A picture containing text

Description automatically generated

**CZ4079**

**FINAL YEAR PROJECT**

**FYP Specific Topics**

Shearman Chua Wei Jie (U1820058D)

FYP Project: SCSE20-0507

FYP Supervisor: Dr Althea Liang

Contents

[Introduction 3](#_Toc60101709)

[Specific Fintech Topics to Tackle 3](#_Toc60101710)

[Deep Neural Networks 3](#_Toc60101711)

[Specific Topics Identified 4](#_Toc60101712)

[Stock Trading Action 4](#_Toc60101713)

[Past Research on Stock Trading Action Prediction 4](#_Toc60101714)

[Proposed method on Stock Trading Action Prediction 5](#_Toc60101715)

[Proposed method on Classification of Sentiments for Financial News 12](#_Toc60101716)

# Introduction

Having conducted the initial experimentations on using Deep Learning models for the prediction of stock market direction, this document will go over the conclusions of the initial findings and how various data dimensions affect the difficulty of the deep learning models to learn the generalized pattern of the input data to give a prediction.

We will then look at specific problems that we would like to extend the initial findings to, in the area of fintech. This is so that we can further study the effect of the significance for each particular data dimension in terms impact of difficulty in learning for deep learning model. Also, we will be looking at how we can apply federated learning to these specific fintech problems.

# Specific Fintech Topics to Tackle

With the results obtained from the initial experiments, as well as the conclusions derived from the experiments, we will now further extend the project in order to observe the use of Deep Learning Models in Fintech applications, as well as deepening our understanding towards what contributes to the difficulty of learning these specific Fintech tasks, as well as to what tasks are more difficult for a particular Deep Learning Model to learn.

## Deep Neural Networks

A deep neural network (DNN) is an artificial neural network (ANN) with multiple layers between the input and output layers. There are different types of neural networks, but they always consist of the same components: neurons, synapses, weights, biases, and functions. The reason Neural Networks are developed in the field of Artificial Intelligence is that we will be able to have Artificial Intelligent models that emulates the cognitive functions of the human brain.

With that being said, over the years, researchers have developed a multitude of deep neural networks, to tackle specific tasks that are required to be solved by artificial intelligence.

For example, Convolutional Neural Network (CNN) is a class of deep neural networks, most commonly applied to analysing visual imagery. They are designed to emulate the visual cortex of the human brain, to process image data for image related tasks, such as image recognition and image classification. This is due to the fact that CNNs are highly capable of learning spatial information from input data.

Another example is the Long short-term memory (LSTM) network, which is a variant of an artificial recurrent neural networks (RNNs). Long short-term memory (LSTM) networks learns by using “remembered” information from previous inputs in addition to its current input data, like how the Long and short-term memory of the human brain works. This allows LSTM networks have an edge over conventional feed-forward neural networks and RNN in Sequence prediction problems like time-series prediction, and text recognition.

With that being said, some tasks will be more difficult for a particular deep neural network (DNN) to learn and some tasks will be easier. The way the input data is presented to these deep neural networks will also affect the ability of the networks to learn and generalize the input data for prediction tasks. Therefore, as an extension to our initial experiments, we would now like to explore the learning abilities of two Deep Learning Networks, Convolutional Neural Network (CNN), and Long short-term memory (LSTM) network, for certain defined Fintech tasks, as well as how the variation to the data provided to the models whilst learning, affect their ability to learn the task at hand.

