



# Opinion Mining and Social Media Sentiment Analysis in the Prediction of Cryptocurrency Prices

School of Mathematical, Physical and Computational Sciences

Individual Project - CS3IP16

Student: Andrew Sotheran  
Student Number: fr005432  
Supervisor: Kenneth Boness  
Word Count: Place Holder

Submission date: Place Holder

## Abstract

The volatility of the stock markets is an aspect that is both hard to predict and to mitigate particularly when relating to the cryptocurrency market. Commodities such as cryptocurrencies are profoundly volatile and have attracted investors in an attempt to make quick profits on the market. These financial commodities are subject to the whim of public confidence and platforms such as Twitter and Facebook are most notably utilised to express opinions. Extrapolating sentiment from such platforms has been used to gain insight into topics across industries, thus applying it to crypto-market analysis could serve to show a relationship between public opinion and market change.

This project looks into public perception of the cryptomarket, by analysing Bitcoin-related tweets per hour for sentiment changes that could indicate a correlation to market fluctuations in the near future. This is achieved by training a recurrent neural network on the severity changes of historical sentiment and price over the past year every hour. The predictions are then shifted forward in time by 1 hour to indicate the corresponding Bitcoin price interval.

## Acknowledgements

I would like to express my gratitude to Dr. Kenneth Boness for his continued support and guidance throughout this project.

Secondly, I want to express gratitude to PhD. Jason Brownlee, of Machine Learning Mastery for having clear and thorough explanations of machine learning concepts and metrics.

I would also like to thank my family for their support during the development of this project.

## Glossary

Bull(ish)/Bear(ish) Markets - Relates to a trend of the market price increasing and decreasing respectively

Highs/Lows - The highest and lowest trading price of a giving period

Fiat Currency - A currency without intrinsic value that has been established as money

BTC - Bitcoin's stock symbol

Twitter - Online social media platform, which allows users to post information or express opinions through messages called "Tweets"

Tweets - The name given for messages posted on the Twitter platform, which are restricted to 280 characters.

Hashtag - Is a keyword or phrase used to describe a topic and allows the tweets to be categorised.

Fomo (Fear of Missing Out) - Is used to describe buying behaviour when stocks are moving suddenly and more buyers appear to enter all of a sudden.

Shorting - Or short sale, is the sale of an asset that the investor buys shares and immediately sells them, hoping to make a profit from buying later at a lower price.

Doubling Down - Is to take further risk on a stock by doubling effort/investment in a hope and attempt to raise the price

RNN - Recurrent Neural Network

LSTM - Long-Short Term Memory Neural Network

# Contents

<b>Abstract</b>	<b>1</b>
<b>Acknowledgements</b>	<b>2</b>
<b>Glossary</b>	<b>3</b>
<b>Introduction</b>	<b>6</b>
<b>Problem Articulation</b>	<b>8</b>
Problem Statement . . . . .	8
Stakeholders . . . . .	8
Project Constraints . . . . .	8
<b>Literature Review</b>	<b>9</b>
Existing Tools . . . . .	9
Related research . . . . .	9
Data Collection . . . . .	10
Twitter and Twitter API . . . . .	10
Tweepy Python Package . . . . .	11
Sentiment Analysis . . . . .	11
Natural Language Processing . . . . .	11
Valence Aware Dictionary and sEntiment Reasoning . . . . .	12
Neural Networks . . . . .	12
Recurrent Neural Network (RNN) . . . . .	13
Long-Short Term Memory (LSTM) . . . . .	14
Kera and TensorFlow . . . . .	15
Optimisers . . . . .	15
Loss . . . . .	16
Regularisation - Dropouts . . . . .	16
Machine Learning . . . . .	16
Naive Bayes . . . . .	16
Other Techniques . . . . .	16
Ngrams . . . . .	16
Scoring and Validation . . . . .	16

