The Incorporation of Financial Methods for Stock Market Predictions by Machine Learning Algorithms.

Stuart Manock

Submitted in partial fulfilment of the requirements of Edinburgh Napier University for the Degree of Master of Science in Computing

School of Computing

August 2020

Supervisor: Dr Andreas Steyven

Second Examiner: Dr John Owens

MSc dissertation check list

Student Name: Stuart John Manock	Matric: 40435519

Please insert this form, loose-leaf, into each copy of your dissertation submitted for marking.

Milestones	**Date of completion**	**Target deadline**
Proposal	29/06/2020	Week 3
Initial report	29/06/2020	Week 7
Full draft of the dissertation	22/08/2020	2 weeks before final deadline

Learning outcome	**The markers will assess**	**Pages**	**Hours spent**
Learning outcome 1 Conduct a literature search using an appropriate range of information sources and produce a critical review of the findings.	* Range of materials; list of references * The literature review/exposition/background information chapter	19 – 50	200
Learning outcome 2 Demonstrate professional competence by sound project management and (a) by applying appropriate theoretical and practical computing concepts and techniques to a non-trivial problem, or (b) by undertaking an approved	* Evidence of project management (Gantt chart, diary, etc.) * Depending on the topic: chapters on design, implementation, methods, experiments, results, etc.	Appendix 51 - 71	

project of equivalent standard.			
Learning outcome 3 Show a capacity for self-appraisal by analysing the strengths and weakness of the project outcomes with reference to the initial objectives, and to the work of others.	* Chapter on evaluation (assessing your outcomes against the project aims and objectives) * Discussion of your project's output compared to the work of others.	72-78	(for exampl e,
Learning outcome 4 Provide evidence of the meeting learning outcomes 1-3 in the form of a dissertation which complies with the requirements of the School of Computing both in style and content.	* Is the dissertation well-written (academic writing style, grammatical), spell-checked, free of typos, neatly formatted. * Does the dissertation contain all relevant chapters, appendices, title and contents pages, etc. * Style and content of the dissertation.		(
Learning outcome 5 Defend the work orally at a viva voce examination.	* Performance * Confirm authorship		1 hour

Have you previously uploaded your dissertation to Turnitin? No

Has your supervisor seen a full draft of the dissertation before submission? No

Has your supervisor said that you are ready to submit the dissertation? Yes

Authorship Declaration

I, Stuart Manock, confirm that this dissertation and the work presented in it are my own achievements.

Where published work has been used, its authors have been clearly stated. This is true for both practical and theoretical input.

All other work is my own, help that was provided to complete this dissertation has been acknowledged.

Where my research has furthered a field, I have tried to the best of my ability to acknowledge the original authors of this work.

I have read and understand the penalties associated with Academic Misconduct.

Signed: S Manock (Stuart John Manock)

Date: 22/08/20

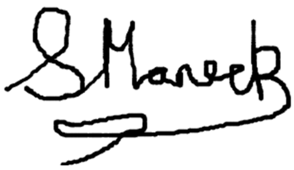
Matriculation Number: 40435519

Data Protection Declaration

Under the 1998 Data Protection Act, The University cannot disclose your grade to an unauthorised person. However, other students benefit from studying dissertations that have their grades attached.

I hereby allow:

The University may make this dissertation, with indicative grade, available to others.

Signed:  (Stuart John Manock)

AbstractContents Table

Table of Figures

Preface and Acknowledgements

I would like to preface this dissertation with special thanks to those who helped me to complete this work. I would like to thank my parents and family. Particularly, my brother Blaine for his advanced technical knowledge and my brother Martin for helping me to focus my attention on particular business components of the dissertation.

I would like to thank my Professor, Dr D Gkatzia whom invigorated me to incorporate their taught topics into my Dissertation.

Finally, I would like to thank Dr Andreas Steyven for taking a leap and supervising my project. His willingness to invest time and effort in a field separate from his immediate skill set was inspiring. This helped with the inspiration and pushed me to optimise and improve my skill set with novel technologies. A big thanks to John Owens who volunteered to undertake the role of Internal Examiner. Both Napier professors accepted my late request and for that I am incredibly grateful.

Chapter 1

1.1 Introduction to Thesis

This project looks to determine how various types of financial information change the profitability of stock market predictions. The main research question focuses on which models and parameters for machine learning (ML) algorithms perform the best as well as how optimising them can improve their performance further. New acquisition methods and styles of data processing are discussed and evaluated. Quantification of performance within the thesis will be achieved using ML industry standard metrics and self-derived performance metrics. External reports, primary and secondary research will make the basis of the discussion. Performance criteria will focus around absolute and relative comparisons.

This thesis aims to detail economic theory and statistical methods used to calculate share price. Within I hope to highlight some of the shortcomings of these methods and explain why they fall short of encompassing the true nature of a market's actions. Beyond this I hope to explain how data science methods – data analytics, deep learning and artificial intelligence powered systems – perform relative to conventional methods of stock market involvement.

I will achieve this by building an array of example computer programs that harness data procurement, processing and utilisation. These methods will be compared and contrasted quantitatively. Each step of this process is variable and extensive in its capacity for research. Acquisition, processing, incorporation and insights of data will all be evaluated with scientific rigour.
I will also try to highlight the issues involved and further explain how neural network style algorithms can outcompete traditional theory by encompassing the complete understanding of market theory through supervised learning. I will test the impact of external data out with individual companies to find which data is most influential in building better prediction software.

Discussions of market information outside the realm of numeracy will be brought into focus. By discussing other data known to influence markets I will be able to comment on how 'softer' data impacts the markets provide paths for further study within this area. Ideally work in this area would have been added. Large volumes of relevant sentiment analysis data are difficult to procure. Literature will still provide a baseline impact for our primary research to be compared to.

This topic is broad in range and complex in its nature. Because of this I do not expect to create a complete description of the market forces using my own experiments. For areas that require other complex ML algorithms I will provide insight from academic literature and give an analysis on its relevance to this thesis.

I will be employing particular software and hardware to complete the practical component of my work. Understanding functionality is key to assessing the correctness of work completed and under pins the quality of any deductions created during the deployment of the software. Further work exploring potential pitfalls and alternative options will be critically assessed for validity and feasibility of action.

There are a wide range of landmark issues that economists believe prevent the prediction of the stock market, namely the Efficient Market Hypothesis. By understanding its theory and others we can further explain how it ranks in credibility.

1.2 Main Research Questions

- Which relevant ML architectures perform the ‘best’ at stock market prediction tasks?
- What are the more impactful financial indicators for prediction success?
- How do we measure this success?
- What steps can be automated to improve the software’s performance?

1.3 Aims of the Study

- From the literature surrounding trading, critically assess a subset of prominent statistical techniques used by career traders to inform trading decisions they make. Investigate how and why new market technologies and methods are outcompeting old techniques to achieve better results.
- Design a set of experiments to create primary sources to quantitatively rank the models discussed. Evaluate how these experiments impact which technologies are most viable and resistant to critique.
- Provide an in-depth account of the methods used to produce these results so the conditions can be reproduced by impartial secondary experimenters.
- Discuss and evaluate these results with regard to the literature and research questions.
- Provide insight into the weaker areas of experimentation and the future steps that would be taken to further improve the robustness and completeness of this study.

1.4 Methodology Overview

To explain how the goals and aims of this thesis will be achieved a diagrammatic approach is shown below.

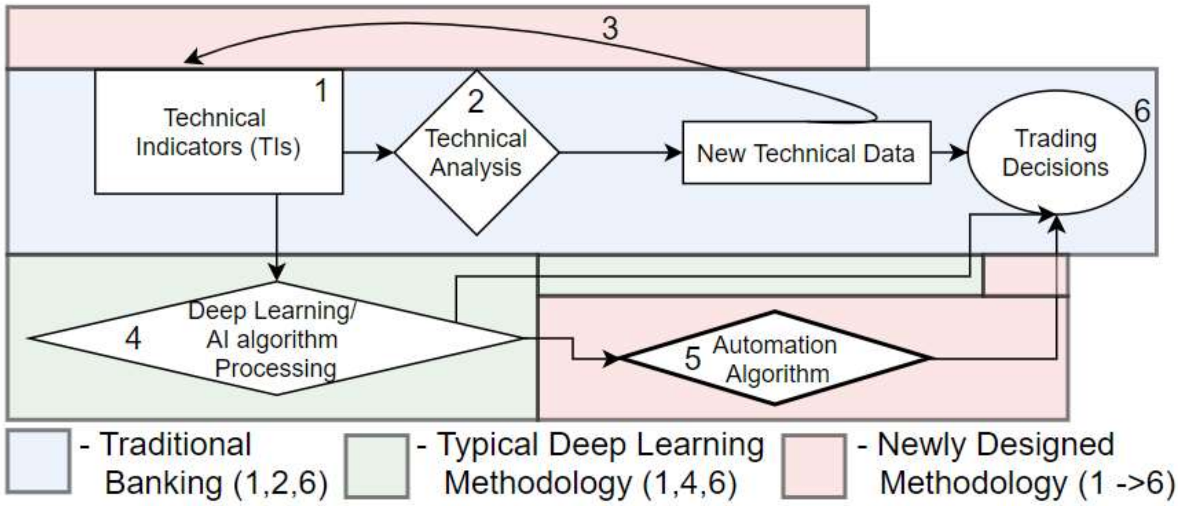


Figure 1 – Different methodologies for stock market prediction (Colour Coded). A diagrammatic approach to the thesis' aims.

Technical Indicators (1) sourced or created (3) for use in different analysis methods. These methods are either conventional (2) as an investor would do using Traditional Banking practices. Alternatively, Deep Learning (4) can be used to predict prices. The subsequent process of Automation (5). The final trading decisions created from all the available data (6). The main design elements focus on the Deep Learning Styles (4) and the Automation (5).

Shown above in Figure 1 is the traditional approach technical analysis is used to inform trading decisions (in blue). By deriving indicators and recursively adding these to our pool of data we will improve our dataset. This dataset will then be cleaned using data processing method. These are discussed later on. This cleaned data is then passed to our deep learning model. This uses complex statistical processing to train and allow sequential predictions to made about companies. This data can then on its own to determine the appropriate trading decisions. We will further this processing by feeding our predictions into component 5 - an automation algorithm - that converts these predictions into real decisions and is able to rate the performance of these decisions against the market.

The experimental methodology follows 3 distinct sections.

First different technical indicators are assessed for their impact on performance.

Secondly, we assess which hyperparameters improve the models performance.

After this we summate these processes and then take our best performing models and coalition of technical indicators to see how well they perform with regard to real world performance.

Our available decisions are:

- Buy – we convert our available money into shares.
- Hold – we conserve the shares and money we currently have
- Sell/Short – We sell any shares we own. Further from this we short the position. By this we are selling shares we do not have with the promise to buy an equal amount at a later time to offset the debt we have accrued.

We compare our performance to a generic strategy where a stock is bought and held or if the share was never purchased and the money kept instead.

1.5 Introduction to documentation structure

This thesis is divided into clear, compartmentalised areas. This allows for secular parts of the report to stand up on their own. However, in addition, particular sections contain cross examination of sections of the report. Within these areas, clear directions to the other materials discussed is highlighted at the beginning of the chapter.

Chapter 1 – Administrative Report Introduction.

- This chapter gives a generalised overview of the total thesis.
- Directionality is provided.
- Its overall aims and goals are explained.
- Explanation for the thesis' structure is given.

Chapter 2 – Literature Review.

- This chapter goes into detail regarding the background and history of stock market trading with regard to computational assistance and the indicators used.
- It outlines how the methodologies behind stock market decisions has drastically changed since its inception, particularly with reference to the continued growth of the IoT.
- It outlines how new styles of algorithm development have grown and developed into the more user-friendly experience employed by traders today.
- It delves into other areas of research for these algorithms and how these techniques have been adopted to understand complex trading behaviour.

- Further research regarding the multifaceted impacts that different spheres of influence have on the outcome of complex machine-like trading algorithms.

Chapter 3 – Methodology and the surrounding in depth theory.

- This chapter explains in detail the high-level structure of the experimental process. Each part of the design process is then broken down and explained at length.

Chapter 4 – Methodology

- The outcomes and creation of the experiments are described within the project's context.

Chapter 5 – Results & Evaluation

- These results are further expanded on with regard to the wider context. Also included are pitfalls and potential improvements on the experimental procedure.

Chapter 6 – Discussion

- Future proposed improvements to work are discussed. Critical evaluation of the work completed is further assessed with regard to the wider field and the work of other.

Chapter 7 - Conclusion

- This chapter summarises the overall state and management of the project. Important and unusual findings are further discussed and concisely summarised.

- Hindsight and potential future work is discussed.

Chapter 2- Literature Review

2.1 Introduction to Literature Review

Here within our literature review details of the current knowledge base and its position relative to the thesis will be discussed. Particular topics of importance are more critically discussed. Less prominent portions of the literature will have additional information referenced for additional reading.

Section 2.1 provide an introduction and overview

Section 2.2 explains how computing and data analytics are changing the approach to stock trading.

Section 2.3 gives an overview of economic theory relevant to the topic.

Section 2.4 explains Technical Indicators and how Technical Analysis can derive more complex Technical Indicators.

Section 2.5 gives an explanation of the role fundamental analysis plays in share price.

Section 2.6 explains other types of analysis surrounding machine learning predictions.

Section 2.7 Discusses deep learning analysis

2.2 The Intersection of Stock Trading and Machine Learning

Computing is intertwined with many aspects of life for people in the developed world. It serves as a method of automation for mundane aspects of living. Thanks to advances in technology, productivity has seen exponential growth in many commercial and academic areas (Ellingrud, 2018). Stock Markets are not immune to these realities.

Data Analytics is one field of technical application. It has provided insight into deeper understanding of complex relationships, paving the way for greater performance. Financial institutions involved in trading have seen the utility of this technology ('The Rise of the Financial Machines', 2019). This first started with Technical Indicators, its analysis and trading decisions. In the more recent sphere of data and deep learning, complex functional algorithms continue to improve computer utility (Figure 1 component 4).

Deep learning has proved immensely effective at a wide range of tasks. These include natural language processing (Nadkarni et al., 2011), computer vision, facial recognition (Phillips et al., 2018), pattern recognition, clustering and other real value tasks. Its utility even extends to Bioinformatics and Fraud detection. This wide versatility is making computer science and data analytics the essential toolkit for businesses and companies moving into the future.

Machines continue to improve their abilities. This is mainly attributed to greater data creation and faster processing in recent years. The holy grail of data science tasks is predicting the stock market. It is complex in its activity and showing that deep learning models can perform well in this task would be a great achievement for the computer community. Any and all information contributes to share price changes (Da et al., 2011). This fact provides an almost infinite data set. Online information plays an increasingly important role in shaping public impressions of companies in the digital age (Li et al., 2018). Techniques to utilise this information in its many forms adds another field of study to inform trading decisions (Strycharz et al., 2018).

With this increase in data availability and production deep learning algorithms continue to improve. An increase in worldwide focus on deep learning has furthered the field with regard to its repertoire of tools to solve varying problems (Hochreiter, S. and Schmidhuber, J., 1997), (Chung et al., 2014). These realities coupled with the increased knowledge of stock market wide behaviour and its relationship to psychology make the feasibility of profitable algorithms for trading a possibility for individual investors (Frydman and Camerer, 2016). This coalition of multiple academic fields - economics, computing and psychology - is likely to continue to make waves in the markets due to its importance and relevance to so many of us.

With big data influencing even the outcome of the US election, now is the time to see if it's ability to influence outcomes can be used for individual widespread financial gain. Data prediction and its use to inform decisions is becoming more prevalent as computer software and hardware continues to improve.

However, this is reducing the competitive advantage for all involved. It is now becoming a race to the bottom. As algorithms fight over smaller crumbs in a faster self-optimising market, new innovations in the fields of data processing and machine learning continue to be found and used to gain razor thin advantages. The volumes of money and impact these tiny deviations can have for large businesses can be huge.

For individual investors the rewards could be relatively better. Their actions make less waves than the large wealth investors – named whales - do. Individual investors are often behind this new computationally inspired approach. Their comparatively smaller skill and knowledge leave them far more susceptible to human emotions that negatively impact their decisions. Understanding the indicators, model architectures and parameters that still provide a competitive advantage ensure that innovative individuals can outcompete their neighbours. This edge can be further refined by the automation process involved, by removing human intervention poor decision making should be avoided.

The varied complexity of the methodology previously discussed in figure 1 and the subsequent misunderstanding of its ability creates a deviation between the reality and truth of its performance. Nevertheless, benefits have been seen and continue to inspire both individual and company data driven analyst that are hoping to change the tide in their favours in an otherwise ruthless and unforgiving game.

2.3 Economic Theory Overview

Economic theory is important when discussing any type of economically based project. This description only looks to provide information relevant to the particular topic at hand. Complexities discussed will be further explained. If further information surrounding this topic is wished: (Dickinson, 2013), (Sen, 1977). This topic is large in scope. The key parts to understand are as follows.

- Value is not what was required to produce something, rather what the highest bidder is willing to pay for that product (Ferguson, 2008).
- Available supply and demand impact price.
    - Substitution Effect and Income Effect (R Varian, 2014).
    - Law of Diminishing Marginal Utility (Rothbard, 2009).
- Value is not intrinsically controlled by currency as currency operates as a form of credit.
    - This means its value is not linked to a limited resource.
    - It operates as a system of supply and demand.
    - Currency devalues reduces purchasing power/unit of currency.
- Stock prices are analysed by three categories of company value.
    - Technical, Fundamental and Other (Li et al., 2018), (Lui and Mole, 1998)
- Stock markets are controlled by both algorithms and humans.
    - Algorithms can be difficult to interpret.
    - Human decisions are grounded in psychology and easier to infer.

Beyond general economic theory, the more relevant Efficient Market Hypothesis is critical to well established stock markets around the globe (Fama, 1970). It has a few types, the main points being that markets are constantly responding to new data. At any single point in time the collective information regarding a company and the wider sphere influencing its price are already factored into the price efficiently. This theory was first initialised by two economists who using technical indicators believed they were buying undervalued stocks. Rather than profit they consistently lost money. If the theory is true it would make the prediction of a stock impossible as all available knowledge is already factored into the share price valuation. Only new information could cause a change.

The further extrapolation of this theory was that markets are not predictable as they follow a random walk. It is theorised that over each time step the movements of a share are random and equally distributed in size. This would imply that over time every participant would perform the same (Malkiel, B. G., 2003). Evidence to the contrary was presented in 2008 and to some extent February 2020, experience and greater market knowledge separated those who benefited and those devastated by these two severe financial downturns. At these times the magnitude of the markets downward trends was immense when compared to the movement upwards over the time between these two dates.

These flaws are not to presume that these theories do not hold any weight. First arising in the 1970’s, they have been held as an economic mantra. This acceptance as gospel has become more foolish over time. Modern economics and the changes to procedure have put this solid economic theory on now shaky foundations (Burton, 2018). It does still hold weight because of the fact that people believe it. It is a difficult theory to refute due to its held importance by so many involved in the financial market.

The rapid adoption of computers has completely changed the landscape of modern stock trading. Bustling trading floors have been replaced with silent computer transactions. Information informing decisions is now largely sourced online (NW et al., 2019). Bots now perform the majority of trading by volume. Couple these changes with the wide spread oversaturation of data sources and we can see why components of these theories like the use of all available data is incorrect. This was magnified in the 2008 crash. Mortgage valuations were publicly available to those investing. The state of these mortgages dictated the valuation of Collateralised Debt Obligations (CDOs) – bundles of mortgages backed by banks – yet no one cared to look at the underlying data within these CDOs. Evidently the available information was not used efficiently by the market or these processes would have been discovered.

This theory also falls short on issues regarding its application. It assumes that particular procedures within the stock market process are immune to manipulation and that all investors have the same information. One example to the contrary is that of high-speed trading. When trading is performed a request is sent out to local exchanges, these local exchanges are then able to pair up people whom wish to buy and sell shares. When trades come in, they must be processed and then redistributed to the market for others to fulfil the order. High frequency trading firms can outcompete competitors due to better hardware and co-localisation to servers running exchanges (Virgilio, 2015).

This ensures that over time they perform statistically significantly better than their slower counterparts.

We see success stories in many businessmen who have consistently increased wealth over successive interaction with the stock market. Wide ranging disputes with the EMK and the idea of random walks have been presented for a large number of markets (Borges, 2010). These presented muddied findings with results presenting random walk being thrown into question regarding the statistical tests used (Liu and He, 1991). The main issue with these comparisons comes in the difficulty of proving non-randomness is occurring as many singular results can be claimed to be random. Any resemblance of distribution can be further explained by other factors in such a complicated system as the stock market.

Beyond these technical issues there lies a deeper problem. The majority of attitudes within the financial market are those of success. Optimists dominate markets. This is often the reasons for categorically disastrous downturns. This issue was exploited by Brownfield Capital over successive years to generate massive profits by taking unlikely negative stances on financial outcomes. People generally underestimate their chances of failure in a phenomenon named Normality Bias (Omer and Alon, 1994). It is grounded in our neurological and evolutionary development. As human standard of living improves, the expectation of negative outcomes from poor decision making is further reduced simply because it rarely occurs in this day and age.