## Specific Topics Identified

### Stock Trading Action

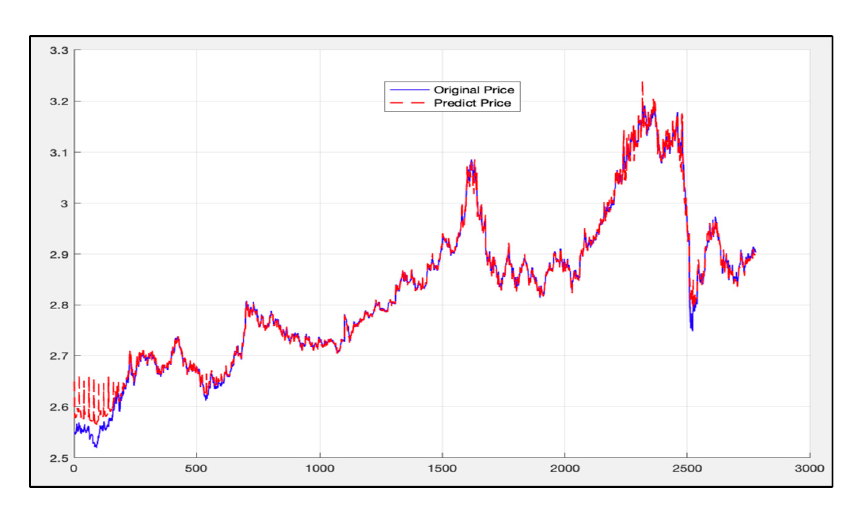
Over the years, much research regarding the use of Deep Learning model for Fintech applications have surrounded around using Deep Learning models for tasks around trading and investing of financial instruments such as Stocks or Forex. Some of these tasks include the prediction of stock prices for the next trading day, stock portfolio management, etc. Therefore, one of the specific task we have identified to solve is the prediction of Stock Trading Actions using Deep Learning models.

### Past Research on Stock Trading Action Prediction

There has been a widespread use of Deep Learning models such as LSTM (Long Short-Term Memory) RNN models, and CNN (Convolutional Neural Network) models for the purposes of stock market related applications in many past research. LSTM models have been used in many past research for the purpose of predicting the **next trading day stock price** of stocks and CNN models have been used for the purpose of predicting the trading action (to buy, hold or sell the stock) to perform on the stocks based on the input data given.

However, the way the models are developed, are not practical for real world applications as well as the fact that we have no definitive answers as to which data dimensions/factors are the most impactful towards helping Deep Learning models learn a generalized prediction pattern for stock trading applications.

For example, LSTM models have been used to predict the stock prices of the next trading day based on the past **n** number of days closing prices of the stock. While in many research paper, the results obtained by these models seems promising, the prices predicted are not really useful to stock traders to decide whether to perform a trading action on the stock. It just informs the stock trader the estimated stock price for the next trading day. In addition, the predicted stock prices are usually “lagging” the actual stock prices which can be seen if the predicted and actual prices are plot on the same graph. What this means that when there is a change in the stock direction ( rising-to-falling or falling-to-rising), the predicted prices can only reflect the change in stock direction a few trading days later. Therefore, stock traders are not able to make use of the predicted stock prices to predict changes in stock directions which can be helpful for traders to make traders. This renders LSTM models to predict the stock prices of the next trading day not very useful for stock trading.



**Figure from Yujie Fang, Juan Chen, and Zhengxuan Xue, Research on Quantitative Investment Strategies Based on Deep Learning showing “lagging” predicted prices**

For previous research using CNN models, researchers uses CNN to determine the "Buy", “Hold”, and "Sell" points in stock prices using 15 different technical indicators with different time intervals and parameter selections for each daily stock price time series to create images.

The technical indicators for each day, RSI, Williams %R, WMA, EMA, SMA, HMA, Triple EMA, CCI, CMO, MACD, PPO, ROC, CMFI, DMI, and PSI values for different intervals (6 to 20 days) are calculated and for each day a 15x15 image is generated by using 15 technical indicators and 15 different intervals of technical indicators.

From one of these research, we can observe that the precision and recall of the model is not very high for the “buy” and “sell” trade action labels. This means that the model will not be very practical for use in a real-life trading scenario as how stock trades profits is by making “buy” and “sell” . However, with that being said, the model can still be profitable if most of the predicted “buy” points are cost lesser than most of the predicted “sell” points.

A plausible reason for the CNN model not performing well when predicting the “buy” and “sell” trade action labels is the fact that the input data contains too many data dimensions and we do not know which data dimensions are actually useful in helping the CNN model learn the prediction of stock trade action labels. In addition, the way the data “image” is created, compresses each technical indicator past 15 days data points as a 1-D vector, and we know that Convolutional networks were inspired by biological processes in that the connectivity pattern between neurons resembles the organization of the animal visual cortex where it may be hard for the CNN model to “visualize” the 1-D vectors of the 15 technical indicators, making it hard to learn the “patterns” in the technical indicators to learn them efficiently to make predictions on stock trade action labels.