<b>Solution Approach</b>	<b>17</b>
Solution Summary . . . . .	17
Data flow Overview . . . . .	17
Packages, Tools and Techniques . . . . .	17
<b>System Design and Implementation</b>	<b>18</b>
Data collection . . . . .	18
Price Time-series Data . . . . .	18
Data processing . . . . .	18
Preprocessing . . . . .	18
Spam Filtering . . . . .	18
Sentiment Analysis . . . . .	18
VADER . . . . .	18
Recurrent Neural Network . . . . .	18
Training and Testing Model . . . . .	18
Validation . . . . .	18
Future Prediction Forecasting . . . . .	18
<b>Testing: Verification and Reflection</b>	<b>19</b>
<b>Discussion: Contribution and Reflection</b>	<b>20</b>
Limitations . . . . .	20
<b>Conclusion and Future Improvements</b>	<b>21</b>
Conclusion . . . . .	21
Future Improvements . . . . .	21
<b>Appendices</b>	<b>24</b>
Appendix A - Project Initiation Document . . . . .	24
Appendix B - Log book . . . . .	37

## Introduction

The premise of this project is to investigate into whether the sentiment in social media has a correlation to the prices of cryptocurrencies and how this could be used to predict future changes in the price.

The chosen cryptocurrency that will be focused in this project will be the currency that has the most community and backing and has been known to lead other fiat currencies, Bitcoin (BTC). Bitcoin is seen as one, if not the first cryptocurrency to bring a wider following to the peer-to-peer token transaction scene since 2009. Although it was not the first token to utilise blockchain technology, it allowed investors to openly trade a public cryptocurrency which provided pseudonymous means of transferring funds through the internet. Thus it has been around longer than most of the other fiat currencies and is the most popular crypto-token due to it's larger community base.

Most financial commodities are subject to the whim of public confidence and are the core of it's base value. A platform that is frequently used for the public to convey their opinions on a commodity is that of Twitter which provides arguably biased information and opinions. Whether the opinions present a basis in facts or not, they are usually taken at face value and can influence the public opinion of given topics. As Bitcoin has been around since 2009 the opinions and information on the commodity are prevalent through the platform. In the paper *Sentiment Analysis of Twitter Data for Predicting Stock Market Movements* by Majhi *et al.* [1] 2.5 million tweets on Microsoft were extracted from Twitter, sentiment analysis and logistical regression performed on the data yielded 69.01% accuracy for a 3-day period on the increase/decrease in stock price. These results showed a *"good correlation between stock market movements and the sentiments of public expressed in Twitter"*.

The background of this project is in response to the volatility of the cryptocurrency market, which can fluctuate at a moments notice and can be seen to be social media driven. The history of the price of Bitcoin and what was being discussed on the currency around it's most volatile period to-date, Nov-2017 to Feb-2018, shows a strong bullish trend which saw Bitcoin reach a \$19,500 high in mid-Dec. While social media, such as Twitter, during that period was had an extremely positive outlook on the cryptocurrency. The trend was short lived and saw the market crash only a month later, with only a couple of sell-offs, expected for the holidays rush, accompanied by negative outlooks posted on social media turned the market against itself which saw the longest bearish market in Bitcoin's history and is still trying to recover today.

Due to how volatile the crypto-market can be, there is a need to either mitigate or to anticipate where the markets are heading. As the crypto-market and Bitcoin are affected by socially constructed opinions, either through Twitter, news articles or other forms of media, there is a way to perform the latter, where the prices of Bitcoin could be predicted based on the sentiment gathered from social media outlets.

The aim of this project is to create a tool that gathers tweets from Twitter, obtains the overall sentiment score of the given text while gathering historical price data for the time period gathering occurs. Features are then extracted from the gathered data and used in a neural network to ascertain whether the price of the currency can be predicted from the correlation between the sentiment and price history of the data.

This report will discuss the justifications for the project and the problems it will be attempting to resolve, the stakeholders that would benefit the most from this system and what this project will not attempt to accomplish. Similar tools will be critiqued and examined for their feature set and credibility in the literature review along with current sentiment analysers, algorithms, natural language processing techniques and neural networks in their respective topics and comparing their accuracy for this project purpose. The solution approach will discuss the decisions and reasoning behind choosing the techniques and tools used for this project and will outline the requirements for this project. Implementation of the chosen techniques and tools, with the discussion of important

functions of the system will formulate the implementation section of this report with an in-detail explanation of the function's use and data flow of the system.

