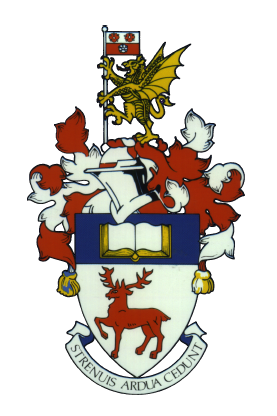
University of Southampton

Faculty of Physical and Applied Sciences

Electronics and Computer Science

**Predicting the Direction of Stock Prices: Utilising Bidirectional Encoder Representations from Transformers (BERT) for Sentiment Analysis**



Author: Roshan Patel

Project Supervisor: Adam Prugel-Bennett

Second Examiner: Geoff Merrett

December 06, 2020

A project report submitted for the award of BSc (Hons) Computer Science

# Abstract

The paper Attention is all you need (2017) [1] utilised an attention mechanism called the transformer to learn contextual relations between words in natural language processing tasks. The result was BERT, a model whose transformer reads entire sequences of words at once, producing benchmark BLEU scores on common translation tasks with minimal training. Despite this, BERT remains an underexplored terrain in the field of stock trend prediction.

Sentiment analysis is a challenging problem in stock trend prediction, and the bidirectional nature of BERT may offer improvements to gauging market sentiment compared to models that do not utilise transformers.

The paper processes a tailored dataset comprised of various sources including news headlines, social media and search engines with different flavours of BERT [2], [3]. The results will be evaluated, and the best model chosen, then refined using statistical market indicators. Models may be improved by establishing new target vectors. The final model’s performance is to be evaluated by back testing over a fixed period using the model as a trading strategy.

# List of Abbreviations

**BERT** = Bidirectional Encoder Representations from Transformers

**NLP** = Natural Language Processing

**DJIA** = Dow Jones Industrial Average

**ML** = Machine Learning

**MSE** = Mean Squared Error

**SGD** = Stochastic Gradient Descent

**SOTA** = State of The Art

**NYSE** = New York Stock Exchange

**NASDAQ** = National Association of Securities Dealers Automated Quotations Exchange

# Chapter 1

# Introduction

## **1.1 Motivation**

Predicting the daily directional changes of a stock price is a problem that many traders face. Aside from fundamental and technical indicators, market sentiment plays a significant role in the price of a stock. Being able to efficiently gauge market sentiment by considering information from a variety of useful sources and applying effective natural language processing techniques to accurately assess sentiment is a challenging problem. In the breakthrough paper *Attention is all you need* (2017) [1], researchers used Bidirectional Encoder Representations from Transformers (BERT) to set new benchmarks for common NLP translation tasks, spending comparatively low training time to achieve favourable performance. Despite this, BERT remains a relatively underexplored terrain in the field of stock trend prediction.

## **1.2 Objectives**

The project will utilise BERT to attempt to form a model that is able to capture market sentiment and will then include other technical indicators to develop a more complete model. The objectives set out for this project are as follows:

1. Data Gathering: Market sentiment data will be web scraped from 3 different sources: social media, news headlines and search engine results. The current aim is to have two types of each source and run scripts daily at 830am EST, capturing 50 results per source. This will be performed continuously over two months resulting in a dataset of 18,000 entries. This is heavily subject to modification and will be influenced by continued research and model performance.
2. Sentiment Analysis: Collected data will be cleaned and pre-processed, after which BERT will be used on Google Cloud Platform (GCP) Compute Engine to produce a vector score for each of the dated string entries. Preliminary tests on a contingency Kaggle data set of similar size took 1.5 hours using a single GPU Tesla P100 16GB.
3. Machine Learning: A model using the vectorised output from BERT will be trained to predict the direction and magnitude of the chosen stock/s closing price for the current day of trading. At this point different flavours of BERT, model tweaking, or additional training may be implemented to try and achieve a favourable performance. Technical indicators such as *simple moving average* will then be included in a new iteration of the model.
4. Comparison and Evaluation: The overall performance of the model will be evaluated. At this point in time three potential metrics have been identified to track overall model performance.
   * 1. The percentage of correct predictions over a fixed N number of days
     2. The Mean Square Error (MSE) of the model prediction compared with the actual price.
     3. The Backtrader Python library which facilitates implementation of a trading strategy to evaluate performance.