### Proposed method on Stock Trading Action Prediction

In order to solve the specific problem of Stock Trading Action Prediction, and to deepen our understanding of the difficulty of learning Stock Trading Action Prediction task by Deep Learning models, we will be utilizing the Convolutional Neural Network (CNN), and Long short-term memory (LSTM) network. From our experiments, we would like to observe the ability of the 2 Deep Neural Networks to solve the specific problem of Stock Trading Action Prediction, or in other words, the difficulty of learning the task of Stock Trading Action Prediction by the networks.

In addition, we would introduce variance to the input data that is fed to the Deep Neural Networks for training. By introducing these variations, we hope that it will allow us to further our understanding of how variations in the data presented to these models affect the difficulty of learning the prediction task.

#### Providing Variance to Input Data for Understanding the Difficulty of Learning in Deep Learning Models

As mentioned above, we will be utilizing the Convolutional Neural Network (CNN), and Long short-term memory (LSTM) network for our experiments. It was also mentioned that the 2 Deep Neural Networks have distinctively different architectures and advantages. Therefore, we would like to observe the effects varying the data dimensions fed to the networks on the difficulty of learning the task of Stock Trading Action Prediction.

In past research that utilizes LSTM networks for stock market related prediction tasks, the input data presented to the networks are tensors of time-series data[1], usually of 3 dimensions, where the first dimension is sample size, second dimension is number of time steps and the last dimension corresponds to the number of features for each time step, usually technical indicators of stocks. The LSTM networks then process each sample sequentially in time steps for the prediction task.

In past research that utilizes CNN networks for stock market related prediction tasks, there are 2 main methods of inputting data to the CNN networks. The first method is to present the technical indicators of the stock to make predictions on, as an image array of a graph of the past **n** trading days[2], similar to how a human trader will view graphs of technical indicators for past trading days and make a decision on a stock trading action. The second method is to present the technical indicators of the stock to make predictions on, as a **n**x**n** array generated by using **n** technical indicators and **n** different intervals of technical indicators[3], which is akin to a human trader viewing the technical indicators for past trading days as a spreadsheet to make a decision on a stock trading action.

**Variation 1 - Varying the Method Input Data is Presented to Deep Neural Networks**

Therefore, for our experiments, the first variation that we will introduce to the Deep Neural Networks, is how the input data is presented to them. In another words, we will vary the dimensionality and presentation of the input data to the two Deep Learning models.

As mentioned beforehand, LSTM models are proficient at solving time-series related prediction tasks and CNN models are apt at processing spatial information and are proficient at solving image related prediction tasks. However, with that being said, LSTM models have been used for at solving image related prediction tasks[4, 5], and CNN models have been utilized in solving time-series related prediction tasks[3].

For the task of Stock Trading Action Prediction, we will be using 4 technical indicators (WILLR, Closing Price, EMA, OBV), from the past 15 days as the input data to predict the Stock Trading Action to be performed for the current trading day. Having established the input data to be used, we then establish 2 methods of presenting the information of the input data to the 2 Deep Learning Models.

The first method is to present the 4 technical indicators, from the past 15 days input data as an image of plots of graphs, of the past 15 trading days, where each graph corresponds to a particular technical indicator, to the Deep Learning Models. The images will be converted into numpy arrays and fed to the models. The second method is to present the 4 technical indicators, from the past 15 days input data as a time-series vector where the 3 dimensions of the vector will be sample size, number of time-steps and the number of features.

|  |  |
| --- | --- |
| Method 1 | Method 2 |
| **Image of technical indicators graphs**    **Numpy Array Of Image** | **Time-series vector of technical indicators** |

The reason for introducing varying the way information from the input data is presented to the 2 Deep Learning model, we hope to understand if one way information from the input data is presented to the Deep Learning models proves more difficult for the models to learn the input data, as compared to the other. The reason behind this variation experiment is that LSTM models and CNN models emulates different cognitive functions of the human brain and the way information from the input data is presented to the 2 Deep Learning models may affect the difficulty of the models in learning of the stock trading action prediction task. For example, as CNN model emulates the visual cortex of the human brain, input data presented in the form of Method 1 may be akin to a human trader looking at graphs of technical indicators for past trading days, and input data presented in the form of Method 2 may be akin to a human trader looking at a spreadsheet of the technical indicators for past trading days, and that these 2 methods of presenting the input data to the CNN model may stimulate the model differently just like how different images may provide different visual stimulation to the visual cortex.