# Problem Articulation

## Problem Statement

The key problems this project attempts to address are that of, an open-source system available to the public that aids in the analysis and prediction of BTC. The accuracy of open-source tools and technology when applied to the trading market scene and to identify whether there is a correlation between Twitter sentiment and BTC price fluctuation. While there are existing tools only a few are available to the public and only provide basic functionality, while others are kept in-house of major corporations who invest into this problem domain.

The other issue presented here is that assuming perfect accuracy can be achieved is naive. As this project will only be using existing tools and technologies thus, there are limitations to accuracy that can be obtained. One of that being the suitability of the tools, there are no open-source sentiment analysers for stock market prediction, thus finding a specifically trained analyser for the chosen domain in highly unlikely. In relation, finding the most suitable machine learning or neural network is equally important as this will determine the accuracy of the predictions and should be discussed in the literature review.

The accuracy and suitability of various machine learning methods and neural networks are a known issue in their respective domains, this investigation should be carried out to determine their suitability for their needed use in this project and should be detailed in the literature review.

This project will focus on the investigation of these technologies and tools to justify whether it is feasible to predict the price of BTC based on historical price and the sentiment gathered from Twitter. Limitations of the system and it's accuracy in predictions should be investigated and discussed to determine the implemented solution is the more suitable compared to other methods.

## Stakeholders

The stakeholders of this system would be those looking to invest in the cryptocurrency markets, in this projects regard, specifically into Bitcoin. Private, Business and public investors would benefit from an open-source system such as this as it will both provide a basis for decisions on whether to invest or not. Public investors, general people looking to invest in the cryptocurrency market, will be those stakeholders that will benefit the most from such a system. Due to the lack of any open-source tools available, these stakeholders could be seen as being left in the dark when it comes to predicting the direction of Bitcoin where Businesses and Enterprises will have a one up, due to having an internal system for predictions.

## Project Constraints

This project will not attempt to justify the accuracy of the chosen algorithm or tools over other algorithms. It will be discussed in the literature review the justifications made on why the chosen algorithm and tools have been used for this project over the others, but accuracy will not be directly compared.

This project will only be coded to predict an hour ahead as the model will be trained on an hourly basis as the data is gathered per hour. Predictions for further in the future can be modelled but will be seen as a future improvement to the system.

The detail of a interface may be subject of change through this project due to time contraints and the focus being the investigation of the impact social media has on market predictions.

# Literature Review

## Existing Tools

An aspect that this project will be attempting to address is that, at the time of writing, there are a limited amount of systems available to the public that either provide sentiment analysis or predictions of the crypto-market. Additionally, none known that combine both sentiment and price analysis to make said predictions on the direction of the market.

Such tools are usually provided by exchanges which correspond the amount of positive and negative sentiments with a suggestion to buy and sell. These tools, however, are vague in their suggestions as they don't provide any further analysis on when the best time would be to conduct an action on the market, and simply display the number of tweets per sentiment level. A well-known cryptocurrency tracking site, Coingecko provides a basic sentiment analysis tool for their top 30 ranking cryptocurrencies tracked on the site. This tool shows the sentiment analysis of tweets from Twitter every hour for a given cryptocurrency. This is displayed as a simple pill on the page showing the ratios of positive, neutral and negative tweets. *See Appendix C for visual representation*

## Related research

There has been a plentiful amount of research conducted in this problem domain. Numerous theses globally have been published in recent years on the topic of cryptocurrency market predictions and analysis, and even more, research conducted on general stock markets further back.