All of these issues highlighted have changed our understanding after the EMH and Random Walk theories were proposed. This works to illustrate my final point. The algorithms within this thesis were created after these theories’ inceptions too.

The final remarks regarding these theories falls back into a discussing of psychology. People first taught these theories are of retirement age now. The economic landscape during that time period is far removed from the realities of that sphere today.

2.4 Technical Indicators

2.4.1 Introduction

Technical Indicators provide market data. Technical analysis is the processing of market data to inform decision making. It is one of the components of market analysis. It's utility in the field of markets is highly disputed (Park and Irwin, 2004).

Our main area of interest will be Technical Analysis and the Technical Indicators (TIs) it produces. Although not as deeply understood as other types of analysis this style of analysis is better for understanding trends and predicting turning points on stock movements (Lui and Mole, 1998). TIs provide deeper insight into companies and their setting within the broader financial market (Colby R.W., 1988). These TIs are used in groups to compound overall effectiveness. This further mitigates risk during financial trading.

As computation power has continued to improve machine learning models have done so too. This is in part due to the greater accuracy large volumes of data provide (Banko and Brill, 2001). They are now used in a wide range of processes as a functioning tool. They generally excel in the shorter term. This holds true for financial trading where high volumes of data are now produced.

Still, using a wide range of technical indicators quickly becomes computationally expensive to train, test and repeat modelling. This issue is further exacerbated when repeating models for statistical significance. For these reasons determining the most impactful features in stock market predictions is critical. Directly relatable work in these areas is difficult to procure, namely owing to working ones being incredibly valuable. Working ones cause a rapid 'alpha destruction effect' where more models predict these prices so the deviations between model and true price shrink. Many articles give primary research on the influence of different features (Kim and Kim, 2019) but the majority of academic research does not use LSTM models, nor do they do successive predictions.

Feature selection may explain which features are the best to determine stock price changes.

The influence will be because these features are the best true metrics for determining value. Still, they do not explain why these features elicit this response. These features and

the information they provide are a constituent part of why it is successful. However, it is important to note the impact blindly believing this indicator will be successful as this creates a self-fulfilling prophecy* (Fernandez et al., 2008).
More features generally perform better than less at predicting price(Nelson et al., 2017).

2.4.2 Common Technical Indicators

Common technical indicators used are provided in most API requests for data.

- Date, Open, Close, Day Average, Trading Volume
  - All utilised in daily time step form.

2.4.3 Derived Technical Indicators

Derived technical indicators are wide in range.

- Moving Average, Linear Regression, Relative Strength Index, Bollinger Bands, Fourier Transformations

Derivations to create these new TIs are explained below. With pros and cons described.

2.4.3.1 Moving Average

Moving average is a process by which previous stock data is used to approximate the predicted price of a stock over a novel time series. This processing ensures that large variability over a short time frame is negated. These moving averages can put varying weight depending on the distance between predicted weight and previous data points. By adding in Moving Average or moderating common technical indicators with it the variance of the test data is reduced. This operation is likely to remove overfitting during training.

Simple moving average uses linear weighting, exponential weighting gives greater influence to data points that are closer to the new points being created. The relationship is best shown by the equation.

$$S_t = \begin{cases} Y_1, & t = 1 \\ \alpha \cdot Y_t + (1 - \alpha) \cdot S_{t-1}, & t > 1 \end{cases}$$

Equation 1 Moving Average

This equation ensures that all numerical data is conserved. It operates on the assumption that novel data cannot influence the system. The influencing factors for stock price changes can manifest in unseen ways. This method of prediction struggles with accuracy when the markets momentum shifts. Given that these events where momentum switches are the greatest value creators, this processing has limited benefit. It can also be computationally expensive to create new data based on long time series averages.

2.4.3.2 Relative Strength Index

Measures current and historical strength and weakness of a stock based on two principles. That over time with no other changes the price of a stock maintains a consistent pricing. This derives from fundamental analysis and its price persistence over the short term. Secondly, changes in price up or downwards must be caused by overbuying or overselling respectively. The RSI can then provide insight into whether a share is over or under-priced. It is a true indicator of momentum in a stock and can inform the consensus of the market as either bullish or bearish.

Mathematically, the RSI is calculated by the following equation.

```
             100
RSI = 100 - --------
            1 + RS

RS = Average Gain / Average Loss
```

Equation 2 - Relative Strength Index

- There are 3 methods to calculate the Relative Strength (RS).
    - Simple Moving Average (SMA) => $\sum_0^n \frac{Gain_1 + Gain_n}{n} + \sum_0^m \frac{Loss_1 + Loss_m}{m}$
    - Exponential Moving Average (EMA_n) => $Cp_n \times M + EMA_{n-1} \times (1 - M)$

    - Cp = Closing Price, M = (2 / (number of observations+1))

- Wilder's Smoothing Method (WSM) => $(2n-1)EMA_n$ (Wilder, J., 1978)

Simple moving average provides equal weighting for each preceding data point. This has been showed to give poorer results than exponential average which weights closer data points more aggressively. This is important because data closer to the predicted day are much better indicators of subsequent price changes (Urquhart et al., 2015).

Lower values for RSI indicate a potential buying opportunity. The opposite is also true. Providing this data to algorithms has shown successful in improving trading performance over a range of ML architectures (Patel et al., 2015). Limited research has been performed directly for LSTM and GRU models specifically.

2.3.3.3 Bollinger Bands

Bollinger Bands are created from the moving average value and its standard deviations. These allow for clear insight into the deviation from the normal value as values drop to the lower band or rise to the higher one. Bollinger bands are useful in determining if normal price movement is not being adhered to (Patel et al., 2015).

2.4.3.4 Computed Indicators

We have discussed a wide range of indicators so far. Many have been created to deal with issues surrounding volatility, trade volumes, averages etc. Macedo et al used many of the indicators previously discussed with varying levels of success (Macedo et al., 2017).

Other methods of modulation have been around but have been severely underutilised in the field of stock market prediction. Share price is not indicative of the true value of a company because currency is now fiat, it is not backed by a stable commodity. This means that to base performance of a company against currency rather than value may be a poor decision and is likely to increase the error pertained to that system. Some key examples of factors that divulge currency from value would be inflation, government backed bonds or interest rates. All of these factors mean that a share price (X) at some time (t) has some value (V). In the proceeding time step X(t+1) -> V(t+1). This relationship is not linear over any considerable time period because of the previous factors. To determine the true change of value of a company it can be important to remove these controlling factors.

This also applies to environmental influencers that are in no way predictable. In March of this year. As the markets often do, it 'Took the Elevator' down, crashing drastically. Now that value was not taken by some hungry bankers and laundered into the Cayman Islands or the like. Rather the value of everything in the market stayed relatively constant. ######The pressures on companies caused by the normally limited number of shares was rapidly alleviated. At the end of this crash almost everyone participating lost equitable amounts of money meaning that the levels of easily liquidatable capital dropped too. Less capital was available, so products became comparatively cheaper. The share price drop improved the utility of money per unit currency and really little value was lost. This is merely here to highlight that value and currency are linked but not one in the same.

Additionally, the factors that influence change are also NOT static in their impacts on the market Inflation rates change daily in some places and so they cannot easily be extrapolated out from a company. Couple this with the fact that many multinational companies' valuations are governed by more than one market that is controlled in some sense by these factors. It becomes a complex soup. Trying to separate the complex unilateral relationships comes with its own array of difficulties.

2.4.4 Feature Extraction

Feature extraction is most commonly associated with Artificial Vision. Some of the insights it has provided theoretically can be mirrored in stock market prediction. Some examples would be the extraction of detail from an image. A visual based system is picking out particular components or characteristics of an image to inform its decision about the contents. A similar process is used in natural language processing tasks where each word is given a particular numerical representation called a embedding. In our task we are using our deep layers to try to extract features and representations about the companies from their historic price data. These features may give insight into any features of the companies. These features may be comprehensible, company ethics, work efficiency, relative sector performance etc. They may be completely undecipherable for us. However by including these hidden layers it is likely to give us a greater insight into the company and its future than simply not including hidden layers.

2.5 Fundamental Analysis

Fundamentals are involved in the company specifics. These include:

- Company structure
- Board Members
- Company Nature
  - Ethics
  - Practices
- Assets
- Relationships
- Numerical Data
  - Out with stock prices

Fundamental Analysis does not include easily predictable decisions. This is because it is so deeply understood. There is no real way to capitalise directly from fundamental analysis. Research involved in grading companies based on their fundamentals and using this to inform long term monetary gains has been shown to be fairly reliable (Shaughnessy, J. 2012). It's relevance to day trading is negligible.

2.6 Other Analysis

2.6.1 Introduction

Other details come in the form of company reports, external news stories and speculative articles written by individuals and companies alike. This style of analysis is the newest and most unknown. The wild west nature of the internet coupled with the rapid availability of swaths of information make predictions using this largely untrusted. Much work has to be done to improve this field into one with well developed practices for improving the reliability of insights gained.

Financial markets are odd, they predicate themselves as mystical complex machines. People believe this, the rapid changing landscape forces people to concede knowledge to have a chance of keeping up.

> *"People want an authority to tell them how to value things, but they choose this authority not based on facts or results.", M J Burry.*

We know through experimental research that the performance of hedge funds and managers is actually worse than self-investment through a number of index funds (S. Tepper, 2017). The commissions and poor decisions these companies take gives worse performance over time.

New forms of analysis are the new kid on the block. Wall Street loves a new exciting reason for it to validate the complexity of trading. Quantitative minds are now in high demand to build algorithms for profit. This area of finance has showed promising returns for consumers (Li et al., 2018).

2.7 Deep Learning Analysis

2.7.1 Introduction

Deep learning has existed for a long time, building upon itself. This increase in complexity has been driven by solutions to issues found within the practical application of the technology. Discussed further are the major improvements up to this point in Deep Learning as well as other factors that have continued to improve the field in its utility. This information pertains to Figure 1 Component 4.

Deep learning is a great tool to create intelligent models. Its theoretical basis originates from how learning occurs within the brain. Particular inputs cause downstream activation of subsequent interconnected nodes and results in an input output functionality. These functions vary in complexity and have become more sophisticated as experimentation and usage becomes more common (Skansi, 2018).

2.7.2 History

Early examples were created by Warren McCulloch and first written about in 1949 (D. Hebb, 1949). From this inception improvements have been continually added to this simple prototype. Arthur Samuel continued this with a simple checkers game. By remembering the subsequent outcomes of previous situations, it was able to 'learn' and improve its technique (Samuel, 1952). This technique was further emulated in 1997 with Deep Blue (Campbell et al., 2002).

In 1959, neurophysiologist D Hubel and T Wiesel discovered two cells within the visual cortex. These simple and complex cells were interconnected with many cells in a hierarchal style and were modulated by the summation of signal voltage causing activation and subsequent propagation of the signal. The architecture of these cells was used to create artificial neural networks that laid the next level of complex convolution after the single cell analogy. "All or nothing" activation functions are akin to binary functions, although this relevance was overlooked at the time.

From here it became apparent that prolonged cellular stimulation was causing reduced responsiveness to the same stimulus. This was rooted in drops in the concentration of polarised ions from depolarisation. From here the idea of modulation was introduced. This reduction in efficacy over time allowed cells to alter their properties in an inverted fashion that ultimately resulted in the idea of backpropagation of information decades later (Goodfellow, Bengio and Courville (2016, p. 221).

K Fukushima was heavily influenced by the work and appropriately tried to emulate the functionality of the simple and complex cells residing in the visual cortex. He used convoluted neural networks to recognise visual information (Fukushima, 1980). This was another example of training without explicit instructions. Sets of rules used over specific direction still led to correct interpretations. This was ground-breaking and demonstrated that abstraction of meaning formed what is perceived as intelligent decision making.

After this work, the field was reinvigorated by J Hopfield with the creation of a recurrent neural network. This was the first example of data output recursion. Recursion benefits from sequential data usage and allows for the extraction of meaning contained with the ordering of data. Recursion is also computationally advantageous as it allows iterations of data input (Sherstinsky, 2020). This styling is important for time series technical data and language-based information. It allows accurate context to be extracted and was a stepping stone to the popular LSTM and GRU networks built today.

With great man power and interest the field of intelligent processing now contains a huge array of deep learning algorithms. These exist and outperform humans in many activities (Liu et al., 2017). This is due to leading academic and industry professionals noticing their effectiveness in a plethora of tasks and allocating resources appropriately. For example, in 2014, the program Deep Mind beat players of Go. This program was then acquired by Google for £400 Million. AI and its foundations continue to be implemented at scale over numerous industries to improve outcomes of all manner of tasks. Many caveats of Deep Learning continue to be altered in the hope of improvement.

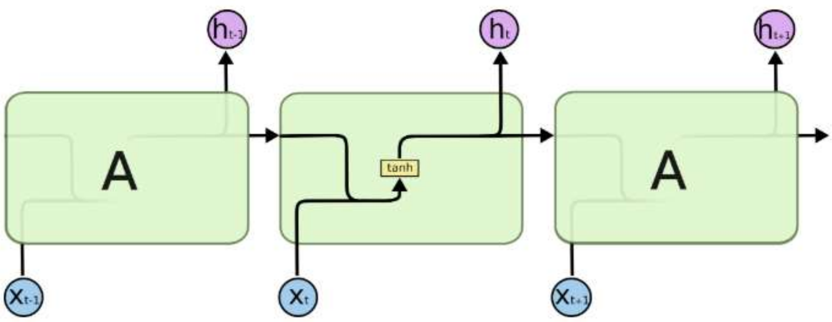
Chapter 3- Introduction to Theoretical Background

Now that former research and work has been discussed we will further explain how the methods used began and developed into those implement in this thesis. The important theory here discusses computational components only. A deeper knowledge in these highlight further areas of research.

3.1 Models

Deep Learning is used for a wide variety of intelligent tasks. The theory underlying their effectiveness will be discussed. Adoption of techniques focusing around other areas will impose the universality of these algorithms. Their utility, which can improve their effectiveness, will also be analysed in depth.

The models built operate on simple processes arrange in complex architectures to produce amazing results. The models described in this thesis are Long Short Term Memory (LSTM) and Gated Recurrent Units (GRUs). Both are based on Recurrent Neural Networks (RNNs) (Bianchi et al., 2017).



The repeating module in a standard RNN contains a single layer.

Figure 2 - "Vanilla RNN Model" Each X value corresponds to a 2D input vector. This is true for all models described. These vectors have individual weights for each component in 2D. Additional Information behind the matrix mathematics used for weights updating can be found (StatsSorceress, 2017).

https://colah.github.io/posts/2015-08-Understanding-LSTMs/

RNNs are sequences of cells. Each contains a hidden state (h_t) and an output (h_{t+1}). At each time step the hidden state is calculated using:

$$h_{(t)} = f\left(h_{(t-1)}, x_t\right)$$

The engaged cell takes the data it is provided (x_t) at one time point and then concatenates that to the output of the previous cell (h_{t-1}). This combined data (in the shape of a matrix) is then processed through some activation function to produce an output for the cell. This process is then repeated over the number of time steps for the operation. Each cell is a summation of the previous ones.

Gradient descent is used to reduce the error between expected and the actual output. These are defined as optimisers in the literature. Learning is then implemented by backpropagating new weights for each operator and function between layers. Each node is connected in sequential order so the initial ones the start of the model receive little impact from this style of learning.

This means that long term relationships between early nodes and layers are not found even over a large number of epochs. We encapsulate this feature of Recurrent Neural Networks by stating they lack a cell state. This causes an ineffective backpropagation of information named the vanishing gradient problem. This is a lesser issue in problems involving close proximity relationships like word prediction. In stocks markets many factors influencing the price are long range and cyclical (Yuksel and Bayrak, 2012). This feature of stock market prediction means it requires a different approach.

3.1.1 Long Short Term Memory (LSTM)

To overcome this problem a cell state was implemented. In 1997, a new style of model named LSTM was proposed. It managed to remove the vanishing gradients problem previously occurring in RNNs by including a cell state. This ensured that data with long series' and many features could be more appropriately modelled to find trends and important relationship contained within the data over time. The cell state was implemented using the internal architecture shown below:

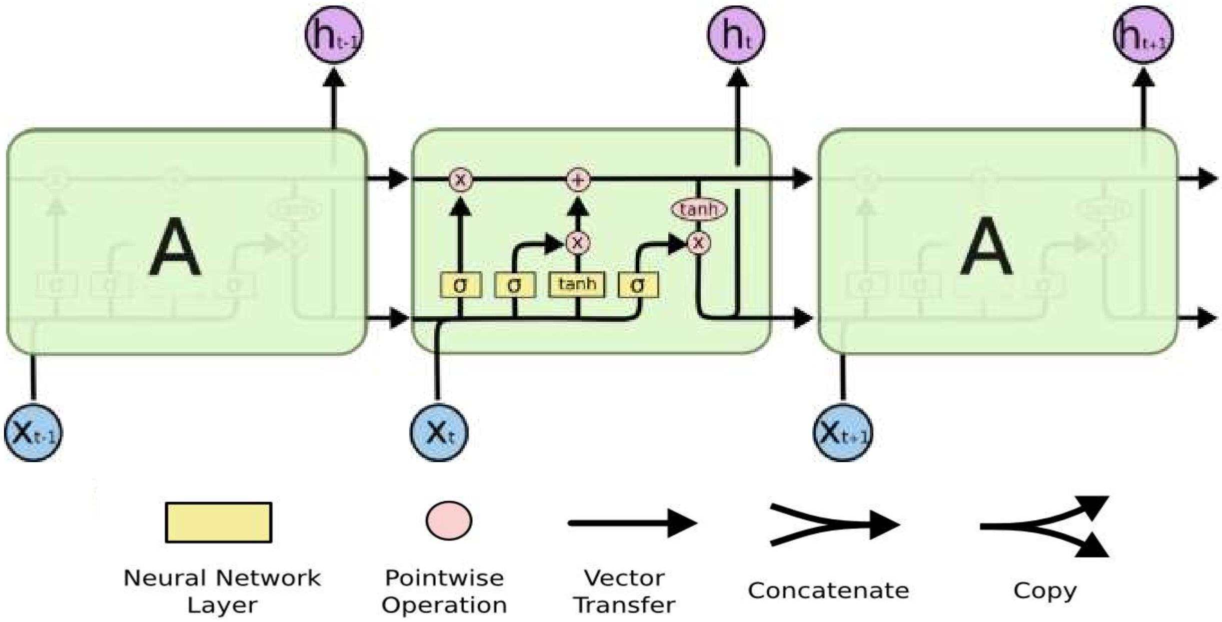


Figure 3 LSTM Model Single Layer Cellular Architecture

https://colah.github.io/posts/2015-08-Understanding-LSTMs/

The model consists of a series of cells that connect in a time dependant manner (denoted by the subscript "t"). This cellular architecture is comprised of Neural Network Layers (NNL) and Pointwise Operators. Numbered from left to right NNL#1 acts as the forget function. NNL#2&3 create the Input Gate. NNL#4 produces the cellular output.

Each component contained within the cell provides information to a key function. The top horizontal line acts as a conveyor belt for information. It is tightly regulated, ensuring that only important changes alter the specific cells state. The cell stores a hidden cell value and a cell state. These allow for both the long and short term relationships to be determined by the internal architecture.

3.1.1.1 Calculating Forget Function

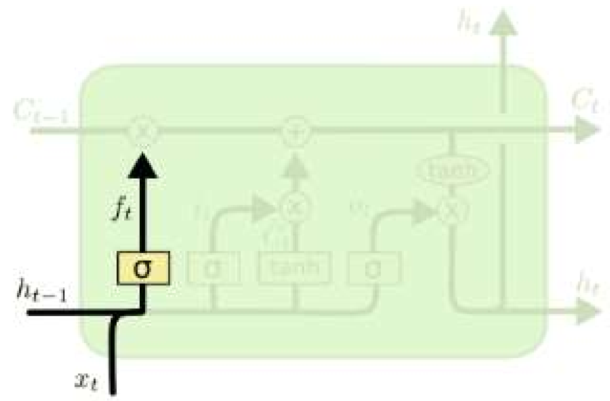


Figure 4 Forget Function – Equation:
$f_t = \sigma\left(w_f * [h_{t-v}x_t] + b_f\right)$

https://colah.github.io/posts/2015-08-Understanding-LSTMs/

The previous cells output (h_{t-1}) is concatenated with the value for this time step (x_t). This data is then fed into the sigmoid activation function. The forget gate (f_t) outputs a value between 0 and 1. This value is used to determine the proportion of the previous cells state should be maintained.