**Variation 2 - Varying the Duration of the Input Data Presented to Deep Neural Networks**

The second variation to the data that we will be introducing is the variation of the duration of the past trading days of the 4 technical indicators to be used as input data. In the first variation experiment, the input data consisted of data of 4 technical indicators from the past 15 trading days. This will act as the control for our second variation. We will then vary the number of past trading days of the 4 technical indicators to be used as input data. The number of days used will be, 5, 10, 15, 20, 25.

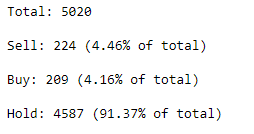
The way the information will be presented to the 2 Deep Learning models, will be the best of the 2 methods the information is presented to the 2 Deep Learning models in **Variation 1** experiments for each of the 2 models, so that the way information presented to the 2 Deep Learning models will be the best way for each of them to ensure that the difficulty in learning of the stock trading action prediction task in **Variation 2** experiments is only affected by the variation of the duration of the past trading days of the 4 technical indicators to be used as input data.

The reason for introducing this variation to the data is to determine whether it is harder for the 2 Deep Learning models to learn and generalize the input data with more trading days’ worth of data or with less trading days’ worth of data. This will hopefully deepen our understanding of how the number of time-steps for time-series data affects the ability of Deep Learning models to identify a pattern in the input data sequence and make an accurate prediction from it.

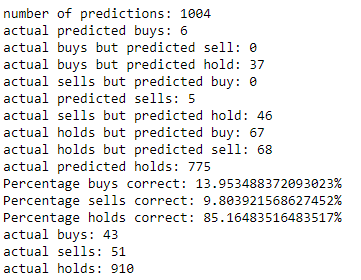
**Variation 3 - Varying the Ratio of Training Data** **for Each Class Presented to Deep Neural Networks**

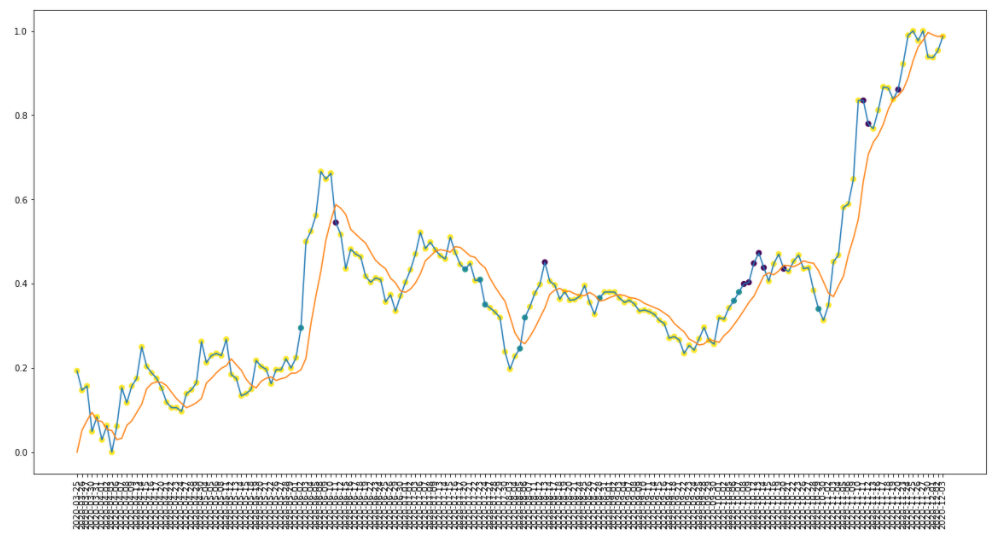
The third variation to the data that we will be introducing is the variation of the Ratio of Training Data for Each Class of the stock trading action prediction task. For the task of the stock trading action prediction, we will be a style of trading that attempts to capture short- to medium-term gains in a stock over a period of a few days to several weeks, known as swing trading. Due to the fact that in swing trading, stock positions are often held for a few days to several weeks, there will be more “hold” Class labels in the training data as compared to “buy” and “sell” labels.