The thesis written by *Evita Stenqvist and Jacob Lonno* of the *KTH Royal Institute of Technology* [2] investigates the use of sentiment expressed through micro-blogging such as Twitter can have on the price fluctuations of Bitcoin. Its primary focus was creating an analyser for the sentiment of tweets more accurately *"by not only taking into account negation, but also valence, common slang and smileys"*, than that of former researchers that *"mused that accounting for negations in text may be a step in the direction of more accurate predictions."*. This would be built upon the lexicon-based sentiment analyser VADER to ascertain the overall sentiment scores were grouped into time-series for each interval from 5 minutes to 4 hours, along with the interval prices for Bitcoin. The model chosen was a naive binary classified vectors of predictions for a certain threshold to *"ultimately compare the predictions to actual historical price data"*. The results of this research suggest that a binary classification model of varying threshold over time-shifts in time-series data was *"lackluster"*, seeing the number of predictions decreasing rapidly as the threshold changed. This research is a good basis of starting research upon, as it suggests tools such as VADER for sentiment analysis and that the use of a machine learning algorithm would be a next step in the project that would yield better more accurate results.

Another thesis written by *Pagolu, Venkata Sasank and Reddy Kamal Nayan, Panda Ganapati and Majhi, Babita* [1] on *"Sentiment Analysis of Twitter Data for Predicting Stock Market Movements"* 2.5 million tweets on Microsoft were extracted from Twitter, sentiment analysis and logistical regression performed on the data yielded 69.01% accuracy for a 3-day period on the increase/decrease in stock price. These results showed a *"good correlation between stock market movements and the sentiments of the public expressed in Twitter"*. Using various natural language pre-processing tweets for feature extraction such as N-gram representation the sentiment from tweets were extrapolated. Both Word2vec and a random forest classifier were compared for accuracy, Word2vec being chosen over the machine learning model. Word2vec, being a group of related shallow two-layer neural network models to produce word embeddings.

A topic that reoccurs in various papers and theses is that of the use and focus of regression techniques and machine learning methods. Few implement a fully fledged neural network, the above paper attempts to use a simple network to achieve predictions of classification of sentiment for stock market movement then correlated this with historical data of prices. An article posted on "Code Project" by Intel Corporation [3] compares the accuracy of three machine learning algorithms; Random Forest, Logistic Regression and Multi-Layer Perceptron (MLP) classifiers on predicting the price fluctuations of Bitcoin with embedded price indices. Results showing "*that using the MLP classifier (a.k.a. neural networks) showed better results than logistic regression and random forest trained models*". This assumption can be backed up by the results from a thesis posted on IEEE [4] which compares a Bayesian optimised recurrent neural network and a Long Short Term Memory (LSTM) network. Showing the LSTM model achieving "*the highest classification accuracy of 52% and a RMSE of 8%*". With an interest in neural networks personally and with little papers utilising them for this purpose a neural network will thus be implemented, and the accuracy of one's predictions with use of sentiment analysis data analysed and discussed.

## Data Collection

### Twitter and Twitter API

Twitter is a micro-blogging platform that was launched in 2006 and provides it's users the ability to publish short messages of 140 characters. The messages published could be of any form, from news snippets, advertisement, or the prevalent publication of opinions which allowed a platform of extensive diversity and knowledge wealth. As of the time of writing, the message character limit was increased to 280 characters, the platform has over 300 million monthly active users and around 1 million tweets are published per day. Due to the length restriction and the primary use of the platform to express opinions Twitter is seen as a gold mine for opinion mining.

The Twitter API has an extensive range of endpoints that provide access from streaming tweets for a given hashtag, obtaining historical tweets for a given time-period and hashtag, posting tweets on a users account and to change settings on a user account with authentication. The exhaustive range of features provided by these endpoints makes data collection from Twitter straight forward as one can target a specific endpoint for the required data. Due to Twitter being the target for opinion mining within this project the Twitter API will ultimately need to be utilised. This can either be used for the gathering of historical tweets or streaming current tweets for the #Bitcoin hashtag.

