksight	tweet	h7fz2HUByfQTur8ZDVTf	1	Kings58583491	None	None	42	2	1734	2020-11-18T01:23:23	@OANN: SpaceX spacecraft successfully docks at ISS - #OANN
stocksight	tweet	iLfz2HUByfQTur8ZGlQ_	1	rhondagilr	Midwest	None	4657	2172	42280	2020-11-18T01:23:23	@ridethenews: Reporting from Tuesday on Kamala + Lindsey, troops, Tes
stocksight	tweet	ibfz2HUByfQTur8ZLFQN	1	gamegrumpsnsp	Victoria, Australia (15/2/19)	None	793	465	28741	2020-11-18T01:23:29	just pulled into a parking lot with designated tesla charging spots we d
stocksight	tweet	irfz2HUByfQTur8ZPFQa	1	Artificialbra1n	None	None	6	479	76558	2020-11-18T01:23:35	@AndyvermautP: The forgotten interview between Elon Musk Bill Gates,
stocksight	tweet	i7fz2HUByfQTur8ZSFSQ	1	JCHovis	Kansas City, MO	None	428	102	1986	2020-11-18T01:23:35	@elonmusk @peterrhague When the ISS gets retired SpaceX should brin
stocksight	tweet	jLfz2HUByfQTur8ZVFTz	1	Jyrice	New York, NY	None	1042	814	6730	2020-11-18T01:23:37	@NASA: The hatches are open and NASA's @SpaceX Crew-1 astronauts
stocksight	tweet	jbfz2HUByfQTur8ZXlSr	1	stonk_the	None	None	288	74	12373	2020-11-18T01:23:41	@Teslarati: Tesla Gigafactory Texas raises its 1st pillar as site shows imm

Figure 1: ElasticSearch database that stores the news articles. Message column is the news article headline.

Searched 6 of 6 shards. 1641 hits. 0.007 seconds

ses	date	message	tweet_id	polarity	subjectivity	sentiment
	2020-11-18T01:21:19	@banny_e: I am certain that this is an artificial intelligence robot that Elon Musk warned us about quoting bland 2012 tweets to blend i	1328870870027219000	0.14628650793650794	0.8015873015873015	neutral
	2020-11-18T01:23:08	@DaddyFRZ: Elon musk does not even believe God exists yet he is on the way to becoming the third richest man on earth. - When pastors te	1328871330092085200	0.302	0.42500000000000004	positive
	2020-11-18T01:23:10	If I got a chance to Elon Musk's AMA, I will definitely ask him.	1328871336387637200	0.28595	0.5	neutral
	2020-11-18T01:23:10	@nasahqphoto: Check out all of the pictures made by the NASA HQ photo team of @NASA's @SpaceX Crew-1 launch from @NASAKennedy! From crew	1328871337050312700	0	0	neutral
	2020-11-18T01:23:14	@Miketrillaa: This why I'm glad I got a Tesla. No hands in the wheel I can sit back and chill. Who road has game strong?	1328871353097732000	0.37911666666666666	0.5333333333333333	positive
	2020-11-18T01:23:23	@OANN: SpaceX spacecraft successfully docks at ISS - #OANN	1328871391383445500	0.62195	0.95	positive
	2020-11-18T01:23:23	@ridethenews: Reporting from Tuesday on Kamala + Lindsey, troops, Tesla	1328871392356552700	-0.17	0	neutral
	2020-11-18T01:23:29	just pulled into a parking lot with designated tesla charging spots we don't belong here lmao	1328871416591044600	0.40115	1	positive
	2020-11-18T01:23:35	@AndyvermautP: The forgotten interview between Elon Musk Bill Gates, A super informative discussion on the Development Of Artificial I	1328871439206899700	0.16273333333333334	0.8333333333333333	neutral
	2020-11-18T01:23:35	@elonmusk @peterrhague When the ISS gets retired SpaceX should bring it back to earth instead of a Viking funeral.	1328871441656402000	-0.1806	0	neutral
	2020-11-18T01:23:37	@NASA: The hatches are open and NASA's @SpaceX Crew-1 astronauts Shannon Walker, @Astro_Soichi, @AstroVicGlover, and @Astro_illini are t	1328871449612996600	0	0.5	neutral
	2020-11-18T01:23:41	@Teslarati: Tesla Gigafactory Texas raises its 1st pillar as site shows immediate progress after shifting to 24/7 operations	1328871465714929700	0.21075	0	neutral
	2020-11-18T01:23:41	@NASA: The hatches are open and NASA's @SpaceX Crew-1 astronauts Shannon Walker, @Astro_Soichi, @AstroVicGlover, and @Astro_illini are t	1328871466176307200	0	0.5	neutral
	2020-11-18T01:23:45	@JElvis47276277: @NASA @SpaceX @Astro_Soichi @AstroVicGlover @Astro_illini @Space_Station Hello @NASA....this is real? -	1328871481833611300	0.1	0.30000000000000004	neutral