In the initial experiments, we just utilized 20 years of daily trading data as the training input data for our Deep Learning models. The training data’s Classes Ratio is as follows:



From the figure, we can observe that the data consisted of mainly data from the “hold” class. For the initial experiments that we have conducted earlier, with this given training data’s Classes Ratio, we were able to achieve a decent validation accuracies between 76% to 84% after training the models, however, of the correctly predicted labels, most of them came from the “hold” class, meaning that the trained models are capable of predicting “hold” class labels, but not “buy” and “sell” class labels, even when we used class weights when training the models.



With that being said, although the models are not able to predict the “buy” and “sell” labels with great accuracy, the predicted “buy” and “sell” points are generally correct, meaning that the “buy” points are when the stock prices are low and the “sell” points occur when the stock prices are high.  


As can be seen from the figure above, the yellow points are “hold” labels, purple points are “sell” labels, and blue points are “buy” labels and that the blue points occur when the stock prices are low and “sell” points occur when the prices are generally high.

Therefore, for the third variation to the data, we would like to observe if changing the Ratio of the amount of training data for each class label affects the difficulty in learning of the stock trading action prediction task for our Deep Learning models.

We will explore how an equal ratio of the amount of training data for each class label, a ratio of training data that contains more training data for “buy” class labels, and a ratio of training data that contains more training data for “sell” class labels affect the difficulty in learning of the stock trading action prediction task for our Deep Learning models.

**Variation 4 - Varying the Periods Input Features are Calculated Over**

For the experiments to be conducted for **Variation 1** through **Variation 3**, the Technical Indicators that will be used are: Closing Price, Exponential Moving Average (EMA), Williams %R(WILLR), and On-Balance Volume (OBV). Amongst these indicators, the Exponential Moving Average (EMA), Williams %R(WILLR) indicators are derived from calculating stock prices over a lookback period, which in the experiments to be conducted for **Variation 1** through **Variation 3** is a standardized lookback period of past 7 trading days.

The fourth variation to the data that we will be introducing is the variation of the lookback period of the Training Data to be used for the stock trading action prediction task. The lookback period of past 7 trading days will act as a control since its the lookback period standardized for the experiments to be conducted for **Variation 1** through **Variation 3**. We then vary the lookback period by decreasing and increasing the lookback period from the lookback period of past 7 trading days.

The reason for introducing the variation of the lookback period of the Training Data to be used for the stock trading action prediction task, is to determine if variation in the lookback period of the Training Data affect the difficulty in learning of the stock trading action prediction task for our Deep Learning models.

**Variation 5 - Varying the Input Features Used as Input Training Data**

The fifth variation to the data we will be introducing, is the variation to the Input Features used as Input Training Data to the Deep Learning models. For the experiments to be conducted for **Variation 1** through **Variation 4**, the Technical Indicators that are used are: Closing Price, Exponential Moving Average (EMA), Williams %R(WILLR), and On-Balance Volume (OBV). Each of these indicators tracks a different aspect and behaviour of a stock.

The Closing Price is quite straightforward where it tracks the Closing Prices of the stock for the past trading days. EMA is a moving average indicator, where moving averages are usually calculated to identify the trend direction of a stock or to determine its support and resistance levels, it is a trend-following—or lagging—indicator because it is based on past prices. Williams %R, also known as the Williams Percent Range, is a type of momentum indicator that moves between 0 and -100 and measures overbought and oversold levels. The Williams %R may be used to find entry and exit points in the market. Finally, On-balance volume (OBV) is a technical trading momentum indicator that uses volume flow to predict changes in stock price.