There are, however, limitations and rate limits imposed on users of the API. Twitter employs a tiering system for the API - Standard, Premium and Enterprise tiers, each of which provides different amounts of access for data collection. If the API were used to capture historical data for a span of 3 months, each tier is allowed to obtain varying amounts of data for different durations; [5]

- A Standard user would be able to capture 100 recent tweets for the past 7 days
- A Premium user would be allowed to capture up to 500 tweets per request for a 30-day span and will have access to a full-archive search to query up to 100 tweets per request for a given time period, with a 50 request limit per month
- An Enterprise user would be able to capture up to 500 tweets per unlimited requests for a 30-day span and will be able to query the full-archive of tweets for a given hashtag up to 2000 tweets per unlimited amount of requests for a given time period

Each tier has individual costs while the standard user negating this as a basic tier. Due to only being elegable for the Premium tier for educational purposes, historical data gathering will be limited to

100 tweets per request with a limitation of 50 requests per month. Furthermore streaming tweets is an Enterprise feature which rules out the the Twitter API for use of streaming current real-time data [6].

### **Tweepy Python Package**

Tweepy is a python package for accessing the Twitter API. It fundamentally accomplishes the same means if one to conduct a GET request to the Twitter API, except it simplifies this into a simple to use API that is easier to implement and automate in python [7]. Consequently, it builds upon the existing Twitter API to provide features such as automated streaming of provided hashtags to the API. It realises this by initialising a listener instance for a provided set of API credentials, handling authentication, connections, creating and destroying sessions. Due to Twitter's streaming API being only available to Enterprise users [6], using Tweepy to stream data for a given hashtag will provide the real-time data needed.

## **Sentiment Analysis**

In short, sentiment analysis is the process and discovery of computationally identifying and categorising the underlining opinions and subjectivity expressed in written language. This process determines the writer's attitude towards a particular topic as either being positive, neutral or negative in terms of opinion, known as polarity classification.

### **Natural Language Processing**

Polarity classification is the focus of sentiment analysis and is a well-known problem in natural language processing that has had significant attention by researchers in recent years [1][2][4][8]. Traditional approaches to this have usually been classified to dictionary-based approaches that use a pre-constructed sentiment lexicons such as VADER or usually confined to machine learning approaches. The later requires an extensive amount of natural language pre-processing to extrapolate vectors and features from given text, this is then fed into a machine learning classifier which attempts to categorise words to a level of sentiment polarity. Natural language pre-processing techniques that would be required for this approach would consist of;

- Tokenisation: The act of splitting a stream of text into smaller units of typographical tokens which isolate unneeded punctuation.
- Removal of domain specific expressions that are not part of general purpose English tokenisers - a particular problem with the nature of the language used in tweets, with @-mentions and #-hashtags
- Stopword removal: Are commonly used words (such as "the", "in", "a") that provide no meaning to the sentiment of a given text
- Stemming: Is used to replace words with common suffixes and prefixes, as in "go" and "goes" fundamentally convey the same meaning. A stemmer will replace such words with their reduced counterparts
- Term Probability Identification and Feature Extraction: This is a process that involves identifying the most frequently used words in a given text, by using a probability type approach on a pre-defined dataset which classifies a range of texts as with overall negative or positive a machine learning algorithm is trained to classify these accordingly.

The former, seen and has been proven to provide higher accuracy than traditional machine learning approaches [9], and need little pre-processing conducted on the data as words have a pre-defined sentiment classification in a provided lexicon. Although these lexicons can be complex to create, they generally require little resources to use and add to.

### Valence Aware Dictionary and sEntiment Reasoning

VADER is a combined lexicon and rule-based sentiment analysis tool that is specifically attuned to sentiments expressed in social media, and works well on texts from other domains. It is capable of detecting the polarity of a given text - positivity, neutrality, and negativity [10]. VADER uses a human-centric approach to sentiment analysis, combining qualitative analysis and empirical validation by using human raters to rate level of sentiment for words in its lexicon. Vader also has emoticon support which maps these colloquialisms have pre-defined intensities in its lexicon, which makes VADER specifically suitable for the social media domain where the use of emoticons, utf-8 emojis and slang such as "Lol" and "Yolo" are prevalent within text. Additionally, VADER is provided as a lexicon and a python library under the MIT license, thus means that it is open-source software. This means that the lexicon can be altered and added to making it able to be tailored to specific topic domains.