# Chapter 2

# Background Theory

## **2.1 The Stock Market**

The stock market is a publicly accessible exchange where buyers and sellers can purchase securities, most commonly in the form of common shares, which represent part ownership of a company. The trading management of these publicly listed companies is conducted by an exchange, the two largest of which are the NYSE and the NASDAQ. To be traded publicly, a company must become listed, which is a lengthy process comprising of an initial public offering (IPO). In an IPO the private company meets with an investment bank and makes certain decisions, such as the number of shares, share price based on factors such as the debt to equity ratio, revenue consistency and many more [4]. From here the investment bank takes on the task of underwriting, assuming legal responsibility for the shares and dictating a set of terms required for listing, which will include the exchange’s own terms such as minimum number of shareholders to qualify for listing. There are many benefits to becoming publicly listed, including increased prestige and diversified ownership, however, there are also many shortfalls such as forced public disclosure, pressure for short term growth and potentially making decisions with greater consideration of market sentiment, favouring market price over intrinsic value.

## **2.1.1 Forces That Move Common Stock**

The price at any given moment is a result of the supply and demand at that point in the market. However, there are 3 main factors that influence the stock price of a company: *Fundamental Factors*, *Technical Factors*, and *Market Sentiment*. *Fundamental*

*factors* refer to a combination of two things, which have a variety of metrics to determine them:

1.) The earnings base of the company.

2.) The valuation multiple (expectations about the future).

*Technical factors* are external conditions that affect the supply and demand of a company’s stock price, such as inflation [4]*.* Finally, *market sentiment* refers to the general outlook of investors toward a particular company and can be influenced by a variety of fundamental and technical factors. With the rapid availability of news as well as opinions on forums and social media constantly being shared and considered, market sentiment plays a large role in the volatility of a stock [5].

## **2.1.2 Metric 1**

## **2.1.3 Metric 2**

## **2.2 Sentiment Analysis in the Financial Domain**

Sentiment analysis is the mining of text to extract subjective information, usually to create meaningful insights. This is commonly achieved through natural language processing techniques and contextual techniques applied to a string of text, where contextual vector representations of words are used to assess the overall sentiment of the text. There are two main challenges with sentiment analysis. The first is correctly mining the intended sentiment from the text and the second is having a representative enough dataset so that the sentiment being captured is as representative of the problem as possible [6].

## **2.3 Sequence Classification with Pre-Trained Language Models**

# Chapter 3

# Implementation

This section will detail the creation of the financial sentiment analyser Fin-DeBERTa and subsequently its utilisation in a trading strategy.

## **3.1 Proposed Methodology**

**Flow chart with machine learning model**

## **3.2 Fin-DeBERTa**

This subsection details the full implementation of the Fin-DeBERTa model including the approach to additional training for downstream classification tasks in the financial domain. It also covers the Bayesian hyperparameter tuning used for training optimisation and the empirical findings within the context of current literature. Consequently, this subsection aims to answer the following **research questions**:

1. What is Fin-DeBERTa’s performance on sequence classification tasks in the financial domain when compared to other transfer learning methods such as ULMFit and ELMo?
2. How does Fin-DeBERTa compare to SOTA methods in multiclass financial sentiment sequence classification?
3. What is the performance improvement of Fin-DeBERTa after Bayesian hyperparameter optimisation?

## **3.2.1 The Financial Phrase-bank**

The Financial Phrase-bank **Source** is an annotated corpus of 4840 labelled financial phrases. The phrases are from English news articles concerning listed companies in OMX Helsinki. This dataset was chosen partly due the text sequences being semantically similar to the target news articles that will be analysed by the transformer model. Moreover, this dataset was selected for its high reliability, as sixteen members comprised of researchers and postgraduate students on financial programs were responsible for labelling the data. The dataset is split into four different groups differentiated by the % agreement on the sequence label: *negative*, *neutral* or *positive*. The dataset was divided into an 80-10-10 train-test-validation split, and all experimental analysis was conducted with 10-fold cross validation.

|  |  |
| --- | --- |
| Dataset | Number of Sentences (% Negative, % Neutral, % Positive) |
| 100% agreement | 2259 (25.2, 61.4, 13.4) |
| >75% agreement | 3448 (25.7, 62.1, 12.2) |
| >66% agreement | 4211 (27.7, 60.1, 12.2) |
| >50% agreement | 4840 (28.2, 59.3, 12.5) |

## **3.2.1 Evaluation Metrics**

The evaluation metrics used are: Accuracy, Macro-averaged F1 score, Weighted F1 score:

**Accuracy:**

**Macro-averaged F1:**

The macro-averaged f1 score involves calculating the f1 score across each class: negative, neutral and positive, individually. The scores are then summed and divided by 3. Through this, an unweighted (macro) f1 score is produced whereby each class is considered equal in size, meaning class imbalance is not accounted for. Although the Financial Phrase-bank dataset is imbalanced, this is likely the most useful f1 metric, as the class distribution of negative, neutral and positive sentiment in forums and news articles will vary depending on current events at the time. Therefore, an f1 score that considers each class equally is preferred as an evaluation metric.