3.1.1.2 Calculating Input Gate Value

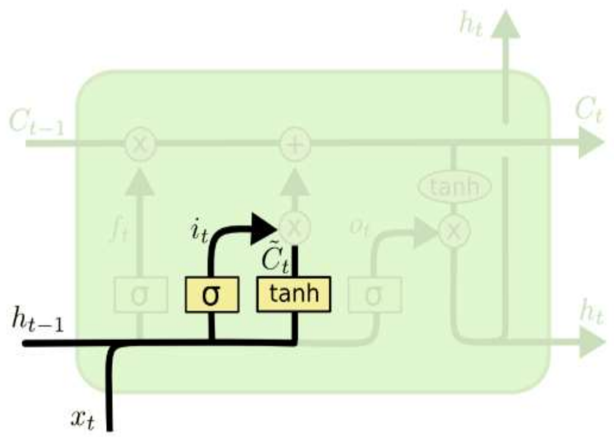


Figure 5 Input Gate of LSTM model –
$i_t = \sigma(w_{i^*}[h_{t-1}, x_t] + b_i)$
$\tilde{c}_t = tanh(W_c * [h_{t-1}, x_t] + b_c)$

https://colah.github.io/posts/2015-08-Understanding-LSTMs/

The same initial data is then fed into another sigmoid layer named the input gate layer producing i_t. This value is then summed with the output of the tanh function C_t (a scaled sigmoid) that is some value between -1 and 1. This relationship is summarised in Figure 6.

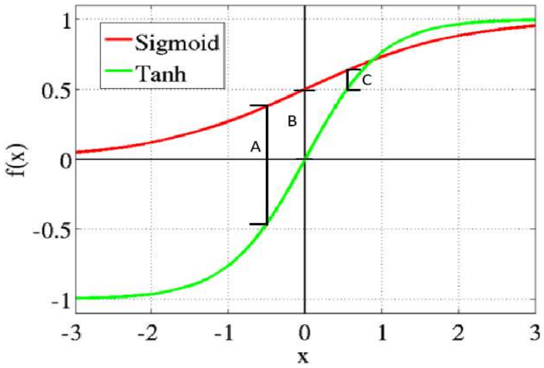


Figure 6 Input Gate Activation Function.

Note that x is the value fed into the neural network layer; not exclusively the input value x_t as shown in other diagrams.

https://towardsdatascience.com/activation-functions-neural-networks-1cbd9f8d91d6

This mechanism ensures that as i_t and $\tilde{c}_t$ tend towards negative (as shown at A), the output for the change in the cell state is larger in magnitude (when compared to B or C in Figure 6). This works well for stocks and shares. The ability to determine direction change is more important than the magnitude of the change for wealth generation. Markets are generally rising over time so increasing importance on negative values is important.

3.1.1.3 Updating the Cell State

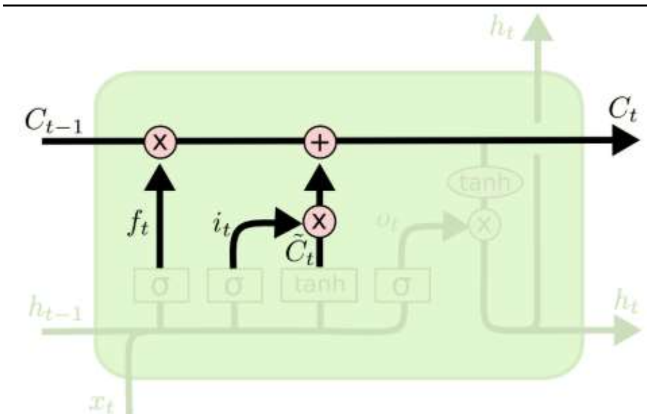


Figure 7 Updating the Cell State:
$$C_t = f_t * C_{t-1} + i_t * \tilde{C}_t$$

https://colah.github.io/posts/2015-08-Understanding-LSTMs/

Now we have performed these methods we can use h_{t-1} and x_t to determine this cell's state. We achieve this using by summing the previous cell state with the forget function.

This only preserves a proportion of the previous cell state's value depending on the forget gate output and the point wide multiplication applied to C_{t-1}. After this the output of C_t and i_t, the Candidate Value, is added to the cell state. This allows information pertaining to the specific data point x_t to be fed into this cells state. This new cell state is then fed into the next cell to act as the baseline.

3.1.1.4 Determining output (h_t)

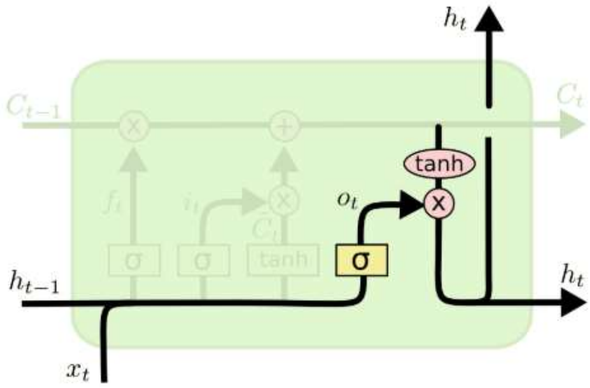


Figure 8 Determining hidden state:
$$o_t = \sigma\left(w_o\left[h_{t-1}, x_t\right] + b_0\right)$$
https://colah.github.io/posts/2015-08-Understanding-LSTMs/

Finally, we determine the output of the cell. This data is passed in two directions:

- To the next layer – (upwards in the diagram)
- To the next cell – (rightwards)

Here we take the cell state from the upper most path and we run this through a tanh function to gain a value between -1 and 1. The newly created data is then summated with the value of the outcome of the [h_{t-1},x_t] and sigmoid function (o_t). This provides the cells hidden state (h_t) that is passed on as described in Figure 8.

All of these operations are examples of forward propagation. These ensure that the outputs created by the inputs can be used effectively as inputs in the subsequent nodes and layers of the network.

3.1.2 Gated Recurrent Units (GRUs)

Gated Recurrent Units (GRUs) are another popular derivation of RNNs. It was first proposed in 2014 for semantic and linguistic processing. The intercell and layer connectivity is the same as the LSTM architecture. It has two gates (reset and update) rather than LSTMs three gates.

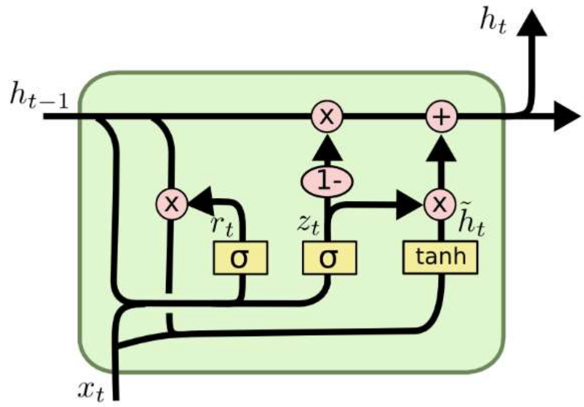


Figure 9 Internal Architecture of GRU

$$z_t = \sigma(w_z[h_{t-1}, x_t])$$

$$r_t = \sigma(w_r[h_{t-1}, x_t])$$

$$\tilde{h}_t = tanh(w \cdot [r_t * h_{t-1}, x_t])$$

$$h_t = (1 - z_t) * h_{t-1} + z_t * \tilde{h}_t$$

https://colah.github.io/posts/2015-08-Understanding-LSTMs/

h_{t-1} is first multiplied by r_t (in the second vertical line downwards). This is the only non-concatenated vector used in GRUs. Proceeding this h_{t-1} and x_t are vector concatenated before any more processing occurs. These values are then used to produce the multiplication and addition factors. The conveyor persists and maintains an addition factor preventing vanishing gradient issues.

3.1.3 Model Comparisons and Similarities

Both of these subtypes of algorithms (LSTM & GRU) have a huge variety of other changes that have been proposed, tested and evaluated. They have varying performance in different areas depending on the goal attempting to be achieved yet they all seem to generalise, giving similar performance (Greff et al., 2017). They also suffer from new issues such as the exploding gradient problem if there are persistent gradients in a consistent direction.

LSTM and GRU models are able to be bidirectional, multi-variant, have/omit dropout, be easily transferable, perform multi-thread operations, produce an array of output values, be stateful/stateless, incorporate multi-input information and maintain some type of memory over the immediate and longer term. The models benefit from these attributes and its proven performance give this model real weight in the competitive sphere of Neural Nets (Nelson et al., 2017). Unfortunately, the mechanisms by which particular algorithms are deemed successful has no standardisation. Experimentation of a large number of common RNNs was performed and found that GRU and LSTM models performed similarly in time series tasks (Greff et al., 2017). It should be noted that GRU models were considerably quicker to train than LSTM models.

3.2 Model Learning

A model learns by navigating the search space. This process is named gradient descent. The search space is a multidimensional area. It navigates by altering the strengths of connections between nodes in different layers. These relationships are known as the weights within the model. The co-ordinates of the space are the weights within the current model. These give a value for the performance of the model. The space is usually multidimensional but is most simply comprehended by imagining a three-dimensional space filled with hills and valleys.

In our picture we only have 2 weight values – represented by the $ø_0$ & $ø_1$ co-ordinates. In

Search Space Representation

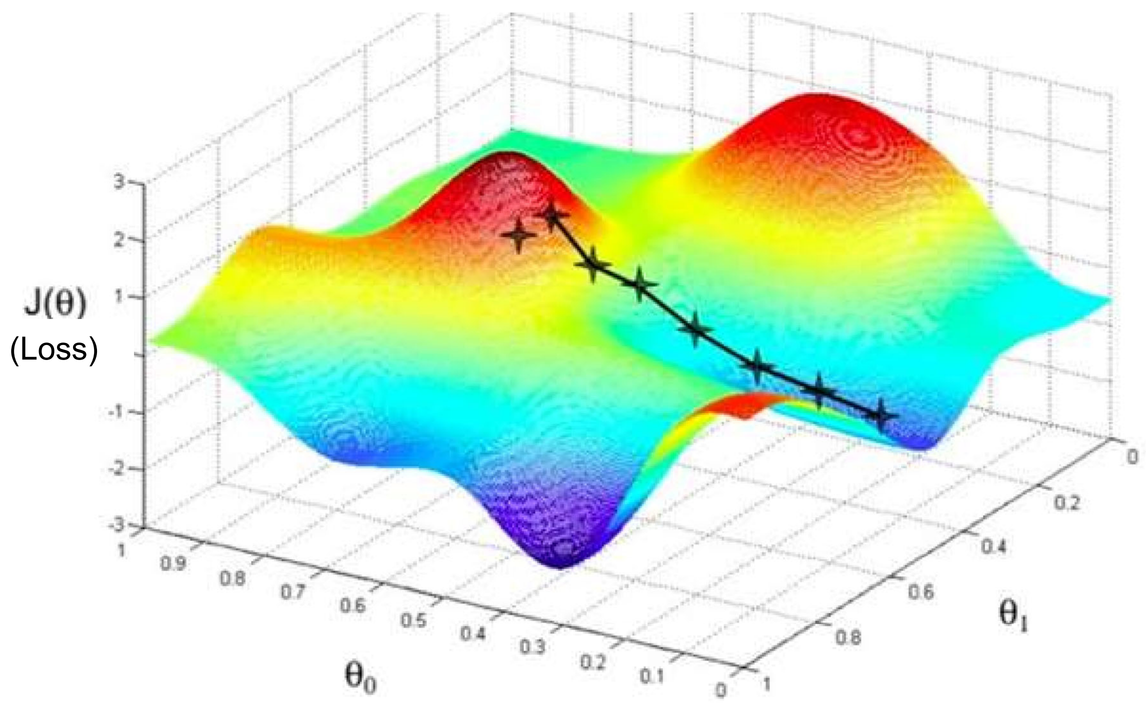


Figure 10 A representation of a search space. Depicted is the actions of a gradient descent mechanism where the loss function is optimised. *https://blog.ramith.fyi/static/basics-matter-scholarx/*

practice the dimensionality is much greater and so not diagrammatically representable. The J(ø) co-ordinate of the search space gives us a numerical representation for the performance of the combination of weights at that particular point. This performance value is determined by the optimiser. More information regarding the pure mathematics can be found (Kang and Duk Seo, 2019).

3.3 Adam Optimiser

When building a model there are varying ways in which we can determine how error is translated into changing weights within the model. These are named optimisers. By using an appropriate optimiser, we can ensure that our models learn quickly and appropriately based on the challenge trying to be overcome. Optimisers operate by utilising derivatives to approximate error in the system. This information is backpropagated through the system and serves to keep the weights of the system constantly adapting to its training.

Adam is the particular optimiser used during my experimentation. Adam is a coalition of two extensions of stochastic gradient descent (SGD) that implement matrix multiplication to determine weights over training epochs (Ruder, 2017). Adam outcompetes conventional SGD methods for stock market predictions by altering how the optimiser functions.

- Adaptive Gradient Algorithm
    - Adapts learning rate (step size) dynamically based on the frequency of parameter values (Kingma and Ba, 2017).
        - Greater learning when uncommon behaviour occurs.
        - Places greater importance on bullish proportions of the market and drastic changes in companies market valuations
        - Fits well as the impact of uncommon behaviour is far greater than that of more normal market movements.
    - Determines learning using first order derivatives.
        - Reduces memory usage and training time for complex 3D inputs.
        - Can handle low magnitude step change gradients.
        - Scalable
        - Works well for small magnitude predictions coupled with high volume training data.

Importantly we are avoiding higher order computational optimisation methods – like Newtons Method (Kovalev et al., 2019) - because of their computational infeasibility when tackling large data sets. In time it may be possible to implement these, receiving their benefit at a relatively reduced cost.

- Root Mean Square Propagation
    - Set learning rate based on changes in gradients between data (uncentered variance).
        - Ensures that noisy data is less impactful than true changes to values.
    - The changes detailed are the EMA of the gradient and the gradient squared (g^2).
        - Decay rates are summed on this moving average.
        - We then calculate the estimates and the true values.

The performance of this algorithm over large epochs and high-volume training data sets is good. This process of optimisation often leads to slowed rates of improvement over epochs (Luo et al., 2019). New optimisation methods continue to be produced that are likely to be better performers in niche areas of development.

The wide scale adoption of novel optimisation methods usually coincides with their implementation in deep learning/data analytics libraries. Before this occurs, it is unlikely that even great new optimisers will be seen and adopted by the wider community (Reddi et al., 2018). New optimisers are often discussed in relation to classification tasks, or binary outputs (Up or Down in price). This may mean that their optimisation performance on a continuous output value task may be less effective.

By controlling how information is backpropagated we can determine the attention focused on particular portions of the time series data used to train. This method of dynamic importance ensures that key areas of the stock market price change and adequately included when training our model quickly, accurately and reliably. Optimisers work by navigating the search space. The inception of optimisers is a topic large in scope (Sun, 2019).

3.4 Hyperparameters

Hyperparameters are parameters that alter the learning process of the model. Here a selection of hyperparameters have been chosen to try and improve the performance of our model. Statistically determining the best configuration for this particular task should improve the performance of our model. This in turn should theoretically lead to a more profitable software program.

3.4.1 Dropout

Dropout is the random removal of nodes within a neural network architecture. The theory behind this is that by removing particular nodes at random during the training process we prevent any nodes from dominating the performance of the model. These particular nodes can prevent further searching of the space for more optimal node biases. Linking back to our Model Learning section.

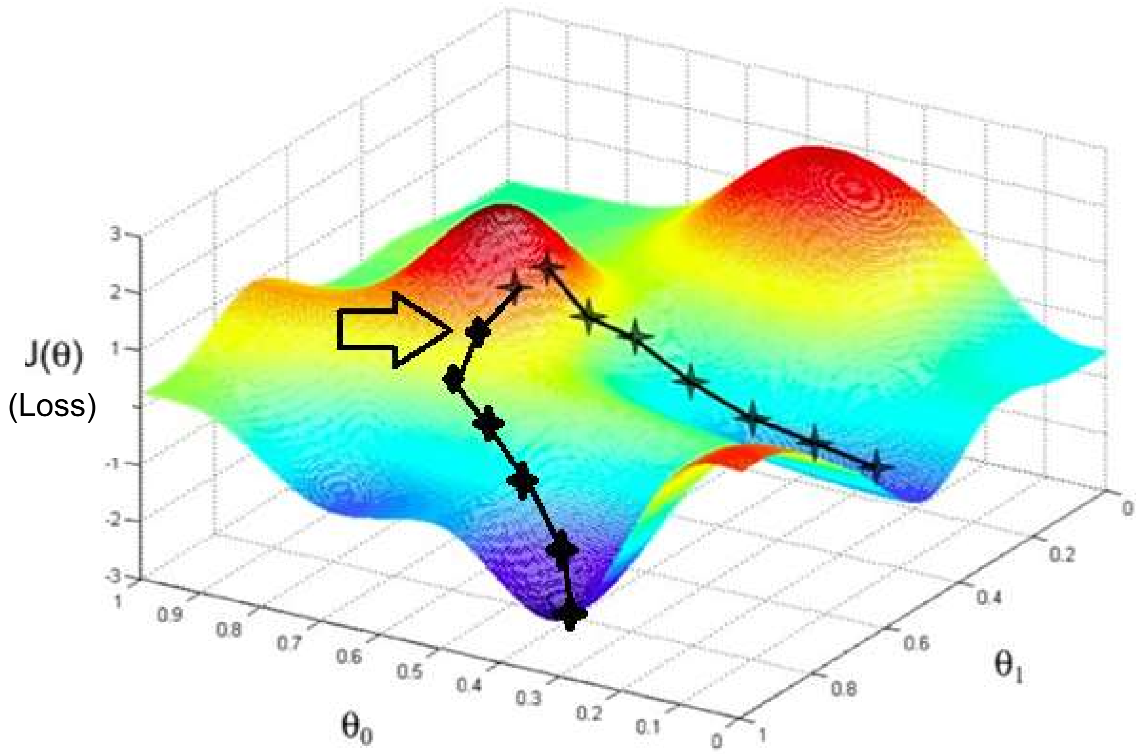


Figure 11 Showing difference in descent path caused by dropout of $\varnothing_0$ for the first epoch.
https://blog.ramith.fyi/static/basics-matter-scholarx/

In Figure 11 if we dropped $ø_0$ for one step its value would not change for that step. This could cause the first step taken to be as shown by the arrow. By moving only on the $ø_1$ axis our model ultimately finds a better minima value than the other path.

3.4.2 Batch size

Batch size determines after how many data points new information is backpropagated to change the model weights. When picking batch size it is important to understand how they impact learning within the search space previously described in section 3.2.

The larger the batch size the greater amount of data we use to update our co-ordinates within the search space. This means that the step we take within our search space will be larger in value. Over the long term this will lead to a continual improvement in performance because there is a large space for the multidimensional co-ordinates to search. The training is less directed towards a local minima because it is likely to overstep and miss them. Searching of the search space is more sparse. When training on smaller batches the opposite occurs, the local search space is more closely searched. This leads to a faster convergence to a minimum but is more likely to find a local minimum rather than the global one.

Testing batch size as values of powers of 2 allows an increase in training speed. This is caused by GPU parallelisation during the matrix multiplication that teaches the model. By increasing the batch size we make the step within the search space of our program larger.

3.5 Statistics

Statistics is a key component of Deep Learning Algorithms. Fundamentally gaining insights is achieved through mathematics and the correlated properties of the data processed. We implement statistics throughout the processing of our model to re-evaluate the weights.

Initially prediction models used to perform predictions used regression models. The simplest being linear regression. These are represented as straight line relationships. To represent more complex non-linear relationships regression models were first used.

3.5.1 Calculating Error

Other utility of statistics within our thesis is the calculating of error within our system. We achieve this in two ways. Mean Squared Error (MSE) is a method to calculate the accuracy and error within our predictive model.

$$MSE = \frac{1}{n} \sum_{n=1}^{n} \left(Y_i - \hat{Y}_i \right)^2$$

Equation 2 Mean Squared Error. Used to determine performance of the prediction model. Where $\hat{Y}$ is the predicted value and Y the actual value.

Mean Squared Error is a helpful determinate for performance but it lacks in weighting large deviations more severely. To counteract this Root Mean Squared Error (RMSE) was also calculated.

$$RMSE = \sqrt{\frac{\sum_{n=1}^{n} \left(Y_i - \hat{Y}_i \right)^2}{n}}$$

Equation 3 Root Mean Squared Error. Improvement of MSE. Large deviations are more critically valued. Where $\hat{Y}$ is the predicted value and Y is the actual value.

https://towardsdatascience.com/what-does-rmse-really-mean-806b65f2e48e

3.6 Automation

The process of automation is key in financial trading. The volumes of data now produced regarding companies is so immense in scope and complex in nature that human processing is simply lacking.

3.6.1 Required Computation

A key reason for automating our system is to run successive experiments to improve the reliability of our results. Saving and recording our data automatically improves performance and removes the tedious parts of developing algorithms and maintaining records around the work performed. Being able to convert the loss values our model has with real true world profit values is an added benefit. Beyond this, removing human intervention within trading should help to improve the outcome as it removes poor human judgement from impacting the actions of the created program.

3.6.2 Psychological Limitations

Additionally, humans are animals. Our decision-making abilities evolved over our species entire history. A large proportion of the pressures that drove this evolution are far devoid from the realities of our lives today. For this reason, our actions are often outdated. Here we can see the requirements that dictate our needs.