VADER was constructed by examining and extracting features from three pre-existing well-established and human-validated sentiment lexicons [10] - (LIWC) Linguistic Inquiry and Word Count, (ANEW) Affective Norms for English Words, and (GI) General Inquirer. This is supplemented with additional lexicon features "*commonly used to express sentiment in social media text (emoticons, acronyms and slang)*" [10] and uses "*wisdom-of-the-crowd*" approach [11] to establish a point of estimations of sentiment valence for each lexical feature candidate. This was evaluated for the impact of grammatical and syntactical rules and 7,500+ lexical features, with mean valence " $i_{\bar{z}}$  zero, and  $SD i= 2.5$ " as a human-validated "gold-standard" sentiment lexicon. [10] *Section 3.1*

VADER is seen as a "Gold Standard" for sentiment analysis, in the paper for VADER, [10] *A Parsimonious Rule-based Model for Sentiment Analysis of Social Media Text*, it was compared against 11 other "*highly regarded sentiment analysis tools/techniques on a corpus of over 4.2K tweets*" for polarity classification across 4 domains. Results showing VADER, across Social media text, Amazon reviews, movie reviews and Newspaper editorials, consistently outperforming other sentiment analysis tools and techniques showing a particular trend in performing significantly higher on analysis of sentiment in tweets. [10] *Section 4: Results*

### Neural Networks

A neural network is a set of perceptrons modelled loosely after the human brain that is designed to recognise patterns in whatever domain it is intended to be trained on. A neural network can consist of multiple machine perceptrons or clustering layers in a large mesh network and the patterns they recognise are numerical which are contained in vectors. Pre-processed data, confined and processed into pre-defined vector labels, are used to teach a neural network for a given task. While this differs from how an algorithm is coded to a particular task, neural networks cannot be programmed directly for the task. The requirement is for them to learn from the information by use of different learning strategies; [12][13]

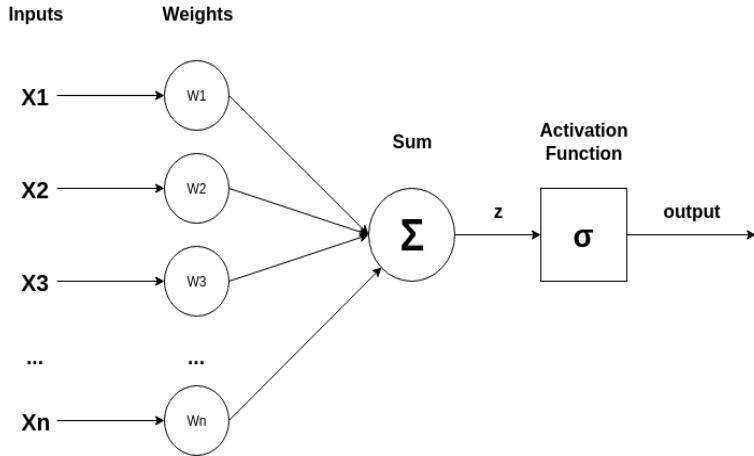


Figure 1: Basic perceptron layout

- Supervised learning: Simplest of the learning forms, where a dataset have been labeled which indicate the correct classified data. The input data is learned upon until the desired result of the label is reached [14]
- Unsupervised learning: Is training the with a dataset without labels to learn from. The neural network analyses the dataset with a cost function which tells the neural network how far off target a prediction was. The neural network then adjusts input weights in attempt to increase accuracy. [13]
- Reinforced learning: The neural network is reinforced with positive results and punished for negative results causing a network to learn over iterations.

### Recurrent Neural Network (RNN)

The type of neural network that is of focus for this project will be that of a Long-Short Term Memory (LSTM), however, it is important to understand how this is an extension of a Recurrent Neural Network (RNN) and how the underlying network works.