**Weighted F1:**

The weighted f1 score is calculated similarly to the macro-averaged f1 score. The difference being instead of dividing the summed f1 score by the number of classes, each score is multiplied by the class’ ratio in the dataset. The class ratios for the full dataset of >50% agreement have been inserted into the equation above to illustrate this; 28.2% *negative*, 59.3% *neutral*, 12.5% *positive*.

## **3.2.2 Baseline Methods**

In order to adequately assess Fin-DeBERTa’s performance on the Financial Phrase-bank dataset, baselines are gathered from four other implementations:

**Bi-LSTM with GLoVE.**

For this implementation, a bi-directional LSTM classifier is implemented with a hidden size of 64 and hidden state size of 128. The last hidden state is mapped to a vector with 3 values by a fully connected feed-forward layer. These 3 values represent the probability of each of the 3 class labels. A dropout layer is included with a dropout probability of 0.3. The optimiser used is the Adam optimiser. The learning rate is set to 3e-5 with a warmup ratio of 0.1. The model was run for 50 epochs with early stopping exercised when there was no improvement on the validation loss for 3 consecutive iterations. The embeddings used for this implementation are the GLoVe embeddings.

**Bi-LSTM with ELMo.**

This implementation is almost identical to the one above, the only difference being ELMo embeddings are used instead of GLoVe embeddings. This is done to observe any advantage of contextualised word representations, as with ELMo the surrounding words influence the representation of the word.

**ULMFit.**

The pretrained ULMFit language model is fine-tuned for classification with the Financial Phrase-bank dataset using the *Fast.ai* library. A fully connected layer is added to the output of ULMFit. Training is conducted for 15 epochs with a learning rate of 4e-3 and an Adam optimiser. The early stopping requirement is the same as the previous two implementations.

**BERT-Base-Cased.**

The bert-base-cased model is a 120-layer, 768-hidden, 12 head, 109M parameter model pretrained on cased English text. It is implemented using the *Huggingface* library and fine-tuned over 2 epochs with a learning rate of 2e-5 and warmup ratio of 0.06 as recommended by Google Research. The Adam optimiser is also used for this implementation.

## **3.2.3 Implementing Fin-DeBERTa**

Create a table with 3 models, electra, deberta, Roberta. Include validation loss and mnli score

Run practice runs with them across 10 fold cross val Show and plot the results

Choose deberta, use the mnli version as well and compare results.

Bayesian optimisation

Report all results in a nice table

The DeBERTa-base-mnli

## **3.3 Data Collection**

The data collection process was conducted with the objective of collecting as much speculative data about general market sentiment and NASDAQ:FB as possible over the 750 day period. In an attempt to capture a fully representative dataset containing both entity based and crowd based perception **Source**, data was sourced from a combination of news providers as well as the discussion forum Reddit.

A total of 57 news providers were web scraped for articles. This included both well-established fact reporting entities [ ‘CNBC’ , ‘Reuters’, ‘Forbes’ ] as well as more speculative entities [ ‘Seeking Alpha’, ‘Market Watch’, ‘InvestorPlace’ ]. This was done with the intention that sentiment scores could be separated by source and treated as individual or aggregated hyperparameters. Additionally, both general news and company specific news was captured in order to capture the general market sentiment as well as the specific sentiment for Facebook. Python scripts were executed once per day at 0925 EST. A full list of news providers is available in the Appendices.

With regards to the Reddit data collection, five subforums or ‘subreddits’ were chosen to collect data from. The five subreddits were chosen based on a selection criteria that considers the size of the subreddit community as well as the mean number of monthly posts returned by the Regular Expression **Put search regex here.** The subreddits selected were: /r/investing, /r/wallstreetbets, /r/stocks, /r/options and /r/CryptoCurrency.

The full dataset is comprised of 103,334 Reddit comments, 7,749 Facebook specific articles and 28,962 general market articles.