Due to the fact that each of these indicators tracks a different aspect and behaviour of a stock, we would like to find out which of these indicators is better for the Deep Learning models to learn the data and make stock trading action predictions, and by removing which indicator, it will cause the learning of the stock trading action prediction task for our Deep Learning models to be more difficult.

We will vary the input data by a leave-one-out method, where we train the Deep Learning models using only 3 out of the 4 original technical indicators used in experiments to be conducted for **Variation 1** through **Variation 4**, as the input data. The input data with all 4 original technical indicators will act as the control for this data variation experiment.

The reason behind introducing this variation to the data is to allow us to further understand for the task of stock trading action prediction, how does varying the type of Input Features used in the Input Training Data affects the difficulty in learning of the stock trading action prediction task for our Deep Learning models.

#### Method to Measure How Much Difficulty of Learning is Affected for the Deep Learning Models

In order to assess the magnitude in the Difficulty of Learning for the Deep Learning Models posed by the various variation experiments, we will need a method to quantify the Difficulty of Learning for the Deep Learning Models.

From the initial experiments, we can see that we cannot rely on accuracy or loss as part of the measurement in the Difficulty of Learning for the Deep Learning Models for the task of Stock Trading Action Prediction task as most of the correctly predicted labels for the validation dataset is of the “hold” label, with poor accuracies obtained for the “buy” and “sell” labels. In addition, it will not be fair to judge the models by how accurately the are able to pinpoint “buy” and “sell” labels as even the best human Stock traders will not be able to do so. A better gauge will be how well the Deep Learning Models are able to make profits from the Stock Trading Actions predicted, i.e., “buy” points are lower than “sell” points, and the difference in price is greater than the trading fees that would have been incurred.

Therefore, in order to determine how well the Deep Learning models have learnt the training data, or in another words how well the models have generalized to the data, we should instead use a measurement whereby the models are measured by how well the Deep Learning Models are able to make profits from the Stock Trading Actions predicted. The reason behind using this method of measurement is that the better a Deep Learning model is at generalizing the Stock Trading Actions Prediction task, the better the model is able to predict high price points to “sell” at and low price points to “buy” at, which in turn allow the model to be able to make profits from the Stock Trading Actions predicted.

In order to measure how well the Deep Learning Models are able to make profits from the Stock Trading Actions predicted, we use an algorithm to trade with the buy points and the sell points predicted by the models on the test set. If the price of the “sell” point is higher than that of the “buy” point by more than the trading cost incurred ( 0.55% for 2 trades actions, buy and sell performed), the model will have made a profit. The algorithm will start with 10,000 dollars to trade and we will determine the profit gained in percentage after performing the trading actions over the trading period of the test set. During a financial trade, if the same label comes consecutively, only the first label is activated and the corresponding transaction is performed. Repeating labels are ignored until the label changes.

**Algorithm:**

**Initial\_capital** = 10 000

**Stocks\_capital** = 0

**Previously\_bought** = False

For trading day in test period:

{ If label is “buy”:

If **Initial\_capital** Not Equal **0** and **Previously\_bought** Equal **False**:

**Stocks\_capital** = **Initial\_capital** \* 0.99725 (due to trade commission of 0.275%);

**Initial\_capital** = 0;

**Previously\_bought** = True;

If label is “sell”:

If **Stocks\_capital** Not Equal **0** and **Previously\_bought** Equal **True**:

**Initial\_capital** = **Stocks\_capital** \* 0.99725 (due to trade commission of 0.275%);

**Stocks\_capital** = 0;

**Previously\_bought** = False;

}

If **Stocks\_capital** Not Equal **0:**

**Initial\_capital** = **Stocks\_capital** \* 0.99725 (due to trade commission of 0.275%);

Profit = (**Initial\_capital /** 10000) \* 100

By finding the profit margin based made by the models on the predicted Stock Trading Actions, we will be able to determine how well the models have actually learn and generalize the data and Stock Trading Action Prediction task by identifying the high price points to “sell” and low price points to “buy”. Therefore, using the profit margin, we will be able to rank the models based on the profit margin and from there determine if the Difficulty of Learning for the Deep Learning Models is affected by the variations, if the variations resulted in more difficult learning for the Deep Learning Models, the profit margin of the test set predictions will be lower and vice versa if , if the variations resulted in less difficult learning for the Deep Learning Models.