Figure 2: Sentiment column represents the sentiment of each news article

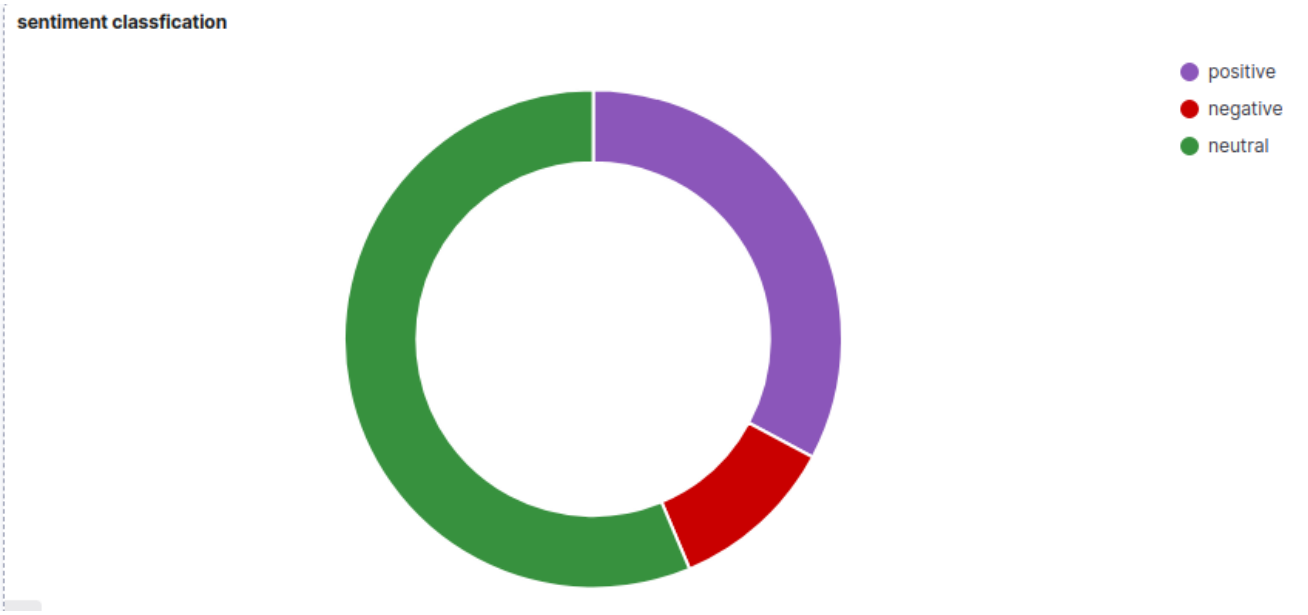


Figure 3: Pie-chart example of sentiment column. Most of the sentiment is “neutral” so the stock price movement for the same time period would be predicted as neutral, meaning a change less than 1% would be expected for the stock price.