Figure 12 Maslow's Hierarchy of Needs – shows human requirements and the order of their importance. *http://www.simplypsychology.org/simplypsychology.org-Maslows-Hierarchy-of-Needs.pdf*

Decision making concerning money affects the entire hierarchy of needs. Due to the way in which society is structured the lower tranches of the pyramid (labelled needs) are entirely controlled by money. This cause emotion to bleed into decisions involving it. A huge number of psychological phenomena are shown with respect to money. Two common misconceptions are the Gamblers (Howel, D,. 2010) and Hot Hand Fallacy (Johnson et al., 2005). Other examples of money manipulation include Anchoring, Bartering theory, among others (Thaler, R. and Sunstein, C., 2009). Many of these theories are encapsulated in derived technical indicators. These psychologically explained phenomena are key in why removing human activity is a financial intelligent decision.

All of the statements do not validify the idea that machines don't make mistakes. In fact, examples of machine exacerbated errors are prevalent in financial markets (Kirilenko et al, 2011). The key reasoning for following this side of the argument is the severity and number of human induced issues within the market.

It is also important to be aware that the prevalence of human error may be linked to the larger samples of time that the market was under human control. Up until now, the current information surrounding this supports automation. This may change in the future, although this is speculative.

Chapter 4- Methodology

4.1 Software

Python 3.7.4 was used for all of the tasks involved in our generalised methodology shown in Figure 1. This was chosen due to its superior machine learning libraries and ability to perform as a great toolkit for most processing. These libraries are well vetted and constantly updated with purposeful documentation. Code was written in Jupyter Notebook. The open source package management system Conda was used making installing different dependencies in localised directories easy.

The data was attained using Alpha Vantage API and processed using Pandas and NumPy. The model was constructed using the Keras API that operates with a TensorFlow backend

Statistical processing was performed using Microsoft Excel and its associated Data Analysis module. The huge range of variables and parameters (grouped under the name metrics) can be implemented when building Recurrent Neural Networks. To gain any statistically reliable insight a proportion of metrics must be held constant. This allows others to be analysed with respect to performance. All metrics not discussed are left as set by the Keras API as described (https://keras.io/api/models/).

4.2 Data Acquisition

Data regarding the companies was sourced as depicted previously in Figure 1 Component 1. The common technical indicators were acquired using an API and presented as a Pandas DataFrame as shown in Figure 9.

	date	1. open	2. high	3. low	4. close	5. volume
0	2020-03-26	6.38	7.3200	6.130	7.28	4119272.0
1	2020-03-25	6.41	6.4500	5.520	6.23	4320762.0
2	2020-03-24	6.46	6.8500	5.925	6.07	6252432.0
3	2020-03-23	5.79	6.1400	5.280	5.93	4336064.0
4	2020-03-20	5.91	6.1800	4.900	5.63	6515815.0

Figure 13 Common Data presented in Pandas DataFrame.

4.2.1 Description of Data Features

- Date: The date of that trading day
- Open/Close: The price of the stock at the start/end of the business day
- High/Low: The highest/lowest price the stock had that day.
- Volume: The number of stocks traded during that day.

These indicators allow the entire process of stock market prediction using technical indicators to occur (Figure 1 Component 1). These price descriptors via their coefficient of determination score have all been shown determine an accurate outcome (Ahmed et al., 2018).

4.3 Data Derivation

After this Technical Analysis (Figure 1 Component 2) was able to be performed. By performing this we can achieve a plethora of benefits. These altercations can improve the data's capacity to train, its representation of true value and its data transformation capability. These improve the quality of the data available for deep learning models (Figure 1 Component 4). All data derivations are easily marked in the source code.

4.3.1 Mid Price

Data is then optionally altered with a mid price. This was the mean of the open, close, high and low prices. This was performed because the open and close prices of shares can be highly volatile. This volatility can reduce the predictability of pricing and over time increases losses. This trading is primary caused by HVT bots that over time lead to losses for individual traders. By performing this we are able to determine a more reliable price that is also likely to be reached on a predicted day. This mitigates and allows automated buy or sell orders to be created.

4.3.2 Relative Strength Index (RSI)

RSI can be optionally included in the data. Its works by determining overbuying and selling of shares by comparing the relative gain and losses in previous day. This may benefit the model in determine price changes as more data generally achieves this.

4.3.3 Bollinger Bands

Bolling bands are commonly used by investors. These bands are able to determine the volitivity of pricing and therefore a deviation from the normal volume of transactions. Calculated from the exponential moving average they are used to determine when shares with spike up or downwards.

Additionally, Bollinger bands are able to scale the current price data more effectively by increasing the feature range by some degree. This improves the operations of sigmoid functions and removes redundancy.

4.3.4 External Normalisation

Another potential method we can implement is Inflation Normalisation. Older lower valuations of companies are not indicative of a less valuable company. We can remove Inflation as a factor to try and improve the impact of older company data. This gives the model more realistic value changes over time.
This process can also be extending to other currency modulators like government backed interest rates or bond return on investment (ROI) values.

4.3.5 Company Normalisation

This method looks to alter company data by normalising it with other companies' data. The grounding for this theory is that by combining two companies we are able to eliminate non company specific features. What is left is a more direct representation of a company.

This is important with regard to true company value as previously discussed. By picking comparatively better company's better performance can be achieved. Often this is all that is needed as outperforming the market is a game of relativity and not absolute profits.
We can instead compare our companies to other companies within the same domain to further isolate its individual performance. Alternatively, we can compare it to a standardised metric (NASDAQ 100) instead to gauge its general performance. Striking the balance between leaving individual performance and removing market wide performance from our data is important as it isolates the company rather than the noisy market activity encompassed in the share price.

4.4 Deep Learning Processing

Once a complete data set has been produced, we can move onto data processing. This is a key step and ensure that the models receive the best possible data. This process should improve the appropriate insight from the data used.

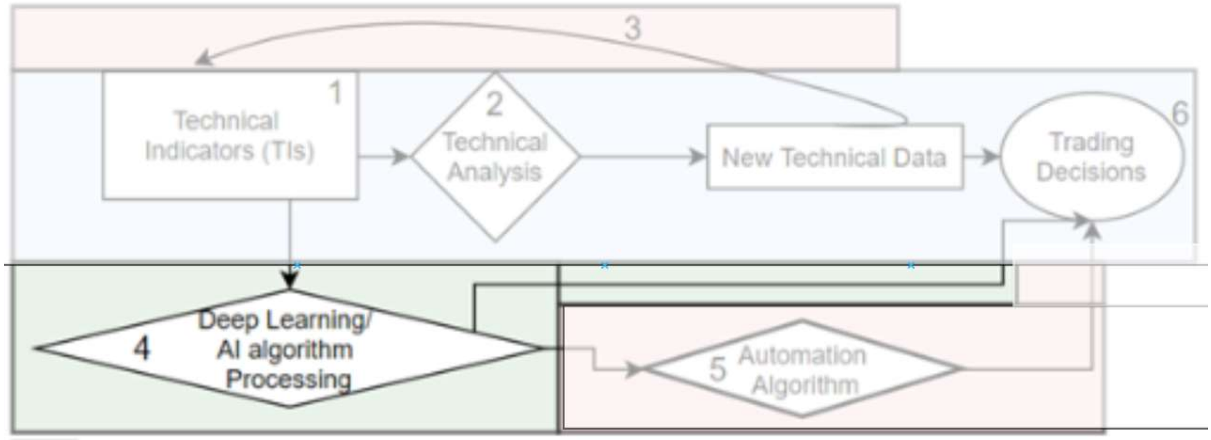


Figure 1 reproduced and focusing on Component 4.

Here we are discussing the transition from TIs (Component 1) to Deep Learning Processing (Component 4) into the deep learning processing.

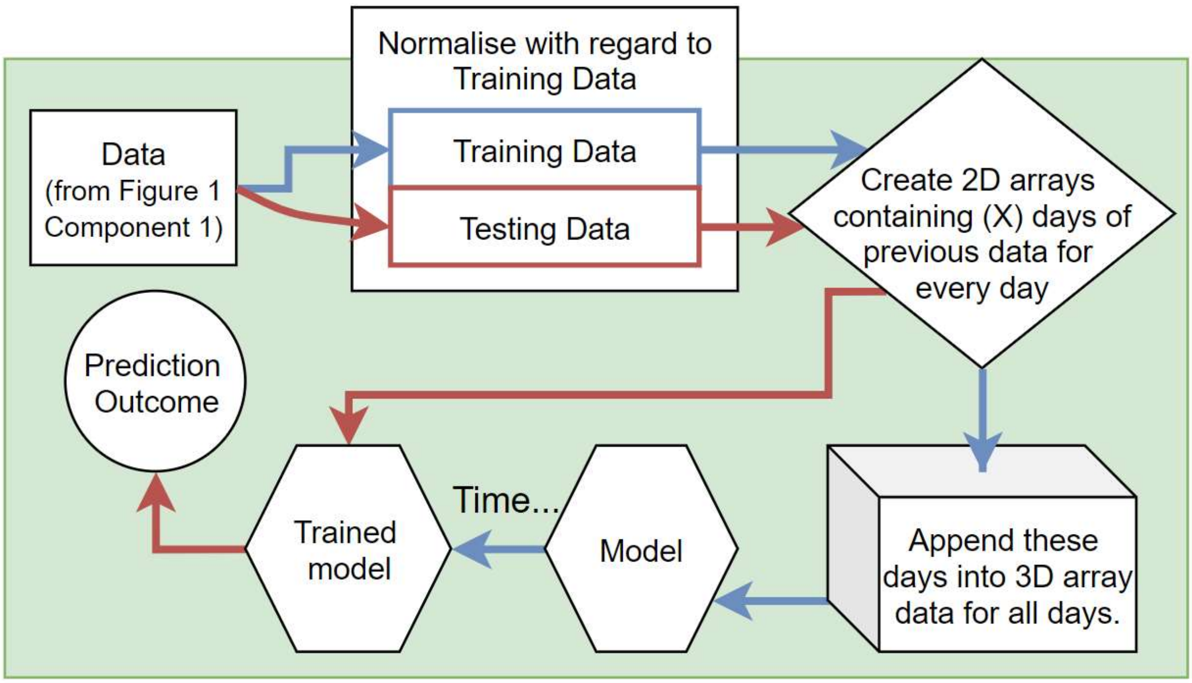


Figure 14 The internal processes of Figure 1 (Component 4).

Figure 13 shows how data is processed at a high level to feed into our algorithm. From here the Trained model is used on the Testing Data to determine its ability with novel data. The blue arrows are implemented first and followed by the red arrows.

First the data is split into training and testing sets. The data is then normalised with respect to training data. This data is then converted into 2D sheets of previous days data in time order. This is then converted into a 3D array (shown in the next figure). Here the model is then trained on the training data only. Then the testing data is used to test the ability of the trained model. The prediction is then created. From here this prediction gives particular performance metrics. These and the prediction and passed onto the Automation component of Figure 1.

4.4.1 Normalisation

We can normalise the whole data set between its smallest and greatest value (conventional) or by selecting windows of data and normalising these windows independently. There is little evidence-based consensus on these data processes. Namely due to the range of applications for models and huge number of variables involved. Normalising with respect to training data ensures the test data does not inform any part of the training process. We achieve this using feature ranging, where data is normalised between two values (0, 1). Using these values ensures that data is handled appropriately by then internal cell architectures.

At the time of writing, the current direction of the market is not overtly positive. This would be named bullish in ‘market speak’. It is important to note that in a bullish market it is likely for predicted share prices to continue rising in price out from within this normalised range. This will cause new prediction to be above 1 in value. This can be detrimental to predictions (see Figure 6), especially as the length of the prediction increases.

It may be better to normalise the training data in a narrower feature range of for example 0.1, 0.9. Doing so causes slightly bullish prices to be feature scaled below 1 and appropriately dealt with by the model.

4.4.2 Creating Arrays

To understand the importance our data's format, presentation and representation it is important to understand what is happening 'under the hood'. As libraries become more succinct and remove monotony, understanding can be pushed to the wayside. Still, understanding how data is inputted allow effective operating, extracting maximal value from the computationally expensive processing.

Shown in Figure 13 are the required arrays.

- The 1D array represents data for one day.
- The 2D array represents a set of days in time series.
- The 3D array represents successive days and their respective previous days.

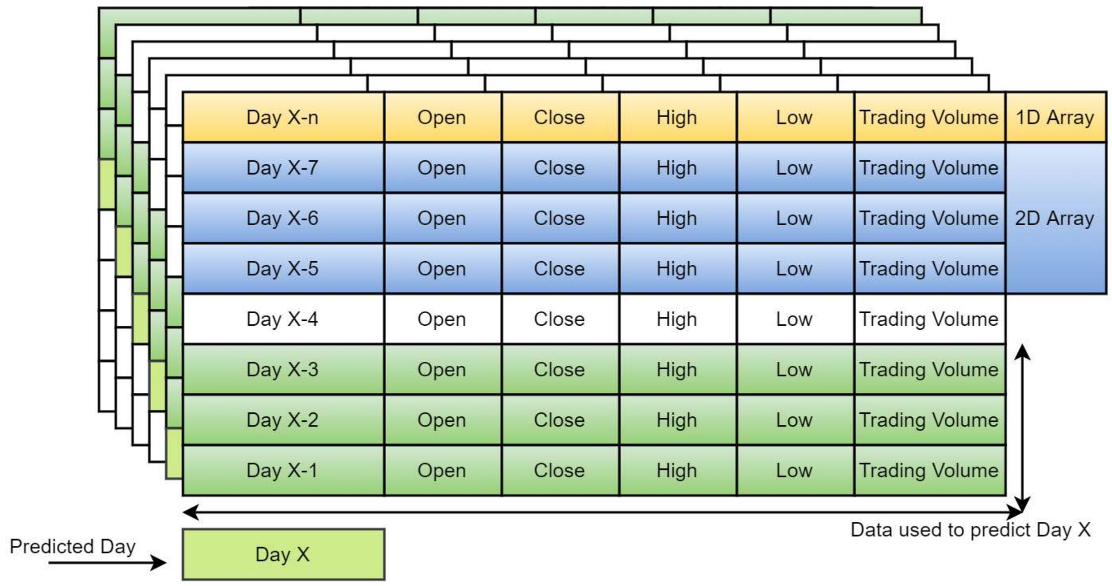


Figure 15 Array Input Data Visualisation

To reiterate, LSTM and GRU models take 3D inputs. In Figure 13 we show how this data is provided. Firstly, 1D arrays for each day of test data are created. Then 2D arrays that utilises X days of historic data as the second dimension are built. From here we are appending these 2D arrays (slices) into 3D ones where the new array dimension is for each day we wish to train on. This ensures learning occurs as each day is predicted for by a precession of previous days.

Our final shape follows the formula:

- (number of days(D),
- number of previous days used to inform one day(P),
- number of features(F))
- shape = (D, P, F)

During training the dark green arrays are passed into the first layer and corresponding cells to inform the next days (cell Day X) predicted value. By comparing true values to predicted ones the cell weights are optimised.

During Testing it is slightly different. Data for the prediction is inputted as a 3D single depth sheet, this is achieved by simple reshaping the 2D array into a 3D one, i.e. (P, F) → (1, P, F). For successive predictions the outcome of the model is used as the most recent 1D array. This is described in Figure 10 where the light green prediction arrays become dark green ones. This style of data creation can be computationally expensive because all features must be conserved. If only some features are optimised and predicted for then successive predictions cannot occur as the length of the newly created 1D array is incorrect.

4.4.3 Model

The model architecture we will use:

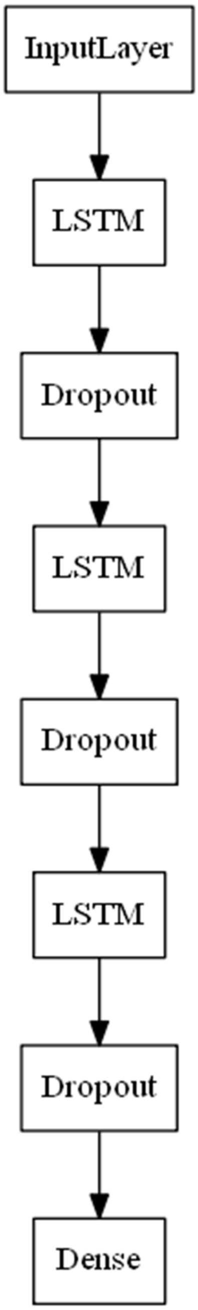


Figure 16 LSTM Deep Learning Model.

Here we are implementing 3 RNN layers, each proceeded by a dropout layer. Layer 1 increases the dimensionality of the output space. This allows more relationships between nodes can be understood by the model. Multiple layers are implemented to increase the number of parameters. The connections between layers allows further complex relationships to be discerned by the model. This model and its parameters determine the performance of prediction stock market movement.

4.5 Automation Process

After the creation of our neural network there is still considerable work required to understand the output we have created. The metrics (MSE&RMSE) from our data is valuable for comparing relative performance but it fails to provide a real world value for the true ability of the predictions. Converting this data into a more useful value was performed using this high level description.

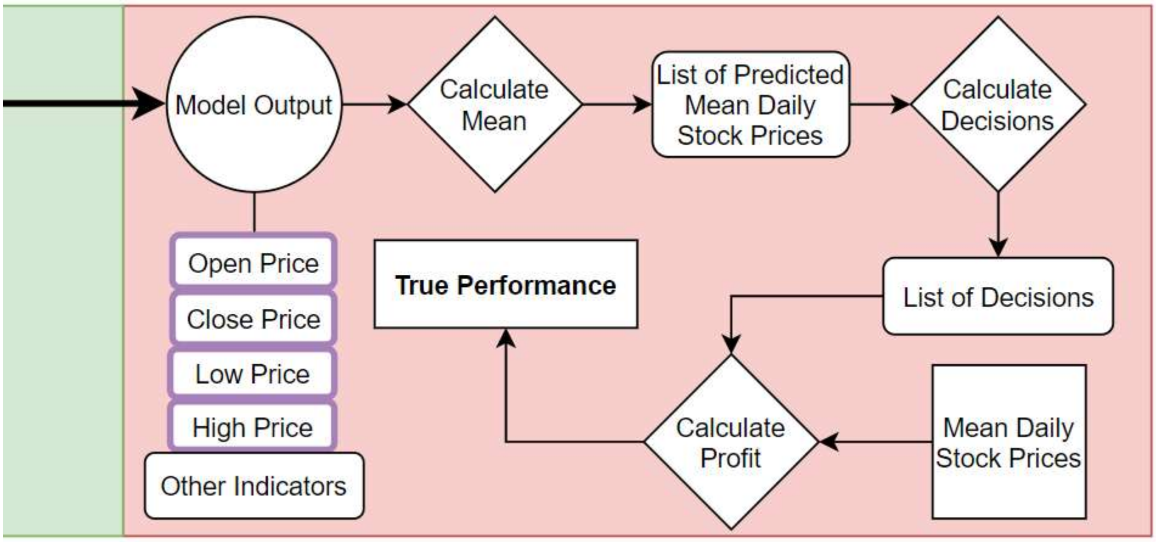


Figure 17 Automation Process's (Figure 1 Component 5) Internal Architecture. This is used to convert our model output into a real world "True Performance" value.

The common technical indicators (shown in purple) from the model prediction are used to condense our daily output into a single value (mean day value). We perform this for as many days as we would like to predict. This gives us a list of predicted daily stock prices. From this data we determine the optimum decisions for each day. To reiterate, the available trading decisions are:

- Buy – we convert our available money into shares.
- Hold – we conserve the shares and money we currently have
- Sell/Short – We sell any shares we own. Further from this we short the position. By this we are selling shares we do not have with the promise to buy an equal amount at a later time to offset the debt we have accrued.

We then use these decisions to determine how much money would we have earned by using the real mean daily stock prices.

Chapter 5 – Results & Evaluation

5.1 Introduction

Here we discuss the results in the same order as the original methodology numbers them. It should be stated that the model architecture testing was performed first. That is because the outcome of the model was the metric used to determine performance for each step. Within each of these sections we will present the results. Discussion regarding the quality and robustness of testing is presented within each subsection.

5 2 Technical Indicators

Our initial tests looked to determine which Technical Indicators were best at improving the performance of our models. Results are compared using standard academic performance metrics. These operate appropriately when comparing the relative performance of models. The different technical indicator data used is labelled: Common(A), Moving Average (MA), Strength Index (RSI) and Bollinger Bands (BB).