Sentiment Analysis for Financial News

Fluctuations in stock prices in stock markets on a day-to-day basis are often not driven by the underlying value of the stock, meaning that an overvalued stock can still increase in stock price and an undervalued stock can still decrease in stock price. This to due to the fact that on a day-to-day basis, stock prices are often affected by the sentiments of traders and investors from news about the stock. A bad piece of news may cause a fall in the stock price and a good piece of news may cause a rise in the stock price.

Therefore, Sentiment Analysis for Financial News is also an important aspect of study in the Fintech field. Ultimately, the goal of Sentiment Analysis for Financial News is to create Artificial Intelligence models that can predict stock prices based on the Sentiments of Financial News. However, in order to achieve that goal, it is important that we create Artificial Intelligence models that can correctly classify the Sentiments of an underlying piece of Financial News, whether the sentiments towards the news is positive or negative.

Therefore, one of the specific task we have identified to solve is the Classification of Sentiments for Financial News using Deep Learning models. For this task, we will be utilizing the Financial PhraseBank from Malo et al. (2014) where the FinancialPhraseBank Dataset contains the sentiments for financial news headlines from the perspective of a retail investor. The dataset contains two columns, "Sentiment" and "News Headline". The sentiment can be negative, neutral, or positive.

### Proposed method on Classification of Sentiments for Financial News

In order to solve the specific problem of Classification of Sentiments for Financial News, and to deepen our understanding of the difficulty of learning Classification of Sentiments for Financial News task by Deep Learning models, we will be utilizing the Convolutional Neural Network (CNN), and Long short-term memory (LSTM) network. From our experiments, we would like to observe the ability of the 2 Deep Neural Networks to solve the specific problem of Classification of Sentiments for Financial News, or in other words, the difficulty of learning the task of Classification of Sentiments for Financial News by the networks.

In addition, we would introduce variance to the input data that is fed to the Deep Neural Networks for training. By introducing these variations, we hope that it will allow us to further our understanding of how variations in the data presented to these models affect the difficulty of learning the prediction task.

#### Providing Variance to Input Data for Understanding the Difficulty of Learning in Deep Learning Models

As mentioned above, we will be utilizing the Convolutional Neural Network (CNN), and Long short-term memory (LSTM) network for our experiments. It was also mentioned that the 2 Deep Neural Networks have distinctively different architectures and advantages. Therefore, we would like to observe the effects varying the data dimensions fed to the networks on the difficulty of learning the task of Classification of Sentiments for Financial News.

**Variation 1 - Varying the Level Which News Headline is Processed by the Deep Learning Models**

Text classification is a task to assign text documents according to its content to one or more classes automatically. For the task of text classification, Deep Learning models have been used as both character-level models and word-level models where text is process in terms of character and word respectively for classification.

For the first variation we will be introducing is to vary the level in which the Classification of Sentiments for Financial News is performed, either on word level or on character level. By introducing this variation, we hope to be able to understand better if Convolutional Neural Networks (CNNs), and Long short-term memory (LSTM) networks have more difficulty in learning of the task of the Classification of Sentiments for Financial News from input text data when text is process in terms of character or word.

**Variation 2 – Adding of Text Noise to the News Headlines Processed by the Deep Learning Models**

Financial news can come in all forms and from many sources. With the growing popularity of digital media, investors and traders not only get financial news from the newspapers, but also digital news platform like Bloomberg and even from social media platforms as well, like Twitter.

With that being said, because financial news can come in all forms and from many sources, the way a same piece of news is written or presented can be different. Therefore, in addition to the main points of the news, other redundant information may also be included inside the new article. When the new articles are scrapped from the sources and fed into the Financial News Classification model, it may contain “noise” words, or words that may not be relevant to the main points of the news. For example, the news “Finnish Talentum reports its operating profit increased to EUR 20.5 mn in 2005 from EUR 9.3 mn in 2004 , and net sales totalled EUR 103.3 mn , up from EUR 96.4 mn .” may be presented in one news sources as “Financial news update – Latest news: Finnish Talentum reports its operating profit increased to EUR 20.5 mn in 2005 from EUR 9.3 mn in 2004 , and net sales totalled EUR 103.3 mn , up from EUR 96.4 mn .” where “Financial news update – Latest news:” are redundant noise words that do not contribute to the point of the main news.