FIRST APPROACH: RANDOM FOREST

The first method we tried was to fit a random forest to our news dataset. The reason we first tried random forest as our model was because it contains ensemble of models and it has the power to work well with non-normalized data. The news dataset had to be converted into numbers before passing it into the model. So, we first converted the news documents in numeric form using TF-IDF vectorization. TF is the term frequency which is frequency of a given word in the document divided by the total number of words in the document. IDF is the inverse document frequency which is logarithm of the total number of docs divided by the number of documents containing a

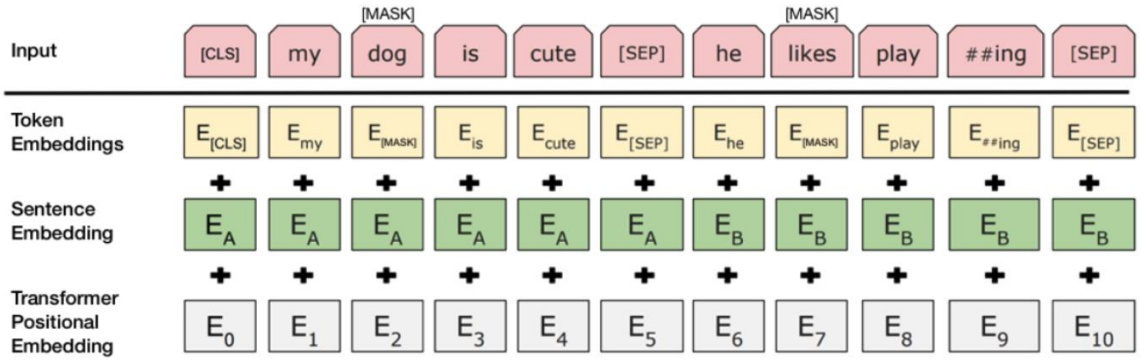
particular word. We thought focusing on the specific key words will help us derive the sentiment out of each news article. However, we were not satisfied with the results obtained. The state-of-the-art models obtained more than 60% accuracy on this kind of problem and the random forest classifier obtained around 35% after some fine tuning. Thus, we wanted to use a huge neural network architecture to capture even better details in the news dataset as features to compensate for the low accuracy score.

SECOND APPROACH: BERT MODEL

Next, we used the standard BERT (Bidirectional Encoder Representations from Transformers) base cased model from the transformers hugging face library, which has around 110M+ parameters. BERT makes use of Transformer, an attention mechanism that learns contextual relations between words (or sub-words) in a text. In its vanilla form, Transformer includes two separate mechanisms — an encoder that reads the text input and a decoder that produces a prediction for the task.

Transformer is an architecture for transforming one sequence into another one with the help of two parts (Encoder and Decoder), but it differs from the previously described/existing sequence-to-sequence models because it does not imply any Recurrent Networks (GRU, LSTM).

In the BERT training process, the model receives pairs of sentences as input and learns to predict if the second sentence in the pair is the subsequent sentence in the original document. During training, 50% of the inputs are a pair in which the second sentence is the subsequent sentence in the original document, while in the other 50% a random sentence from the corpus is chosen as the second sentence.



Source: BERT [Devlin et al., 2018], with modifications

To help the model distinguish between the two sentences in training, the input is processed in the following way before entering the model. A [CLS] token is inserted at the beginning of the first sentence and a [SEP] token is inserted at the end of each sentence. A sentence embedding indicating Sentence A or Sentence B is added to each token. A positional embedding is added to each token to indicate its position in the sequence.

To predict if the second sentence is indeed connected to the first, the following steps are performed. The entire input sequence goes through the Transformer model and the output of the [CLS] token is transformed into a 2×1 shaped vector, using a simple classification layer (learned matrices of weights and biases). Finally, we calculate the probability of IsNextSequence with softmax.

Data pre-processing:

The tokenizer module converts the input text from our training dataset into tokens. The encode plus module takes tokens as input and outputs input_ids and attention mask which are the two important input data fed to the BERT model. Encode plus is configured to output the length of 160 for input_ids and attention mask. Padding parameter is set to true to get an equal length input_ids and attention_mask for all input text. The DataLoader module is used to split the dataset into batches of size 16.