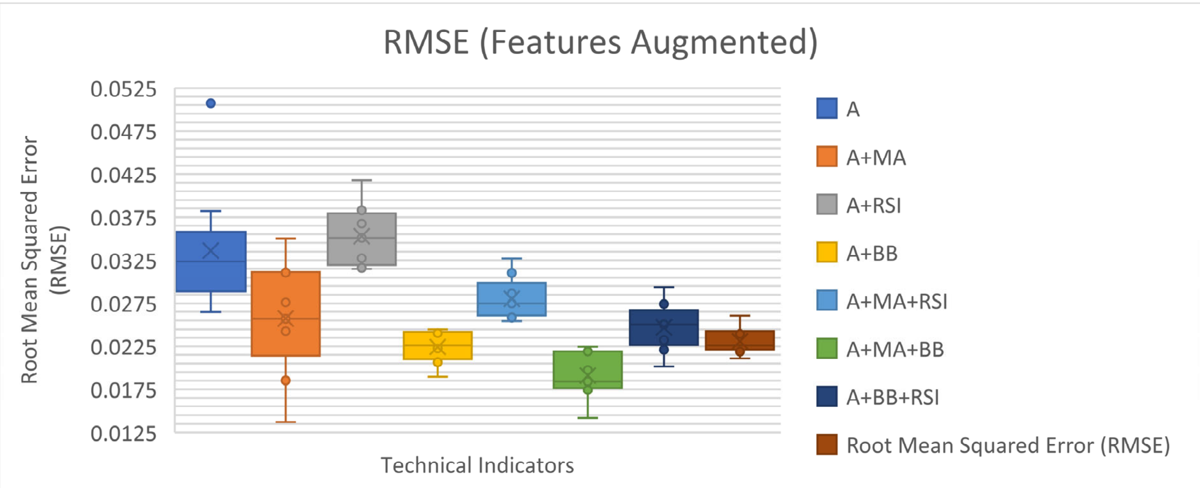


Figure 18 Testing Root Mean Squared Error for the Technical Indicators to determine which will perform best at price prediction. A lower value indicates a more accurate prediction.

In Figure 17 the performance and variance of the different combinations of the technical indicators are assessed with respect to root mean squared error. The data was further derived by dividing by the total number of features and multiplying by those used for the final prediction for the model (common indicators). Lower values indicate a better performance. Increasing the number of parameters reduced the deviation around the mean performance value but did not always improve the performance.

In Figure 18 we implemented a t-test to determine if the root mean squared error (RMSE) values were significantly different from the null hypothesis stating no difference or the result of experimental variance.

	A	A+MA	A+RSI	A+BB	A+MA+RSI	A+MA+BB	A+BB+RSI	A+BB+RSI+MA
A	N/A	0.446328852	0.0260069	0.165904437	0.232189706	0.001182383	0.74322111	0.6523423
A+MA	0.446328852	N/A	0.0017364	0.941106845	0.038465049	0.007414527	0.053391625	0.005251056
A+RSI	0.026006917	0.001736425	N/A	1.7471E-05	0.013784101	2.72982E-06	0.007144318	0.025734658
A+BB	0.165904437	0.941106845	1.747E-05	N/A	0.00081697	0.036265307	0.00081697	6.22459E-06
A+MA+RSI	0.232189706	0.038465049	0.0137841	0.00081697	N/A	7.37045E-06	0.856581515	0.446316904
A+MA+BB	0.733365333	0.052572504	3.586E-05	0.046426884	0.856581515	N/A	0.440580411	0.112067523
A+BB+RSI	0.74322111	0.053391625	0.0071443	0.00081697	0.856581515	0.006151831	N/A	0.370759387
A+BB+RSI+MA	0.6523423	0.005251056	0.0257347	6.22459E-06	0.446316904	0.006781727	0.370759387	N/A

0 ← P Value → 0.05+

Figure 19 A Heatmap showing the relative P values of different comparison Technical Indicator groups.

Figure 18 displays whether the mean values are significantly different from each other by colour and P value. The redder a value the more statistically significant it is. Using Figure 17&18 we can see that the Moving Average and Bollinger Bands composite give the best performance and have the most statistically significant results.

Derived technical indicators are important but their impact on MSE and RMSE is often counterintuitive. This is because the calculation of (R)MSE is achieved through the summation of more data. This process means that better common indicator prediction values can be clouded by the other predicted values. The proportion of the error dictated by these non-direct technical indicators is removed to counteract this phenomenon. This gives a better true representation of the utility of the models we have created in a real world financial setting. Here we can see the change in performance of A+MA+BB when compared to Figure 18.

5.3 Normalisation

Figure 19 the performance of our normalised data (company specific & market generalised) with respect to traditional data.

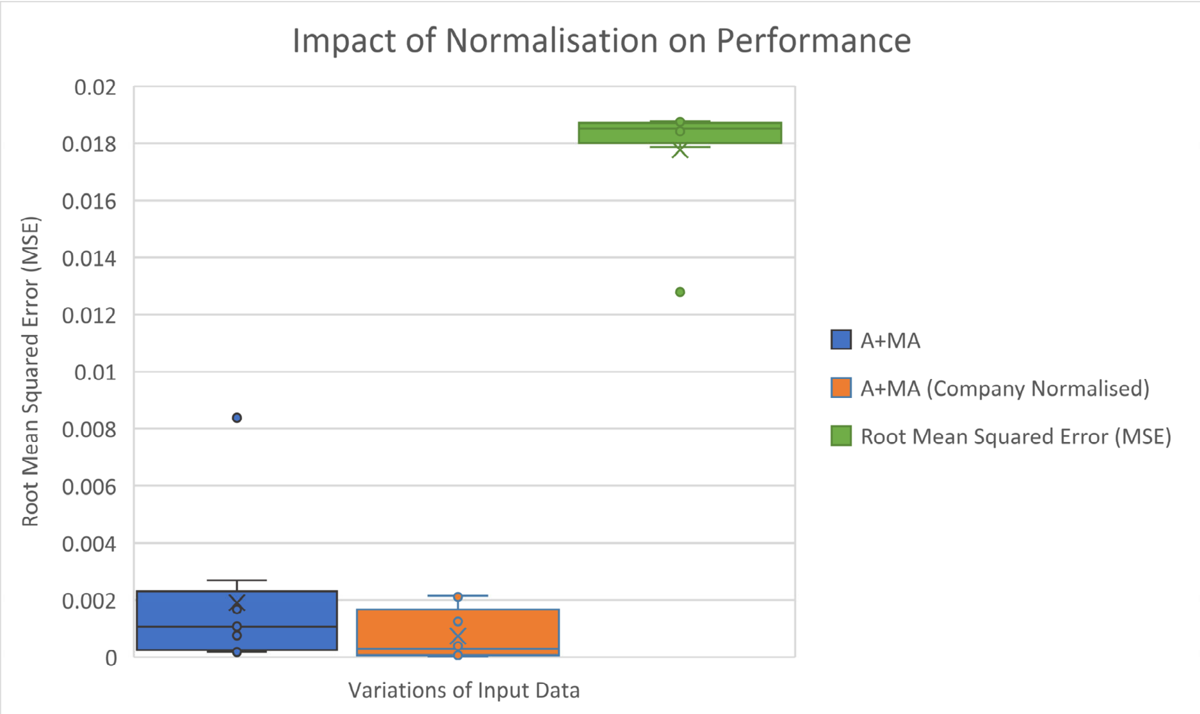


Figure 20 – Impact of normalisation on performance

Here we present the data regarding company and index normalisation. By attempting to remove market wide variables the data should theoretically be better to teach our models as the number of factors impacting changes in price should be removed.

The mean value for company normalised data is the best of all those tested. This was performed at my discretion as the companies that had to be picked needed to be similar. When the company data was normalised with respect to the NASDAQ index we had much worse results. This is probably due to the fact that the companies within the Index are high performing companies. All companies listed are fighting to increase in value and get onto the selective index. That means that generally weaker companies are replaced with better. So the factors removed are not just those of the market but of the best performers in the market. A better idea may have been to normalise with an entire random selection of companies. The non-normalised data was used to remove the selection process that would have created non-random picks.

5.4 Model Parameters & Composition

5.4.1 Model Comparison (LSTM & GRU)

Here we have presented the difference in performance of our GRU and LSTM models. All experiments were performed with the same novel parameters values to ensure that no specific optimisation learnt in previous experiments were accidentally included and influencing the performance. The performance is shown in Figure 20

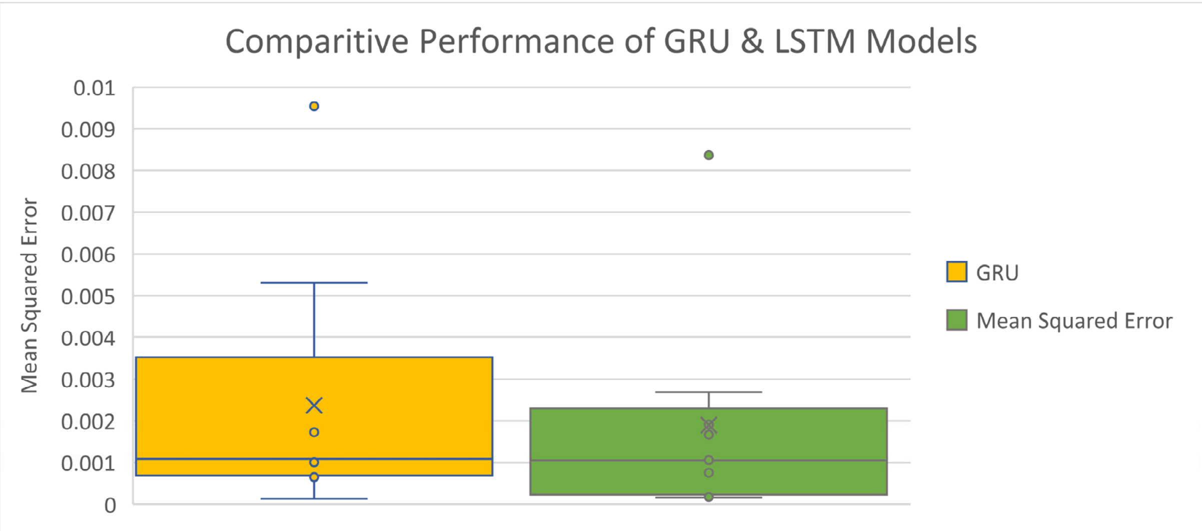


Figure 21 Comparison of GRU and LSTM model performance.

Here in Figure 20 we can see the same performance is as proposed for other time series tasks: both perform almost exactly the same, with LSTM performing marginally better as seen in some other sequential tasks (Jozefowicz et al., 2015). The distribution of LSTM performance is also tighter in its distribution.

5.4.2 Layers 1&2

It was covered in literature that in a Neural Network with a depth of 3 layers it is best to make both the 1st and 2nd layers the same size (Ahmed, 2020). The different combinations of connections and their weights are what allow this first part of learning to occur. This is named feature extraction. Only the second LSTM layers dimensionality was altered and then used for layers 1 & 2.

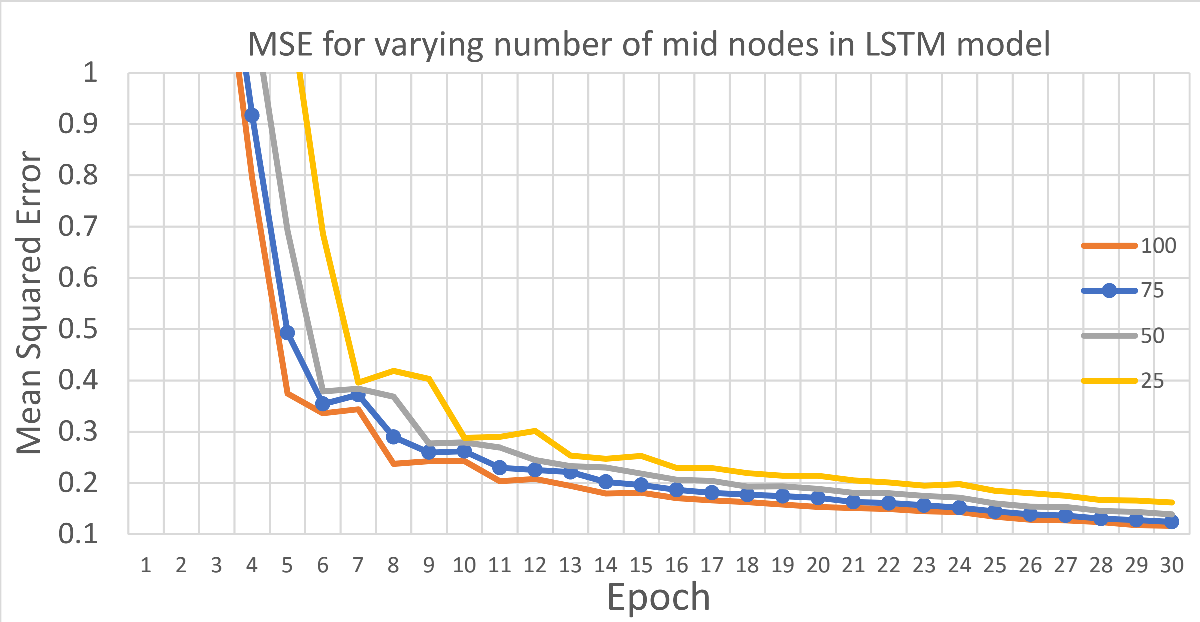


Figure 22 MSE for varying dimensionality of 2nd LSTM layer over a range of 30 epochs (iterations over training set).

In Figure 21 we observed a larger number of nodes improves performance over the entire epoch range. This is the expected performance found from reading literature (Ahmed et al., 2018). This data also shows that larger node numbers are quicker to optimise than their smaller node counterparts.

To further examine the performance, we are now looking at it relative to the node count. By examining performance/node it helps us to see the utility of the different setups with respect to computational output.

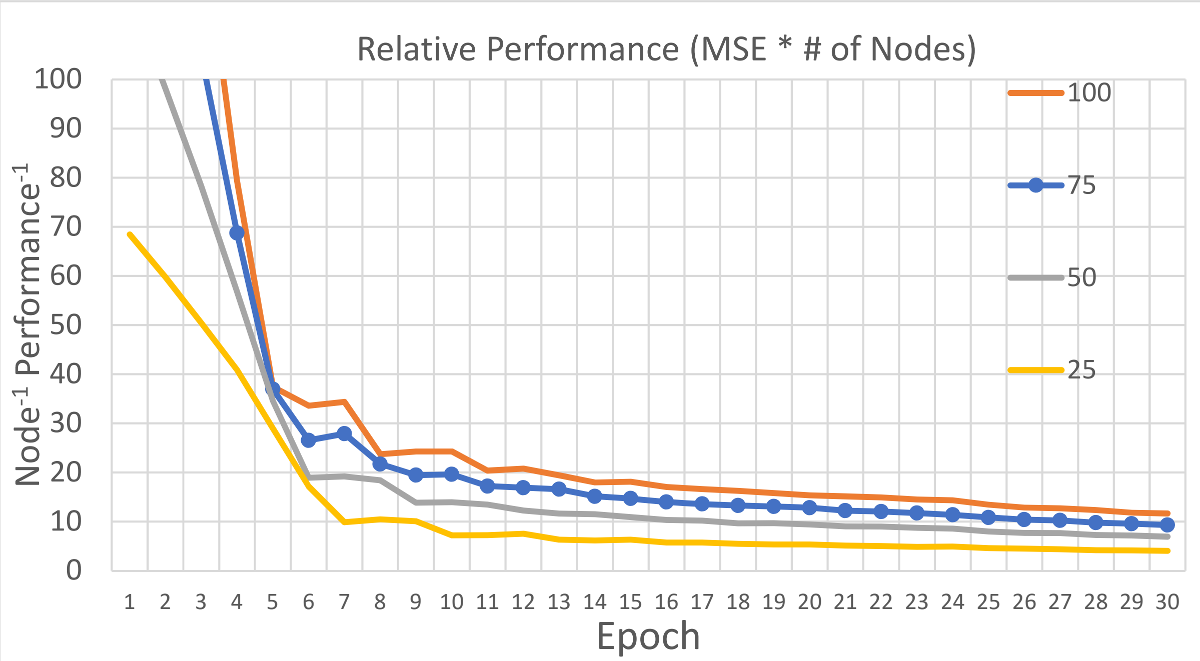


Figure 23 Performance of the 2nd LSTM layer per node. A lower value indicates better performance per node in the layer.

In Figure 22 as the number of nodes increases the performance output of per node declines. Here we settled on using the 50 dimensionality layer as a good performer whilst removing less impactful, wasteful computation as seen in the higher node variations. This wasteful computation retained 72, 77 & 90% of the performance at epoch 30 for 100, 75 & 50 dimensionality respectively when compared to the 25 node example. If time is an issue it seems that a reduction in the number of layers still trains the model. Surprisingly the difference in the relative performances reduces over the epochs, this means that the higher node valued layers are achieving a faster learning rate as epochs are iterated through.

5.4.3 Layer 3

We tested the dimensionality of the last LSTM layer to see how altering this improved the performance of our model.

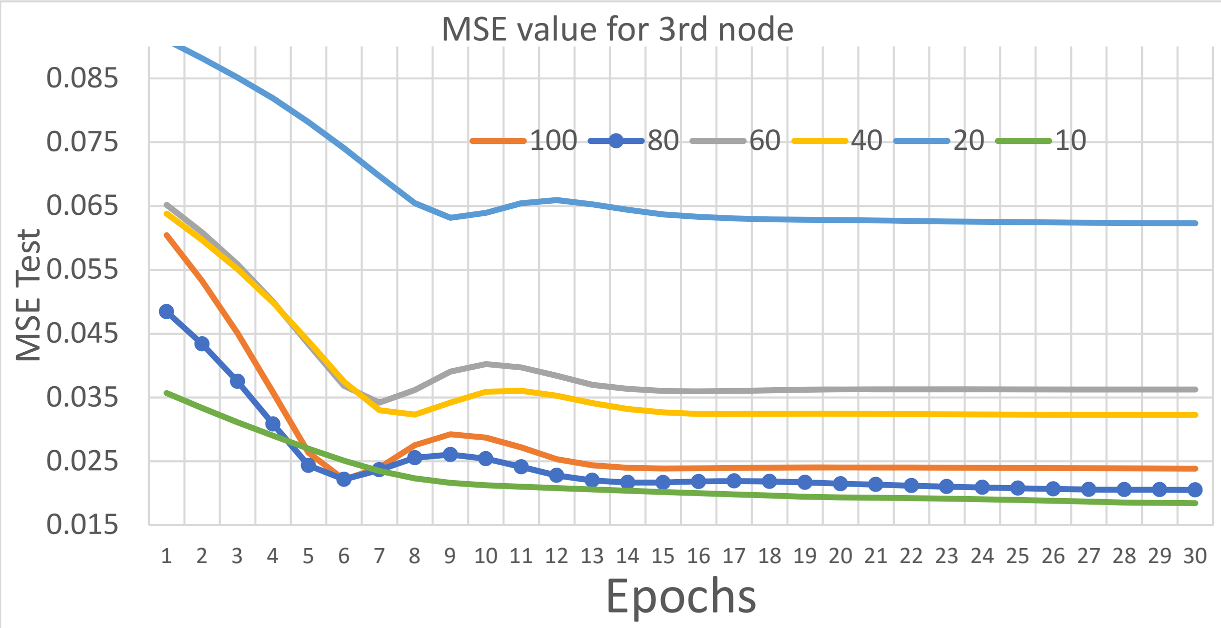


Figure 24 Performance relative to changes in the dimensionality of the 3rd LSTM node. From this graph we can observe that the performance of a 20 dimension node is poor with respect to others.

The best performance in ranked order: 10, 80, 40, 60, 20. The worst performing was a dimensionality of 20. We also found that the best performer was the 10-dimension model. We opted to move forward with a value of 10 because of the good performance and the good generalisation it exhibited. This seems a strange result as all other nodes follow the principle observed in the previous node dimensionality selection. The condensation of data into a smaller number of nodes means that the final dense node that produces the output is performing less of the critical condensing of information. By splitting this task over two layers more complex computation is occurring. This seems to produce a better result for our 10-node example. It is not an issue of smaller nodes providing less values resulting in a smaller sum when calculating the MSE. The prediction output size was maintained.

5.4.4 Dropout

Here we tested dropout and its impact on performance. We altered the dropout values simultaneously.

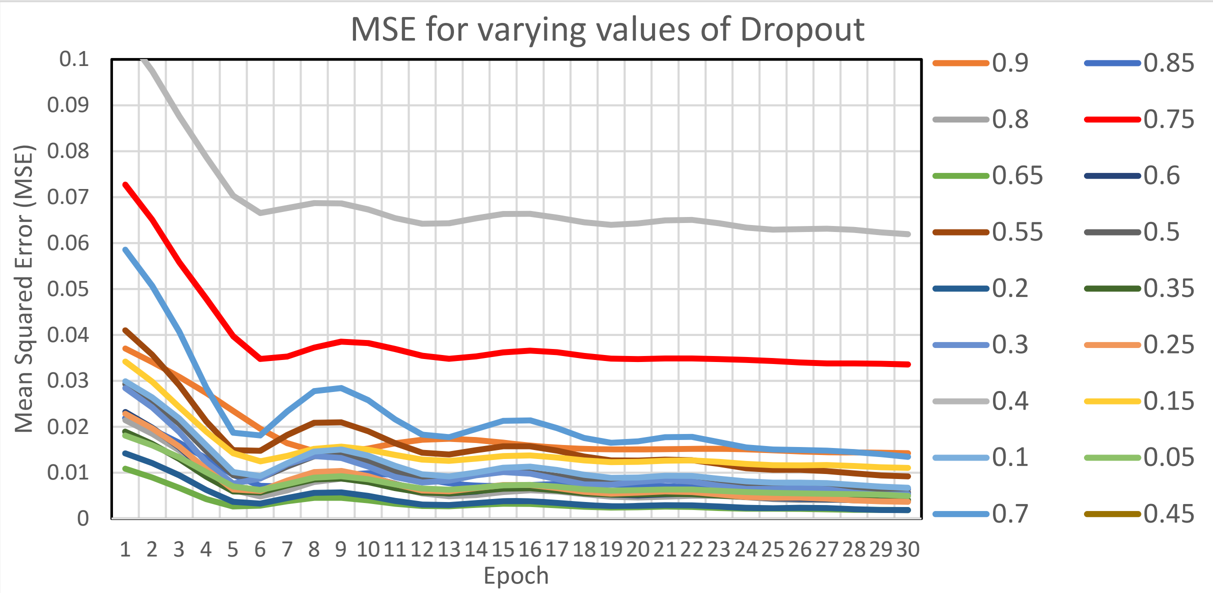


Figure 25 Showing how varying the dropout rate impacts the performance of the model.