Therefore, for the second variation to the data for the Deep Learning Models, we will add neutral phrases or sentences to the FinancialPhraseBank Dataset news headlines as “noise” and observe if this variation to the data affects the difficulty of the Deep Learning models to learn the task of Classification of Sentiments for Financial News.

**Variation 3 – Varying the Ratio of Training Data for Each Class Presented to Deep Neural Networks**

The third variation to the data that we will be introducing is the variation of the Ratio of Training Data for Each Class of the Classification of Sentiments for Financial News task. Similar to the Stock Trade Action Prediction task, the ratio of data for each class label for the FinancialPhraseBank Dataset is not balanced.

In the dataset, 12.5% of the data have a “negative” label, 28.1% of the data have a “positive” label, and 59.4% of the data have a “neutral” label. Therefore, we can see that the dataset have imbalanced classes with more training data for some classes and less training data for other classes, which may affect the difficulty of the Deep Learning models to learn the task of Classification of Sentiments for Financial News.

Therefore, for the third variation to the data, we would like to observe if changing the Ratio of the amount of training data for each class label affects the difficulty in learning of the Classification of Sentiments for Financial News task for our Deep Learning models.

We will explore how an equal ratio of the amount of training data for each class label, a ratio of training data that contains more training data for “negative” class labels, and a ratio of training data that contains more training data for “positive” class labels affect the difficulty in learning of the Classification of Sentiments for Financial News task for our Deep Learning models.

#### Method to Measure How Much Difficulty of Learning is Affected for the Deep Learning Models

In order to assess the magnitude in the Difficulty of Learning for the Deep Learning Models posed by the various variation experiments, we will need a method to quantify the Difficulty of Learning for the Deep Learning Models.

For the task of Classification of Sentiments for Financial News task for Deep Learning models, we will assess the magnitude in the Difficulty of Learning for the Deep Learning Models posed by the various variation experiments by the average improvement of validation accuracy upon every epoch divided by the most recent accuracy value. The formula is derived below:

**Learning improvement = Average(Accuracy Improvement of Every Epoch) / Most Recent Epoch’s Accuracy**

By finding the Learning improvement as defined by the formula above, we will be able to quantify how well the models have actually learn and generalize the data and Classification of Sentiments for Financial News task. Hence, we will be able to rank the models based on the Learning improvement and from there determine if the Difficulty of Learning for the Deep Learning Models is affected by the variations, if the variations resulted in more difficult learning for the Deep Learning Models, the Learning improvement will be lower and vice versa if the variations resulted in less difficult learning for the Deep Learning Models.

[1] Y. Fang, J. Chen, and Z. Xue, "Research on quantitative investment strategies based on deep learning," *Algorithms,* vol. 12, no. 2, p. 35, 2019.

[2] G. Hu *et al.*, *Deep Stock Representation Learning: From Candlestick Charts to Investment Decisions*. 2018, pp. 2706-2710.

[3] O. Sezer and M. Ozbayoglu, "Algorithmic Financial Trading with Deep Convolutional Neural Networks: Time Series to Image Conversion Approach," *Applied Soft Computing,* vol. 70, 04/27 2018, doi: 10.1016/j.asoc.2018.04.024.

[4] A.-A. Nahid, M. A. Mehrabi, and Y. Kong, "Histopathological Breast Cancer Image Classification by Deep Neural Network Techniques Guided by Local Clustering," *BioMed Research International,* vol. 2018, p. 2362108, 2018/03/07 2018, doi: 10.1155/2018/2362108.

[5] J. Geng, H. Wang, J. Fan, and X. Ma, "SAR image classification via deep recurrent encoding neural networks," *IEEE Transactions on Geoscience and Remote Sensing,* vol. 56, no. 4, pp. 2255-2269, 2017.