Training:

The BERT model is trained on the training dataset. Huggingface Transformer's library allows us to train the Pytorch deep learning models. The pre-trained BERT model is downloaded, and fully connected layers are attached at the end for sentiment classification. The model is then fine-tuned meaning only the weight of the fully connected layers are updated whilst the weight of the pre-trained BERT layers is untouched. The encoded feature vector is obtained using BERT layers which is then fed to a fully connected layer for sentiment classification. The model is trained for 10 epochs using AdamW's optimizer with an initial learning rate of 2e-5 which is then gradually reduced during training using a learning rate scheduler.

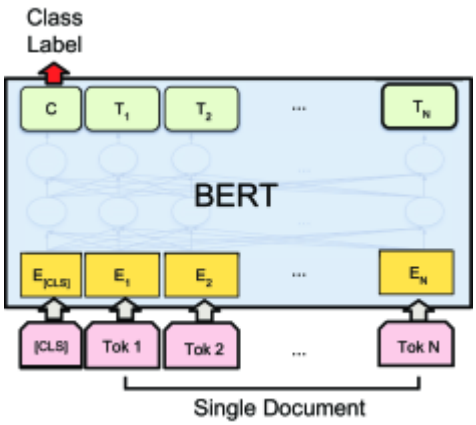


Figure 4: BERT model architecture diagram for sentiment classification.

Testing:

Following the training of the model, we got the sentiments for the Apple news articles dataset from the BERT model. Then we evaluated its performance by comparing these sentiments to the labels obtained from the stock dataset obtained from Yahoo Finance. By doing this, we were able to determine how good the sentiment is as a stock price predictor. In this project, we calculated accuracy for model evaluation.

RESULTS:

The BERT model was fine-tuned on the financial news dataset and the model achieved an accuracy of 78% during training. Next, the model was used on the test dataset to predict the sentiment for Apple news articles on a given day. The predictions were compared against the ground truth labels. In this task, the BERT model achieved an accuracy of 53.8%. Hence, we were able to predict same day stock price movement with an accuracy of 53.8%.

CONCLUSION AND FUTURE WORK:

In this paper we propose a model that can be used to determine the importance of sentiment analysis for stock price movement. The model is based on training the BERT model on a financial news dataset that is later used to predict sentiment based on news articles for a given date. By this method in this case study, we obtained an accuracy of 53.8% when predicting stock price movement.

The biggest limitation of the current approach is using sentiment as the only feature to predict stock price movements, when there are many other factors that have a meaningful impact on the stock market. Therefore, providing more stock information to the model could help us make more accurate predictions. Based on other papers found, incorporating a Long Short-Term Memory

(LSTM) model architecture could potentially give us better results. For example, it can be considered to add features like open price, close price or volume of stock traded to the LSTM, in addition to the sentiment, to make more accurate predictions on whether stock prices will go up or down.

REFERENCES:

Source code for our project can be found here: https://github.com/VIJAYG4/sentiment-analysis

Lee, H., & Surdeanu, M., & MacCartney, B., & Jurafsky, D. (2014). “On the Importance of Text Analysis for Stock Price Prediction.” Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC’14), pp. 1170–75.

Guo, X., & Li, J. (2019). A Novel Twitter Sentiment Analysis Model with Baseline Correlation for Financial Market Prediction with Improved Efficiency. 2019 Sixth International Conference on Social Networks Analysis, Management and Security (SNAMS), Social Networks Analysis, Management and Security (SNAMS), 2019 Sixth International Conference On, 472–477

Das, S., Kumar, R., Kumar, M., Kumar, S. (2018). Real-Time Sentiment Analysis of Twitter Streaming data for Stock Prediction, Procedia Computer Science Volume 132, (pp. 956-964), ISSN 1877-0509

Jacob Devlin, Ming-Wei Chang, Kenton Lee, Kristina Toutanova (2019). BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. arXiv:1810.04805

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, Illia Polosukhin (2017). Attention Is All You Need. arXiv:1706.03762.