As shown, high values for dropout perform worse. The best performing dropout rate as the number of epochs increase are 0.65 and 0.2. It was decided to implement 0.2 as conserving more data was seen as more appropriate.

High value dropouts (0.7 – 0.9) suffered from either poor MSE values or poor convergence. High dropout rates are likely to reduce the noisy biases because the dependency on performance of the model falls to a small number of nodes. This will reduce the ability for the search space to be search to be reduced. This prevents a good minima value from being found as particular axes of the search space cannot be explored.

The model was not overfitting on the data it viewed. The likely cause of this being that the predictions previous days (2D array – Section 4.4.2) overlapped with the testing data's previous days. This issue was unavoidable as artificially separating the data by a considerable time difference would ruin the performance and point of training by making the requirements of the model unnecessarily high.

5.4.5 Batch Size

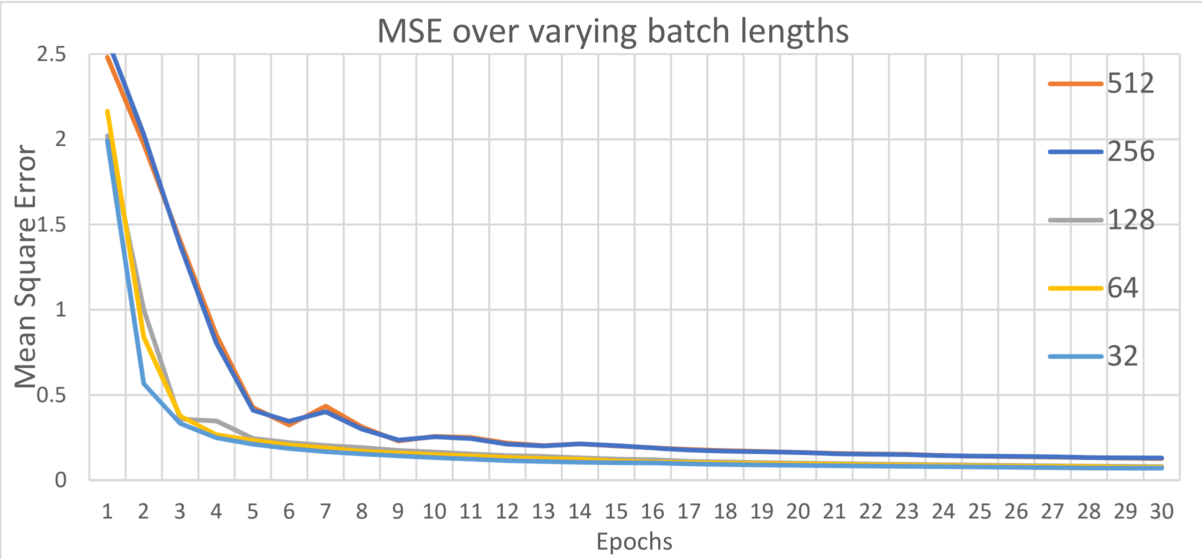


Figure 26 Showing how batch lengths alter the performance and speed of convergence over 30

We found that a lower batch size actually performed better over our data set. This is probably caused because the subset of data is noisier. This noise may help to avoid local minima whilst still converging to a good value quickly. Large batch sizes reduced the stochasticity of the gradient descent implemented by the optimiser.

5.4.5 Chosen Model Configuration

The Data Points and Prediction Steps values were chosen due to training time costraints. Ideally these values would be larger.

Data Points(Days): 2060

Prediction Steps (Days): 60

Training Split: 0.8

Batch Size: 32

Parameters Used: Common Indicators + Moving Average + Bollinger Bands

Model: LSTM

Dropout Rate: 0.2

Layers 1&2: 50 Nodes

Layer 3: 10 Nodes

We will use these parameters for our True Prediction testing in the last part of our experimental results.

5.5 True Performance valuation

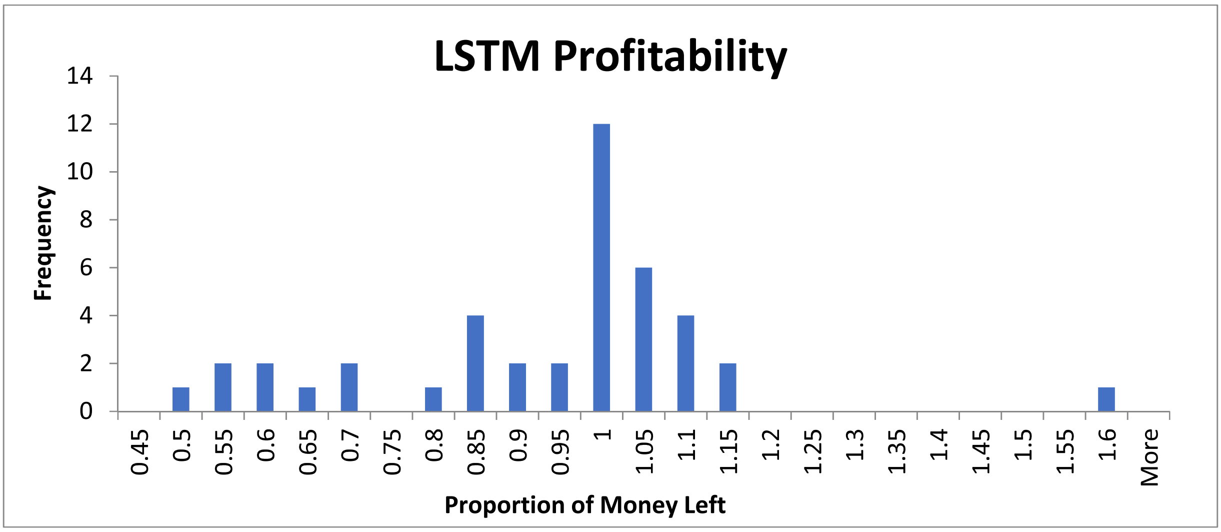


Figure 27 Prediction Algorithm Profitability Histogram

Figure 27. The profitability of our model was not successful in outcompeting a hold strategy or a non-involvement strategy. The results were not normally distributed. The proportion of loses is far greater than those with significant gains. The results performed worse than random. This experimental procedure has provided potential reinforcement for the efficient market hypothesis. Prediction may ultimately lead to poorer results.

The mean profit value is 92% of the original value. With a median return of 98% of the original value. Both are losses over many iterations. The average value compared to the original is statistically significantly less profitable. The experimentation failed to create a profitable algorithm using technical indicators. The decisions made were also tested by inverting the choices, this also did not prove profitable. This is owing to the non-binary choices available (Buy, Hold, Short).

Chapter 6 Discussion

6.1 Introduction to Discussion

6.2 Results

6.2.1 Technical Indicators

Bollinger bands and Moving Average were the best candidates for improving performance. When creating our augmented features graph (Figure 17) we made an assumption that the distribution of the error values from our models' different outputs were equal in size. The closed nature of the Keras API used made it impossible for that data to be extracted from the software. If possible, the exact proportion of error caused by the common indicators that were used to define our mean prediction price could be calculated. It was difficult to compare this aspect of our methodology with others. The wide spread belief in academia that more data equals better performance means other academics gloss over this component of the model. It is not the most interesting or engaging but data shows that further work should consider the negative impact superfluous data can have on performance.

6.2.2 Model Architecture

6.2.2.1 Model

Our models optimisation followed that found in the literature (Greff et al., 2017), (Jozefowicz et al., 2015). The model work appropriately. The implementation of LSTM and GRU models in research is rarely discussed in detail within the majority of papers discussing these programs. Different initialisation bias's and library controlled processes were not discussed in literature to simplify and condense lengthy reports (Greff et al., 2017), (Nelson et al., 2017), (Naik & Mohan, 2019). This made it impossible to definitively know how testing had been implemented. The optimal performance of these algorithms may not have been uncovered given the time it would take to search the search space to find the best combination of all available parameters.

Data within our study was split into 2 categories. Training and Testing/Validation. Other work around this topic had work separated similarly or into 3 groupings: Training, Testing and Validation. This made comparisons difficult.

Our model also operates using successive predictions. This process means that the number of features must be conserved in the model output for the successive input data to be created. This means that all values must be optimised for. If particular ones are given greater optimisation importance (weighting) the quality of the successive outputs would be worse.

6.2.2.2 Nodes

Better performance was found when higher numbers of nodes were implemented. If computationally feasible, a greater number of nodes is more appropriate for an optimal performance value. Other data has found large node values can cause serious, rapid overfitting. This can be an issue for prediction of longer lengths (Ahmed et al., 2018).

6.2.2.3 Dropout:

We found for dropout that the best performer was a value of 0.2 dropout. Theory dictates some sweet spot value for dropout based on the sample size and the dimensionality of the layers (Gal & Ghahramani, 2016). Small values of dropout had comparatively negative results than higher dropout values. This is probably due to the reliance on particular nodes that are thoroughly trained and then removed for later epochs by random. This could have drastically reduced performance for particular experimental runs. Other academic research found the removal of any nodes caused a diminished performance. This was not present in our data as the scope of the dimensionality was not large enough. This was due to the computational time taken to run successive experiments.

6.2.2.4 Batch Size

For batch size the academic literature was of a higher calibre than my research. It uses more complex combinations of layers and higher volumes of technical data. This is due to the availability of different, better hardware that can process more streams of data in parallel. It makes repeating this kind of experimentation difficult for an average user. This issue throws up worrying concerns about the validity of the experimental procedure if it is provide but cannot be run by others. This issue is very prevalent in the commercial field where large tech companies, specifically google, make outlandish claims that cannot be verified due to the huge amount of computational power required to repeat experiments. This reduced the pool of available information as only impartial research founded on publicly available information was used.

6.3 Literature

Continuing on, other issues surrounding literature involved the use of newspaper articles (like the economist). These kinds of companies are not neutral in their stance on financial issues. Many reports are simplified for the common reader, the mathematics can be overwhelming so it is important to make these areas of research approachable. However, the simplification can often hide how experiments are performed, code is rarely appended so ensuring that the results are valid is difficult. Promising results, no matter how small, are always interesting. These tap into our brains reward centres and enthuse us into reading and improve the light in which the author is portrayed. This leaves the field of data analytics open to some air of obscurity and non-scrutiny.

The financial aspect of the insights found also play into this. If an algorithm really worked the decision to publish it and saturate the number of people that know about it thus diminishing your edge in the market seems foolish. Those insights would be kept quiet and the financial advantage enjoyed.

The sparse nature of available information specifically surrounding stock market predictions and algorithm development for this field make literature reading difficult. It also makes the possibility of good models a hopeful possibility as they are likely unreported. Work outside of this immediate field was performed to draw relative comparisons about RNN models and their theoretical capabilities. This was then used to create assumptions about the transferring of theory to the field of stock predictions. The wide variety of models and the consistent changes to software used can make the process tedious. Beyond these issues comes a benefit; primary research becomes a necessity and this ensures that the reproducibility of your results is high. If interesting insights are found, models can be seeded to reproduce results and models can also be saved to ensure that the pseudo-random process of training can be conserved.

6.4 Automation Process

The automation process was something I could find little/no previous work on. Perhaps because the impact of building a continuous value predictor over a binary classifier was deemed too insignificant to others. The performance values attained allowed us to translate our performance beyond academic metrics into true real world applications. This process could be beneficial in the area of more advanced trading where predictions about changes in price can be propositioned within the market and be highly profitable.

There are issues surrounding this automation component that would be great to tackle in further work. These include but are not limited to, how the spread price (difference between buy and sell price at a point in time) impacts the profitability of the prediction. How variance in performance can be included in the models perceived risk level.. The inclusion of fundamental analysis and subsequent grading of companied to inform the proportion of money you wish to risk when investing in a particular stock. It is important to realise that these caveats may help to improve the performance but they are achieving through conventional economics and not any form of deep learning.

Chapter 7 Conclusion

7.1 Summary

This thesis started with the goal of creating a high performance model for stock market predictions. LSTM and GRU models were implemented due to their aptitude at handling time series data. We found that the 'best' performing model was an LSTM model (parameter information in section 5.4.5). Technical Indicators and hyperparameters of the LSTM model were tested for their impact on the performance of the prediction software.

We found which performed best and which were of higher variance. Over long periods of trading this variance will be negligible in the outcome of results but it is important to note its impact for smaller traders who are charged a larger proportion of their money to diversify and buy and sell shares at a high rate due to the spread that exists in markets.

We produced a software component to accurately convert our predictions into true valuations. These valuations are what we define as 'best'.. We were able to automate the entire process except for the movement of decisions onto a trading platform. That would still need to be performed manually.

7.2 Hindsight & Future Work

7.2.1 Hindsight

Time constraints and the difficulty in programming the automation process for the thesis reduced the scope of experimentation that was previously hoped for in the Initial Proposal. The process of real world application would not be limited by computation power. The rewards of success would mitigate any upfront costs. Building less but better performing models would have been a more impactful an given a better outcome to the thesis. The inclusion of intra-day prices may have been a better path to go down as it has seen some success for traders (Urquhart et al., 2015).

After the experiments were performed it became obvious that the performance metric chosen were not indicative of a real world value predictor. This issue was avoided by other academics but their reasoning for binary up or down movement prediction wasn't explained. If the thesis was repeated again using the output of the automation process instead of the Adam optimiser to train the model would be much better. By linking the output to the training mechanism there would be not separation between the model optimisation parameters and real-world performance.

7.2.2 Normalisation

It would have been interesting to normalise company data with respect to a random assortment of companies rather than the high performing index used. This process would reduce the variance seen in the company specific normalisation as well. This component could be furthered by using the training data to approximate the linear growth of the results. Hence, we can implement a new divisor for normalising the training data or an augmented max feature range value.

7.2.3 Dropout

Implementation of a new style of dropout called DropConnect could have improved the learning process. This style drops the weights of nodes rather than severing the connection and has proved effective in visual tasks implementing highly connected RNNs (Wan et al., 2013). This would cause the loss of particular time sensitive information. By removing this the search space would be easier to search and better optima potentially found.Examining a wider scope of parameters and different model subtypes would likely lead to more insights about how best to build a more accurate system. Having improved hardware capabilities and a more solid knowledge of python and its memory allocation processes would likely result in faster and more successful experimental runs.

7.2.4 Sentiment Analysis

Beyond improvements to the current architecture and parameters discussed it would be interesting to include a level of sentiment analysis regarding companies and their public image on the internet. It has shown effective on microblogs like twitter (Li et al., 2018). Text is one of the most important ways of information presentation in existence. It allows data and knowledge regarding subjects to be shared concisely and accurately through mediums like newspapers, books and online resources. It follows general rules and structure. Written English has pretty standardised rules so predicting it is easy considering the large volumes of proper English text available for use in training.

Since the inception of this thesis the car company Tesla has been on a blazing streak of increased valuation. This is in part due to the presence of their CEO whom is very active with the community of people interested in his company. Drawbacks for sentiment analysis are wide in scope due to the wild west nature of the internet. People can artificially inflate data so to proceed forward with this work would require a great deal of caution and conscientiousness. Following up by the inclusion of companies public image in the models inputs would make for an exciting successor to the thesis.

References

Ahmed, W., Bahador, M., Kumar, A., & Ekeberg, Ö. (2018). *The accuracy of the LSTM model for predicting the S&P 500 index and the difference between prediction and backtesting*. 35.

Ahmed, Y. (2020, January 12). *Getting rich quick with machine learning and stock market predictions*. Medium. https://towardsdatascience.com/getting-rich-quick-with-machine-learning-and-stock-market-predictions-696802da94fe

Banko, M., & Brill, E. (2001). Scaling to Very Very Large Corpora for Natural Language Disambiguation. *Proceedings of the 39th Annual Meeting of the Association for Computational Linguistics*, 26–33. https://doi.org/10.3115/1073012.1073017

Bengio, Y. (2012). Practical recommendations for gradient-based training of deep architectures. *ArXiv:1206.5533 [Cs]*. http://arxiv.org/abs/1206.5533

Bianchi, F. M., Maiorino, E., Kampffmeyer, M. C., Rizzi, A., & Jenssen, R. (2017). Recurrent Neural Network Architectures. In F. M. Bianchi, E. Maiorino, M. C. Kampffmeyer, A. Rizzi, & R. Jenssen (Eds.), *Recurrent Neural Networks for Short-Term Load Forecasting: An Overview and Comparative Analysis* (pp. 23–29). Springer International Publishing. https://doi.org/10.1007/978-3-319-70338-1_3

Borges, M. R. (2010). Efficient market hypothesis in European stock markets. *The European Journal of Finance*, *16*(7), 711–726. https://doi.org/10.1080/1351847X.2010.495477

Burton, N. (2018). *An Analysis of Burton G. Malkiel's A Random Walk Down Wall Street* (1 edition). Macat Library.

Campbell, M., Hoane, A. J., & Hsu, F. (2002). Deep Blue. *Artificial Intelligence*, *134*(1), 57–83. https://doi.org/10.1016/S0004-3702(01)00129-1

Chung, J., Gulcehre, C., Cho, K., & Bengio, Y. (2014). Empirical Evaluation of Gated Recurrent Neural Networks on Sequence Modeling. *ArXiv:1412.3555 [Cs]*. http://arxiv.org/abs/1412.3555

Da, Z., Engelberg, J., & Gao, P. (2011). In Search of Attention. *The Journal of Finance*, *66*(5), 1461–1499. https://doi.org/10.1111/j.1540-6261.2011.01679.x

Dickinson, Z. C. (2013). Economic Motives: A Study in the Psychological Foundations of Economic Theory, with Some Reference to Other Social Sciences. In *Economic Motives*. Harvard University Press. https://hup.degruyter.com/view/title/323377

The rise of the financial machines. (n.d.). *The Economist*. Retrieved 24 August 2020, from https://www.economist.com/leaders/2019/10/03/the-rise-of-the-financial-machines

Ellingrud, K. (2018). *The Upside Of Automation: New Jobs, Increased Productivity And Changing Roles For Workers* [Financial News Article]. Forbes. https://www.forbes.com/sites/kweilinellingrud/2018/10/23/the-upside-of-automation-new-jobs-increased-productivity-and-changing-roles-for-workers/

Fama, E. F. (1970). Efficient Capital Markets: A Review of Theory and Empirical Work. *The Journal of Finance*, *25*(2), 383–417. JSTOR. https://doi.org/10.2307/2325486

Ferguson, C. E. (2008). The Neoclassical Theory of Production and Distribution. In *Cambridge Books*. Cambridge University Press. https://ideas.repec.org/b/cup/cbooks/9780521076296.html

Fernandez, P., Aguirreamalloa, J., & Liechtenstein, H. (2008). *The Equity Premium Puzzle: High Required Premium, Undervaluation and Self Fulfilling Prophecy* (SSRN Scholarly Paper ID 1274816). Social Science Research Network. https://doi.org/10.2139/ssrn.1274816

Frydman, C., & Camerer, C. F. (2016a). The Psychology and Neuroscience of Financial Decision Making. *Trends in Cognitive Sciences*, *20*(9), 661–675. https://doi.org/10.1016/j.tics.2016.07.003

Frydman, C., & Camerer, C. F. (2016b). The Psychology and Neuroscience of Financial Decision Making. *Trends in Cognitive Sciences*, *20*(9), 661–675. https://doi.org/10.1016/j.tics.2016.07.003

Fukushima, K. (1980). Neocognitron: A self-organizing neural network model for a mechanism of pattern recognition unaffected by shift in position. *Biological Cybernetics*, *36*(4), 193–202. https://doi.org/10.1007/BF00344251

Gal, Y., & Ghahramani, Z. (2016). A Theoretically Grounded Application of Dropout in Recurrent Neural Networks. In D. D. Lee, M. Sugiyama, U. V. Luxburg, I. Guyon, & R. Garnett (Eds.), *Advances in Neural Information Processing Systems 29* (pp. 1019–1027). Curran Associates, Inc. http://papers.nips.cc/paper/6241-a-theoretically-grounded-application-of-dropout-in-recurrent-neural-networks.pdf

Girish, G., & Pinjare, S. L. (2016). *Audiogram equalizer using fast fourier transform*. 1877–1881. https://doi.org/10.1109/SCOPES.2016.7955772

Greff, K., Srivastava, R. K., Koutník, J., Steunebrink, B. R., & Schmidhuber, J. (2017). LSTM: A Search Space Odyssey. *IEEE Transactions on Neural Networks and Learning Systems*, *28*(10), 2222–2232. https://doi.org/10.1109/TNNLS.2016.2582924

Gupta, S., & Dinesh, D. A. (2017). Resource usage prediction of cloud workloads using deep bidirectional long short term memory networks. *2017 IEEE International Conference on Advanced Networks and Telecommunications Systems (ANTS)*, 1–6. https://doi.org/10.1109/ANTS.2017.8384098

Howell, David C. 2010. Statistical methods for psychology, Chapter 5. 7th ed. Boston: Wadsworth.

Hussainey, K., Mgbame, C. O., & Chijoke-Mgbame, A. M. (2011). Dividend policy and share price volatility: UK evidence. *Journal of Risk Finance*, *12*(1), 57–68. http://dx.doi.org/10.1108/15265941111100076

Jansen, S. (2018). *Hands-On Machine Learning for Algorithmic Trading: Design and implement investment strategies based on smart algorithms that learn from data using Python*. Packt Publishing Ltd.

Joseph Johnson, Gerard J. Tellis, Deborah J. Macinnis, Losers, Winners, and Biased Trades, Journal of Consumer Research, Volume 32, Issue 2, September 2005, Pages 324–329, https://doi.org/10.1086/432241

Jozefowicz, R., Zaremba, W., & Sutskever, I. (2015). *An Empirical Exploration of Recurrent Network Architectures*. 9.

Kang, A., and Duk Seo, J. (2019). Understanding Backpropagation as Applied to LSTM.pdf. Google Docs. Retrieved 6 August 2020, from https://drive.google.com/file/d/1RTLuTLWzyGUNvXstOdYlhdN-HbnGBntg/view.

Kim, T., & Kim, H. Y. (2019). Forecasting stock prices with a feature fusion LSTM-CNN model using different representations of the same data. *PLOS ONE*, *14*(2), e0212320. https://doi.org/10.1371/journal.pone.0212320

Kingma, D. P., & Ba, J. (2017a). Adam: A Method for Stochastic Optimization. *ArXiv:1412.6980 [Cs]*. http://arxiv.org/abs/1412.6980

Kingma, D. P., & Ba, J. (2017b). Adam: A Method for Stochastic Optimization. *ArXiv:1412.6980 [Cs]*. http://arxiv.org/abs/1412.6980

Kirilenko, Andrei and Kyle, Albert and Samadi, Mehrdad and Tuzun, Tugkan. (2011). The Flash Crash: The Impact of High Frequency Trading on an Electronic Market. SSRN Electronic Journal. 10.2139/ssrn.1686004.

Kovalev, D., Mishchenko, K., & Richtárik, P. (2019). Stochastic Newton and Cubic Newton Methods with Simple Local Linear-Quadratic Rates. *ArXiv:1912.01597 [Cs, Math, Stat]*. http://arxiv.org/abs/1912.01597

Li, T., van Dalen, J., & van Rees, P. J. (2018). More than just Noise? Examining the Information Content of Stock Microblogs on Financial Markets. *Journal of Information Technology*, *33*(1), 50–69. https://doi.org/10.1057/s41265-016-0034-2

Liu, C. Y., & He, J. (1991). A Variance-Ratio Test of Random Walks in Foreign Exchange Rates. *The Journal of Finance*, *46*(2), 773–785. JSTOR. https://doi.org/10.2307/2328848

Liu, W., Wang, Z., Liu, X., Zeng, N., Liu, Y., & Alsaadi, F. E. (2017). A survey of deep neural network architectures and their applications. *Neurocomputing*, *234*, 11–26. https://doi.org/10.1016/j.neucom.2016.12.038

Lo, A. W., Mamaysky, H., & Wang, J. (2000). *Foundations of Technical Analysis: Computational Algorithms, Statistical Inference, and Empirical Implementation* (Working Paper No. 7613; Working Paper Series). National Bureau of Economic Research. https://doi.org/10.3386/w7613

Lui, Y.-H., & Mole, D. (1998). The use of fundamental and technical analyses by foreign exchange dealers: Hong Kong evidence. *Journal of International Money and Finance*, *17*(3), 535–545. https://doi.org/10.1016/S0261-5606(98)00011-4

Luo, L., Xiong, Y., Liu, Y., & Sun, X. (2019). Adaptive Gradient Methods with Dynamic Bound of Learning Rate. *ArXiv:1902.09843 [Cs, Stat]*. http://arxiv.org/abs/1902.09843

M, H., E.a., G., Menon, V. K., & K.p., S. (2018). NSE Stock Market Prediction Using Deep-Learning Models. *Procedia Computer Science*, *132*, 1351–1362. https://doi.org/10.1016/j.procs.2018.05.050

Macedo, L. L., Godinho, P., & Alves, M. J. (2017). Mean-semivariance portfolio optimization with multiobjective evolutionary algorithms and technical analysis rules. *Expert Systems with Applications*, *79*, 33–43. https://doi.org/10.1016/j.eswa.2017.02.033

Malkiel, B. G. (n.d.). *The Efficient Market Hypothesis and its Critics*. 47.

Mcleod, S. (n.d.). *Maslow's Hierarchy of Needs*. 16.

Nadkarni, P. M., Ohno-Machado, L., & Chapman, W. W. (2011). Natural language processing: An introduction. *Journal of the American Medical Informatics Association*, *18*(5), 544–551. https://doi.org/10.1136/amiajnl-2011-000464

Naik, N., & Mohan, B. R. (2019). Optimal Feature Selection of Technical Indicator and Stock Prediction Using Machine Learning Technique. In A. K. Somani, S. Ramakrishna, A. Chaudhary, C. Choudhary, & B. Agarwal (Eds.), *Emerging Technologies in Computer Engineering: Microservices in Big Data Analytics* (pp. 261–268). Springer. https://doi.org/10.1007/978-981-13-8300-7_22

Naseer, M., & Tariq, Dr. Y. (2015). The Efficient Market Hypothesis: A Critical Review of the Literature. *The IUP Journal of Financial Risk Management*.

Nelson, D. M. Q., Pereira, A. C. M., & de Oliveira, R. A. (2017). Stock market's price movement prediction with LSTM neural networks. *2017 International Joint Conference on Neural Networks (IJCNN)*, 1419–1426. https://doi.org/10.1109/IJCNN.2017.7966019

Nti, I. K., Adekoya, A. F., & Weyori, B. A. (2020). A systematic review of fundamental and technical analysis of stock market predictions. *Artificial Intelligence Review*, *53*(4), 3007–3057. https://doi.org/10.1007/s10462-019-09754-z

NW, 1615 L. St, Washington, S. 800, & Inquiries, D. 20036 U.-419-4300 | M.-419-4349 | F.-419-4372 | M. (2019, July 23). Trends and Facts on Online News | State of the News Media. *Pew Research Center's Journalism Project*. https://www.journalism.org/fact-sheet/digital-news/

Omer, H., & Alon, N. (1994). The continuity principle: A unified approach to disaster and trauma. *American Journal of Community Psychology*, *22*(2), 273–287. https://doi.org/10.1007/BF02506866

Park, C.-H., & Irwin, S. H. (2004). The Profitability of Technical Analysis: A Review. *SSRN Electronic Journal*. https://doi.org/10.2139/ssrn.603481

Patel, J., Shah, S., Thakkar, P., & Kotecha, K. (2015). Predicting stock and stock price index movement using Trend Deterministic Data Preparation and machine learning techniques. *Expert Systems with Applications*, *42*(1), 259–268. https://doi.org/10.1016/j.eswa.2014.07.040

Pennings, J. M. E. (2002). Stock-market Psychology: How People Value and Trade Stocks: Karl-Erik Wärneryd, Edward Elgar Publishing Limited, Cheltenham, UK and Northampton, Massachusetts, USA, pp. 352, 2001, ISBN 1-84064-736-1, 69.95 Pounds Sterling, $100.00. *Journal of Economic Psychology*, *23*(4), 546–548. https://doi.org/10.1016/S0167-4870(02)00103-4

Phillips, P. J., Yates, A. N., Hu, Y., Hahn, C. A., Noyes, E., Jackson, K., Cavazos, J. G., Jeckeln, G., Ranjan, R., Sankaranarayanan, S., Chen, J.-C., Castillo, C. D., Chellappa, R., White, D., & O'Toole, A. J. (2018). Face recognition accuracy of forensic examiners, superrecognizers, and face recognition algorithms. *Proceedings of the National Academy of Sciences*, *115*(24), 6171–6176. https://doi.org/10.1073/pnas.1721355115

Qiu, J., Wang, B., & Zhou, C. (2020). Forecasting stock prices with long-short term memory neural network based on attention mechanism. *PLOS ONE*, *15*(1), e0227222. https://doi.org/10.1371/journal.pone.0227222

R Varian, H. (2014). Intermediate Microeconomics A Modern Approach Ninth Edition. *W. W. Norton & Company; International Student Edition*, *1*(Ninth). https://www.academia.edu/38113085/Intermediate_Microeconomics_A_Modern_Approach_Ninth_Edition

Reddi, S. J., Kale, S., & Kumar, S. (2018). *ON THE CONVERGENCE OF ADAM AND BEYOND*. 23.

Rothbard, M. N. (2009). *Man, economy, and state: A treatise on economic principles ; with Power and market: government and the economy*. MISES.

Ruder, S. (2017). An overview of gradient descent optimization algorithms. *ArXiv:1609.04747 [Cs]*. http://arxiv.org/abs/1609.04747

Rydberg, T. H. (2000). Realistic Statistical Modelling of Financial Data. *International Statistical Review / Revue Internationale de Statistique*, *68*(3), 233–258. JSTOR. https://doi.org/10.2307/1403412

Samuel, A. L. (n.d.). *Some Studies in Machine Learning Using the Game of Checkers*. Retrieved 25 June 2020, from https://core.ac.uk/reader/193399598

Sharif, T., Purohit, H., & Pillai, R. (2015). Analysis of Factors Affecting Share Prices: The Case of Bahrain Stock Exchange. *International Journal of Economics and Finance*, *7*(3), p207. https://doi.org/10.5539/ijef.v7n3p207

Shaughnessy, J. (2012). What works on Wall Street : the classic guide to the best-performing investment strategies of all time. New York: McGraw Hill.

Shah, Nisarg & Manubhai, Patel. (2015). A Comparative Study on Technical Analysis by Bollinger Band and RSI.

Sherstinsky, A. (2020). Fundamentals of Recurrent Neural Network (RNN) and Long Short-Term Memory (LSTM) Network. *Physica D: Nonlinear Phenomena*, *404*, 132306. https://doi.org/10.1016/j.physd.2019.132306

Skansi, S. (2018). Machine Learning Basics. In S. Skansi (Ed.), *Introduction to Deep Learning: From Logical Calculus to Artificial Intelligence* (pp. 51–77). Springer International Publishing. https://doi.org/10.1007/978-3-319-73004-2_3

StatsSorceress. (2017, May 27). *Forget Layer in a Recurrent Neural Network (RNN) -*. Data Science Stack Exchange. https://datascience.stackexchange.com/questions/19196/forget-layer-in-a-recurrent-neural-network-rnn

Strycharz, J., Strauss, N., & Trilling, D. (2018). The Role of Media Coverage in Explaining Stock Market Fluctuations: Insights for Strategic Financial Communication. *International Journal of Strategic Communication*, *12*(1), 67–85. https://doi.org/10.1080/1553118X.2017.1378220

Sun, R. (2019). Optimization for deep learning: Theory and algorithms. *ArXiv:1912.08957 [Cs, Math, Stat]*. http://arxiv.org/abs/1912.08957

Tepper, s., 29th August, 2017, The Buffett Challenge, hedge funds vs. index funds, 9 years on, CNBC Markets

Thaler, R. and Sunstein, C. (2009). Nudge : improving decisions about health, wealth, and happiness. New York: Penguin Books.

Tharavanij, P., Siraprapasiri, V., & Rajchamaha, K. (2015). Performance of technical trading rules: Evidence from Southeast Asian stock markets. *SpringerPlus*, *4*. https://doi.org/10.1186/s40064-015-1334-7

Urquhart, A., Batten, J. A., Lucey, B. M., McGroarty, F., & Peat, M. (2015). *Does Technical Analysis Beat the Market? – Evidence from High Frequency Trading in Gold and Silver* (SSRN Scholarly Paper ID 2652637). Social Science Research Network. https://doi.org/10.2139/ssrn.2652637

Virgilio, G. (2015, June 22). *High-frequency trading and the efficient market hypothesis*.

Wan, L., Zeiler, M., Zhang, S., LeCun, Y., & Fergus, R. (2013). *Regularization of Neural Networks using DropConnect*. 9.

Yuksel, E., & Bayrak, O. T. (2012). Cyclical Behavior of Stock Exchange Index by Sectors: A Case from Turkey. *Procedia - Social and Behavioral Sciences*, *62*, 947–951. https://doi.org/10.1016/j.sbspro.2012.09.161

Chapter 8 Appendix

Appendix 1 Project Proposal

1. Student details

First name	Stuart
Last (family) name	Manock
Edinburgh Napier matriculation number	40435519

2. Details of your programme of study

MSc Programme title	Computing F/T
Year that you started your diploma modules	2019

3. Project outline details

Please suggest a title for your proposed project. If you have worked with a supervisor on this proposal, please provide the name. You are strongly advised to work with a member of staff when putting your proposal together.

Title of the proposed project	The Incorporation of Financial Methods for Stock Market Predictions by Machine Learning Algorithms
Is your project appropriate to your programme of study?	Yes
Name of supervisor	Dr Andreas Steyven

4. Brief description of the research area - background

Background:

Technical indicators (TIs) provide deeper insight into companies and their setting within the broader financial market (Colby R.W., 1988). These TIs are used in groups to compound overall effectiveness. This further mitigates risk during financial trading. As computation power has continued to improve their utility has done so too. They are now used in a wide range of processes as a functioning tool. This holds true for financial trading where high volumes of data are produced.

Current ‘falsifiable’ theories (Efficient market) and are now starting to be demoted to just hypotheses (Malkiel, 2003) (Naseer & Tariq, 2015). This is due to a shift in how and why stocks and shares are traded (Lo et al., 2000). Long gone are the days of floor traders.

Computing in the field of data science has proved to be incredibly effective. The theoretical basis of - converting data into a mathematical container, feeding these as inputs to inform known outputs, reiterating this over large data sets to build more accurate models, - this has proved immensely effective at a wide range of tasks. These include natural language processing (Nadkarni et al., 2011), computer vision, facial recognition (Phillips et al., 2018), pattern recognition, Stock prediction (M et al., 2018), clustering & real value tasks.

TIs are not the only usable resource at the disposal of Deep Learning. Any and all information contributes to share price changes (Da et al., 2011). Online information is playing an increasingly important role in changing public impression of companies in the digital age (Li et al., 2018). Techniques to utilise this information in its many forms adds another field of study to inform trading decisions (Strycharz et al., 2018).

More data is available to investors. An increase in worldwide focus on Deep Learning has further improved the field with regard to its repertoire of tools to solve varying problems [1].

These realities coupled with the increased knowledge of stock market wide behaviour and its relationship to psychology make the feasibility of profitable algorithms for trading a possibility for individual investors (Frydman & Camerer, 2016). This coalition of fields is likely to continue to make waves due to its importance and relevance to so many of us.With this information I look forward to improving upon recent advances and testing their efficacy at this difficult and complex task.

5. Project outline for the work that you propose to complete

The idea for this research arose from:

Personal Interests in the stock market. Previous studies involving data processing, algorithm design and deep learning.

The aims of the project are as follows:

1. From the literature, critically assess statistical techniques used by traders to inform decisions.
2. Investigate how and why new market technologies (ML) are outcompeting old techniques.
3. Test if integration of old and new technologies can create more optimal results.
4. Design a set of experiments. Use to quantitate my research & inform financial indicator performance.

5. Provide an in-depth account of the methods used to allow reproduction.

6. Discuss and evaluate these results with regard to the literature.

7. Discuss weaker areas of experimentation and future steps that could further improve the robustness and completeness of these studies.

The main research questions that this work will address include:

1. **Which ML architectures and overall methodologies for stock market prediction perform the 'best'.**

2. What are the 'best' financial indicators.

3. How we define 'best'.

4. How other variables are influencing stock predictions.

5. Steps to automate and remove human intervention from impacting performance (~Genetic Algorithms).

The software development/design work/other deliverable of the project will be:

The creation of a program. Input – company ticker, prediction time, style of algorithm, useful metric inputs. Output – A program that lists this information in a simple but informative way. Potential to provide this in a user friend UI (time permitting).

The project deliverable will be evaluated as follows:

How much benefit these trading decisions have as an absolute and relative value.

The project will involve the following research/field work/experimentation/evaluation:

Find what metrics best predict share price movement. How deep learning deployment had influenced markets. Utilising & building software to scrap financial details, calculate metrics from said data, feed data into deep learning architectures to create beneficial trading decisions. Evaluate how they perform against funds, ftse 100, relevant sectors of market, other algorithms etc.

This work will require the use of specialist software:

Web scrappers/APIs, Data Processing and deep learning architecture implementation (Python).

This work will require the use of specialist hardware: Could be implemented using TPUs to improve efficiency and speed of insights. CPU & GPU implementation will likely be used for financial reasons.

The project is being undertaken in collaboration with: N/A?

6. References

Colby, R.W. (1988). The Encyclopedia of Technical Market Indicators.

Malkiel, B. G. (2003). The Efficient Market Hypothesis and its Critics. 47.

Naseer, M., & Tariq, Dr. Y. (2015). The Efficient Market Hypothesis: A Critical Review of the Literature. The IUP Journal of Financial Risk Management.

Lo, A. W., Mamaysky, H., & Wang, J. (2000). Foundations of Technical Analysis: Computational Algorithms, Statistical Inference, and Empirical Implementation (Working Paper No. 7613; Working Paper Series). National Bureau of Economic Research. https://doi.org/10.3386/w7613

Nadkarni, P. M., Ohno-Machado, L., & Chapman, W. W. (2011). Natural language processing: An introduction. Journal of the American Medical Informatics Association, 18(5), 544–551. https://doi.org/10.1136/amiajnl-2011-000464

Phillips, P. J., Yates, A. N., Hu, Y., Hahn, C. A., Noyes, E., Jackson, K., Cavazos, J. G., Jeckeln, G., Ranjan, R., Sankaranarayanan, S., Chen, J.-C., Castillo, C. D., Chellappa, R., White, D., & O'Toole, A. J. (2018). Face recognition accuracy of forensic examiners, superrecognizers, and face recognition algorithms. Proceedings of the National Academy of Sciences, 115(24), 6171–6176. https://doi.org/10.1073/pnas.1721355115

M, H., E.a., G., Menon, V. K., & K.p., S. (2018). NSE Stock Market Prediction Using Deep-Learning Models. Procedia Computer Science, 132, 1351–1362. https://doi.org/10.1016/j.procs.2018.05.050

Da, Z., Engelberg, J., & Gao, P. (2011). In Search of Attention. The Journal of Finance, 66(5), 1461–1499. https://doi.org/10.1111/j.1540-6261.2011.01679.x
Li, T., van Dalen, J., & van Rees, P. J. (2018). More than just Noise? Examining the Information Content of Stock Microblogs on Financial Markets. Journal of Information Technology, 33(1), 50–69. https://doi.org/10.1057/s41265-016-0034-2

Strycharz, J., Strauss, N., & Trilling, D. (2018). The Role of Media Coverage in Explaining Stock Market Fluctuations: Insights for Strategic Financial Communication. International Journal of Strategic Communication, 12(1), 67–85. https://doi.org/10.1080/1553118X.2017.1378220

Frydman, C., & Camerer, C. F. (2016). The Psychology and Neuroscience of Financial Decision Making. Trends in Cognitive Sciences, 20(9), 661–675. https://doi.org/10.1016/j.tics.2016.07.003

[1] - https://trends.google.com/trends/explore?date=all&geo=US&q=Deep%20Learning

7. Ethics

If your research involves other people, privacy or controversial research there may be ethical issues to consider (please see the information on the module website). If the answer below is YES then you need to complete a research Ethics and Governance Approval form, available on the website: http://www.ethics.napier.ac.uk.

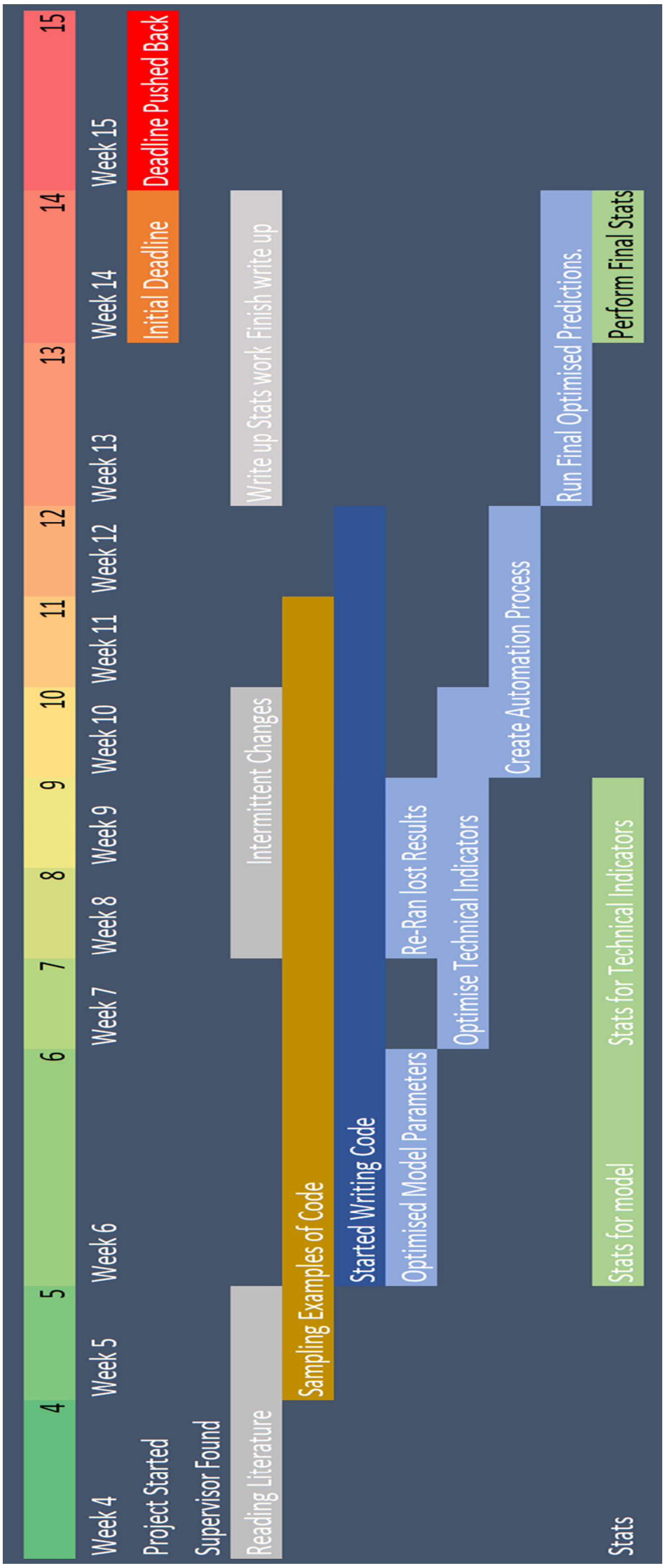
Does this project have any ethical or governance issues related to working with, studying or observing other people? (YES/NO)	Unsure? Financial success may come at others loss. Is that ethically wrong?

8. Confidentiality

If your research is being done in conjunction with an outside firm or organisation, there may be issues of confidentiality or intellectual property.

Does this project have any issues of confidentiality or intellectual property? (YES/NO)	YES, you cannot replicate any findings or release this information with any external party without expressed permission.

Appendix 2 Gantt Chart Overview



Appendix 3

EDINBURGH NAPIER UNIVERSITY

SCHOOL OF COMPUTING

PROJECT DIARY

Student: Stuart Manock

Supervisor: Dr Andreas Steyven

Date: 08/06/2020

Last diary date: N/A

Objectives:

Formulate of basic structure for the dissertation. Begin looking at articles that discuss common theory of economics. Explain how computers have begun to dominate the business of stock trading. Give a set of examples, focusing primary on predicting the stock market. Potentially add other components of computing that highlight the invasive impact technology has had on the financial industry.

Progress:

Very basic structure was achieved. Began to build a repertoire of articles using Zotero to continue explaining and building a comprehensive introduction. Started to examine Neural nets and their fruition into the highly utilisable technologies they are today. Explained how advancements in other areas of 'intelligent' functionality has paved the theory that these styles of computer algorithms may have utility in the financial sphere. Started to look more at NLP and it's huge impact and success in predicting market movements.

Supervisor's Comments:

EDINBURGH NAPIER UNIVERSITY

SCHOOL OF COMPUTING

PROJECT DIARY

Student: Stuart Manock

Supervisor: Dr Andreas Steyven

Date: 10/06/2020

Last diary date: 08/06/2020

Objectives:

Start coding. Look at building conventional statistical methods for determining the movement of stocks and shares.

Progress:

Found that most methods were very poor at deducing the predicted outcomes. Falls in line with my preconceived notions of their effectiveness. Shone a spotlight on the requirement of secular variables for these kinds of algorithms to work. This goes in complete contradiction to AI algorithms that learn and formulate their numerical expressions/functions to map to these complex and numerous variables.

Still unsure of the promise of this kind of masters work. However it is holding my attention and the barrier for success is set so low (51% accuracy) that I am optimistic.

Currently struggling to build a cohesive plan as many studies are well equipped at holding all but the dependant variable constant. These studies, helpful that they are, struggle to give insight into the internal workings of the AI algorithms. Unsure how to tackle this intrinsic issue with black box style algorithms.

However the poor results of statistical methods tend to make any better predictions stand out as successful. Additionally, these statistical results give a baseline metric for which I can compare the AI algorithms I develop. Also allows for the effectiveness of subsequent Genetic Algorithms (to perform the leg work of trading) to be compared against two styles of prediction which should create more robust results.

Supervisor's Comments:

EDINBURGH NAPIER UNIVERSITY

SCHOOL OF COMPUTING

PROJECT DIARY

Student: Stuart Manock

Supervisor: Dr Andreas Steyven

Date: 14/06/2020

Last diary date: 10/06/2020

Objectives:

Write up literary review on the statistical methods tested previously. Explain why they are not appropriate for stock market prediction with regard to novel techniques and a change in attitude towards stock price fluctuations. Set out plan for building LTSM model, research their effectiveness and the parameters best used to inform decisions.

Progress:

Literary review still feels a bit light. However, the main topics are there. Currently conversing with a friend who teaches economics to gain their insight on the relevance of economic theory with regard to building prediction software. Discussing the differences between numerical investments and other fields (ethical, long term, short term, ETFs, Funds, Commodities, FX etc)

Supervisor's Comments:

EDINBURGH NAPIER UNIVERSITY

SCHOOL OF COMPUTING

PROJECT DIARY

Student: Stuart Manock

Supervisor: Dr Andreas Steyven

Date: 16/06/20

Last diary date: 14/06/20

Objectives:

Complete first LTSM model for Lloyds banking group. Discuss pitfalls of this process. Write up discussion regarding NLP program for twitter posts.

Progress:

Data retrieval, processing and integration into the algorithm complete. Twitter NLP processing discussed with respect to previous studies. Pitfalls and potential fixes have also been laid out and discussed in depth in article. Still looking to tweak and design LTSM model around components of economic theory as 'features' for feature detection in the composite layers of the network created. Looking to fully create the entire component structure of the dissertation report. Only written parts of introduction, literature review and methodology for building statistical and AI style algorithms for stock price prediction.

Supervisor's Comments:

EDINBURGH NAPIER UNIVERSITY

SCHOOL OF COMPUTING

PROJECT DIARY

Student: Stuart M

Supervisor: Dr Andreas Syvant

Date: 25/06/2020

Last diary date:

Objectives:

Finish Proposal and Complete Initial Report

Progress:

Proposal and Initial report discussed at length during online meeting. Both forms fully finished and altered over the weekend. Not all component of technical indicators finished. Critical review not fully flushed out as summary of literature review provided. Potential task for coming week. Dockerisation of program and recursion for multiple examples to be completed for automnomous collection of data.

Supervisor's Comments:

SCHOOL OF COMPUTING

PROJECT DIARY

Student: Stuart M

Supervisor: Dr Andreas Syvant

Date: 01/07/2020

Last diary date: 25/06/2020

Objectives:

Continue onto further models and metrics for inputs to models. Continue critical review with regard to the technologies used.

Progress:

Critical review improving. Completed work on fully explaining how LSTM models work. Learnt to reshape *new* 4D model (4th dimension was intra-daily details) into 3D model for LSTM model. Moved onto tranformers following conference talk outlining their importance in dealing with these problems. Currently trying to implement attention portion of this model to determine which characteristics of daily trading is likely to impact work.

Still not competed the finalisation and full automation of collecting results.

Supervisor's Comments:

EDINBURGH NAPIER UNIVERSITY

SCHOOL OF COMPUTING

PROJECT DIARY

Student: Stuart M

Supervisor: Dr Andreas Syvant

Date: 07/07/2020

Last diary date: 01/07/2020

Objectives:

Build fully automated system for data collection.

Progress:

This has been improving but has been taking a great deal longer than first believed. However, this style of automation will improve the quality of the data and subsequent outcomes. Difficulty in normalising volumes of data (as companies may not exist for a long period of time). This has caused issues with API requests that are limited in their functionality (because they are free).

Currently working on building a "results datafame". This will then allow for the results to be outputted as a csv file. This will be imported into excel where excels functionality will assess the performance of the data. This should provide a clear and simple way to compare all kinds of metrics (parameter tuning, model choice, etc.).

Supervisor's Comments:

EDINBURGH NAPIER UNIVERSITY

SCHOOL OF COMPUTING

PROJECT DIARY

Student: Stuart M

Supervisor: Dr Andreas Syvant

Date: 17/07/2020

Last diary date: 07/07/2020

Objectives:

Build fully automated system for data collection.(Finished)

Do statistical analysis on results.

Progress:

Automated system finished. However incredibly buggy. Python does not handle high memoy usage effectively so allow of black screens and difficulty if I run the algorithm for too long.

Access to another computer would be ideal but hosting this software is unlikely to work well. Sorted out API request i

Supervisor's Comments:

EDINBURGH NAPIER UNIVERSITY

SCHOOL OF COMPUTING

PROJECT DIARY

Student: Stuart M **Supervisor:** Dr Andreas Syvant

Date: 23/07/2020 Last diary date: 17/07/2020

Objectives:

Does start of statistical analysis, found which model runs best based on RMSE.

Trying to continue with building the automation processing component, having difficulty organising the work.

Progress:

Cant find help on changing memory allocation for python. Making computer unusable during model development. Still not under time pressure yet. Doing further reading around automation process. Starting to think about TI parameters

Supervisor's Comments:

EDINBURGH NAPIER UNIVERSITY

SCHOOL OF COMPUTING

PROJECT DIARY

Student: Stuart M **Supervisor:** Dr Andreas Syvant

Date: 30/07/2020 Last diary date: 23/07/2020

Objectives:

Test Technical Indicators for performance

Finish automation process

Progress:

Doing TI performance without automation finished as need results to work with.

Will have to just incorporate the automation performance metric later.

Supervisor's Comments:

EDINBURGH NAPIER UNIVERSITY

SCHOOL OF COMPUTING

PROJECT DIARY

Student: Stuart M **Supervisor:** Dr Andreas Syvant

Date: 2/08/2020 Last diary date: 30/07/2020

Objectives:

Test Technical Indicators for performance (done) Finish automation process (in progress)

Progress:

Contacted brother for advice on building the automation system. Had a phone call and he helped my draw out diagrammatically what I wanted the software to do. Learnt a lot about handling arrays in python.

Supervisor's Comments:

EDINBURGH NAPIER UNIVERSITY

SCHOOL OF COMPUTING

PROJECT DIARY

Student: Stuart M **Supervisor:** Dr Andreas Syvant

Date: 5/08/2020 Last diary date: 02/08/2020

Objectives:

Finish automation process (done) model final automation results (in progress) Continue with write up

Progress:

Program now running, slow progress but achieving it. Now performing statistical analysis on results. Seem disappointing but hesitant to change my goals after starting.

Supervisor's Comments:

Do not change goals them.

EDINBURGH NAPIER UNIVERSITY

SCHOOL OF COMPUTING

PROJECT DIARY

Student: Stuart M

Supervisor: Dr Andreas Syvant

Date: 13/08/2020 Last diary date: 05/08/2020

Objectives:

Continue with write up (in progress)

Progress:

What I have achieved does not match fully with my intial goals. Therefore, removing less relevant sections of write up has created a lot of work. Focused down on the specific question because of the scope of my question is too wide.

Supervisor's Comments:

That is fine to do.

Appendix 4 - PC Hardware Architecture

```
-----------          APICs          ----------------------------------------------------------------------
Socket 0             (total threads:8)
        -- Core 0 (ID 0)  -- Thread 0      0        -- Thread 1      1
        -- Core 1 (ID 1)  -- Thread 2      2        -- Thread 3      3
        -- Core 2 (ID 2)  -- Thread 4      4        -- Thread 5      5
        -- Core 3 (ID 3)  -- Thread 6      6        -- Thread 7      7
-----------          Timer          ----------------------------------------------------------------------
ACPI timer           3.580 MHz      Perf timer      10.000 MHz      Sys timer       1.000 KHz
-----------          Processors Information         ------------------------------------------------------------------
---
Socket 1                            ID = 0
        Number of cores                     4 (max 4) Number of threads      8 (max 8)
        Manufacturer                GenuineIntel Name                        Intel Core i7 6700HQ
        Codename                    Skylake Specification            Intel(R) Core(TM) i7-6700HQ CPU
@ 2.60GHz
        Package (platform ID)       Socket 1440 FCBGA (0x5)
        CPUID                       6.E.3
        Extended CPUID                      6.5E    Core Stepping            R0
        Technology                  14 nm           TDP Limit                45.0 Watts
        Tjmax                       100.0 °C        Core Speed               3091.7 MHz
        Multiplier x Bus Speed      31.0 x 99.7 MHz
        Base frequency (cores)      99.7 MHz        Base frequency (ext.)    99.7 MHz
        Stock frequency             2600 MHz        Max frequency            3500 MHz
        Instructions sets           MMX, SSE, SSE2, SSE3, SSSE3, SSE4.1, SSE4.2, EM64T, VT-x,
AES, AVX, AVX2, FMA3, TSX
        Microcode Revision          0xCC
        L1 Data cache               4 x 32 KBytes, 8-way set associative, 64-byte line size
        L1 Instruction cache        4 x 32 KBytes, 8-way set associative, 64-byte line size
        L2 cache                    4 x 256 KBytes, 4-way set associative, 64-byte line size
        L3 cache                    6 MBytes, 12-way set associative, 64-byte line size
        Max CPUID level                     00000016h      Max CPUID ext. level     80000008h
        Cache descriptor            Level 1, D, 32 KB, 2 thread(s)
        Cache descriptor            Level 1, I, 32 KB, 2 thread(s)
        Cache descriptor            Level 2, U, 256 KB, 2 thread(s)
        Cache descriptor            Level 3, U, 6 MB, 16 thread(s)
        FID/VID Control             yes     Turbo Mode              supported, enabled
        Max non-turbo ratio         26x     Max turbo ratio         35x
        Max efficiency ratio        8x      O/C bins                none
        Ratio 1 core                35x Ratio 2 cores               33x Ratio 3 cores               32x
        Ratio 4 cores               31x
        IA Voltage Mode                     PCU adaptive IA Voltage Offset 0 mV
        GT Voltage Mode                     PCU adaptive GT Voltage Offset          0 mV
        LLC/Ring Voltage Mode               PCU adaptive LLC/Ring Voltage Offset    0 mV
        Agent Voltage Mode          PCU adaptive    Agent Voltage Offset     0 mV
        TDP Level                   45.0 W @ 26x    TDP Level                35.0 W @ 21x
        Voltage 0                   1.10 Volts (VID) Voltage 1               +0.00 Volts (IA Offset)
        Voltage 2                   +0.00 Volts (GT Offset)  Voltage 3       +0.00 Volts (LLC/Ring
Offset)
        Voltage 4         +0.00 Volts (System Agent Offset)         Power 00        26.37 W (Package)
        Power 01                    22.88 W (IA Cores)      Power 02        0.12 W (GT)
        Power 03                    3.36 W (Uncore)         Power 04        2.28 W (DRAM)
```

Clock Speed 0 3091.66 MHz (Core #0) Clock Speed 1 3091.66 MHz (Core #1)
Clock Speed 2 3091.66 MHz (Core #2) Clock Speed 3 3091.66 MHz (Core #3)
Core 0 max ratio 35.0 (effective 35.0) Core 1 max ratio 35.0 (effective 33.0)
Core 2 max ratio 35.0 (effective 33.0) Core 3 max ratio 35.0 (effective 33.0)

----------- BIOS ---

UEFI Yes Chipset Yes

Northbridge Intel Skylake-H rev. 07 Southbridge Intel Skylake-H PCH rev. 31
Graphic Interface PCI-Express
Memory Type DDR4 Memory Size 16 GBytes
Channels Dual
Memory Frequency 1063.9 MHz (1:16)
CAS# latency (CL) 15.0 RAS# to CAS# delay (tRCD) 15
RAS# Precharge (tRP) 15 Cycle Time (tRAS) 36
Row Refresh Cycle Time (tRFC) 278 Command Rate (CR) 2T
Uncore Frequency 3091.7 MHz Host Bridge 0x1910

----------- Memory SPD ---

DIMM # 1
SMBus address 0x52 Memory type DDR4
Module format SO-DIMM
SDRAM/Module Manufacturer (ID) SK Hynix (AD00000000000000000000)
Size 8192 MBytes Max bandwidth DDR4-2132 (1066 MHz)
Manufacturing date Week 04/Year 16 Nominal Voltage 1.20 Volts
EPP no XMP no AMP no
JEDEC timings table CL-tRCD-tRP-tRAS-tRC @ frequency
JEDEC #1 9.0-9-9-22-31 @ 666 MHz JEDEC #2 11.0-11-11-27-38 @ 814 MHz
JEDEC #3 12.0-12-12-30-42 @ 888 MHz JEDEC #4 13.0-13-13-32-45 @ 962 MHz
JEDEC #5 14.0-14-14-35-49 @ 1037 MHz JEDEC #6 15.0-15-15-36-50 @ 1066 MHz
JEDEC #7 16.0-15-15-36-50 @ 1066 MHz
DIMM # 2
SMBus address 0x52 Memory type DDR4
Module format SO-DIMM
SDRAM/Module Manufacturer (ID) SK Hynix (AD00000000000000000000)
Size 8192 MBytes Max bandwidth DDR4-2132 (1066 MHz)
Manufacturing date Week 04/Year 16 Nominal Voltage 1.20 Volts
EPP no XMP no AMP no
Mainboard Model 0N7TVV (0x000001B5 - 0x0052B6C0)

----------- Hardware Monitors ---

Hardware monitor D3D
Hardware monitor NVIDIA NVAPI
Voltage 0 0.93 Volts [0x39D] (GPU) Temperature 0 62 degC (143 degF) [0x3E] (GPU)
Clock Speed 0 405.00 MHz [0x195] (Graphics) Clock Speed 1 405.00 MHz [0x195] (Memory)

Clock Speed 2 n.a. (Processor)
Hardware monitor D3D Hardware monitor Intel I/O
Temperature 0 65 degC (149 degF) [0x41] (GPU) Clock Speed 0 0.00 MHz [0x0]
----------- DMI ---
SMBIOS Version 2.8
DMI BIOS
vendor Dell Inc. version 1.7.0 ROM size 16384 KB
DMI System Information
manufacturer Dell Inc. product XPS 15 9550 SKU 06E4
DMI Baseboard
vendor Dell Inc. model 0N7TVV
revision A00
DMI OEM Strings
string[0] Dell System string[1] 1[06E4] string[2] 3[1.0]
string[3] 12[www.dell.com] string[4] 14[1] string[5] 15[0]
string[6] 27[38411684174]
DMI Processor
manufacturer Intel(R) Corporation model Intel(R) Core(TM) i7-6700HQ CPU @ 2.60GHz
clock speed 2600.0 MHz FSB speed 100.0 MHz multiplier 26.0x
DMI Physical Memory Array
location Motherboard usage System Memory correction None max capacity 32 GB max# of devices 2
DMI Memory Device
designation DIMM A format SODIMM type DDR4
total width 64 bits data width 64 bits size 8 GB
DMI Memory Device
designation DIMM B format SODIMM
type DDR4 total width 64 bits
data width 64 bits size 8 GB
----------- Storage ---
Name INTEL SSDSCKKF256H6 SATA 256GB Revision LBFD16N
Capacity 238.5 GB Type Fixed, SSD Bus Type SATA (11)
Features SMART, TRIM Volume c:\, 237.2 GBytes
----------- API ---
NVIDIA I/O API NVIDIA NVAPI
----------- Display Adapters ---
ID 0x1070107 Name NVIDIA GeForce GTX 960M
Board Manufacturer Dell Codename GM107 Technology 28 nm
Memory size 2 GB Memory type GDDR5 Memory vendor - Hynix
PCI device bus 1 (0x1), device 0 (0x0), function 0 (0x0)
Vendor ID 0x10DE (0x1028) Model ID 0x139B (0x06E4)
Revision ID 0xA2 Performance Level 0
Core clock 405.0 MHz Memory clock 405.0 MHz
Software ---

Windows Version Microsoft Windows 10 (10.0) Home 64-bit (Build 18362)