Recurrent Neural Networks (RNN) are a robust and powerful type of neural network and is considered to be among the most encouraging algorithms for use of classification, due to the fact of having internal memory. RNNs are designed to recognise patterns in sequences of presented data or most suitably, time-series data, genomes, handwriting and stock market data. Although RNNs were conceptualised and invented back in the 1980s [15] they've only really shown their potential in recent years, with the increase of computational power due to the level of sequencing and internal memory store to retrain. Due to this 'internal' memory loop, RNNs are able to remember data and adjust neurons based on failures and alternating parameters. The way this is accomplished, knowing how a standard neural network such as a feed-forward network, should initially be understood. [16]

A standard, feed-forward neural network has a single data flow with an input layer, through hidden computational layers, to an output layer. Therefore any node in the network will never see the same data again. However, in an RNN data is cycled through a loop over the same node, thus two inputs into the perceptron. Decisions are influenced by previous data that it has previously learned from if any, which in turn affects output and the weights of the network. [17]

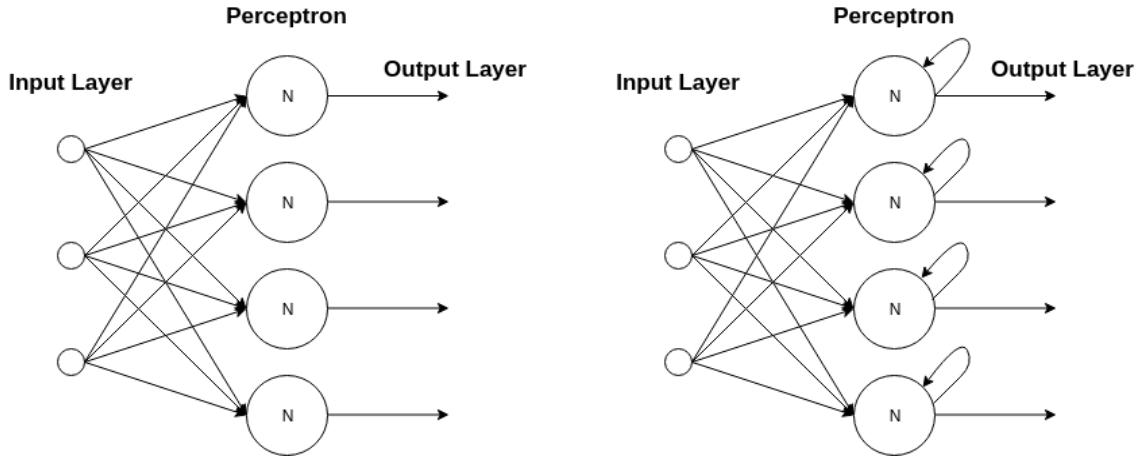


Figure 2: Feed-forward network (left) vs Recurrent Neural network (right)

The act of tweaking weights to alter the processing of the next iteration of data in an RNN is called backpropagation, which in short means going back through the network to find the partial derivatives of the error with respect to the weights after output has occurred. Along with gradient descent, an algorithm that adjusts the weights up or down depending on which would reduce the error. There are however a few obstacles of RNNs;

- Exploding Gradients: Is when gradient descent assigns high importance to the weights. As in the algorithm assigns a ridiculously high or low value for the weights on iteration which can cause overflow and result in NaN values [18]
- Vanishing Gradients: Is when the values of a gradient are small enough that weights cannot be altered and the model stops learning. [19]

These issues are overcome by the concept of Long-Short Term Memory neural networks, coined by *Sepp Hochreiter and Juergen Schmidhuber, 1997* [20].

### Long-Short Term Memory (LSTM)

LSTMs are an extension of recurrent neural networks capable of learning long-term dependancies and were conceptualised by *Sepp Hochreiter and Juergen Schmidhuber, 1997* [20]. LSTMs were explicitly designed to avoid long-term dependency problems such as exploding and vanishing gradients. As they are an extension of RNNs they operate in almost the exact same manner, but stores the actual gradients and weights in memory which allows for LSTMs to read, write and alter the values. A way of explaining how this works is seeing the memory block as a gated cell, where 'gated' is that the cell decides whether or not to store or alter data in its memory based input data and the importance assigned to it. In a sense it learns over time of which values and data is important.

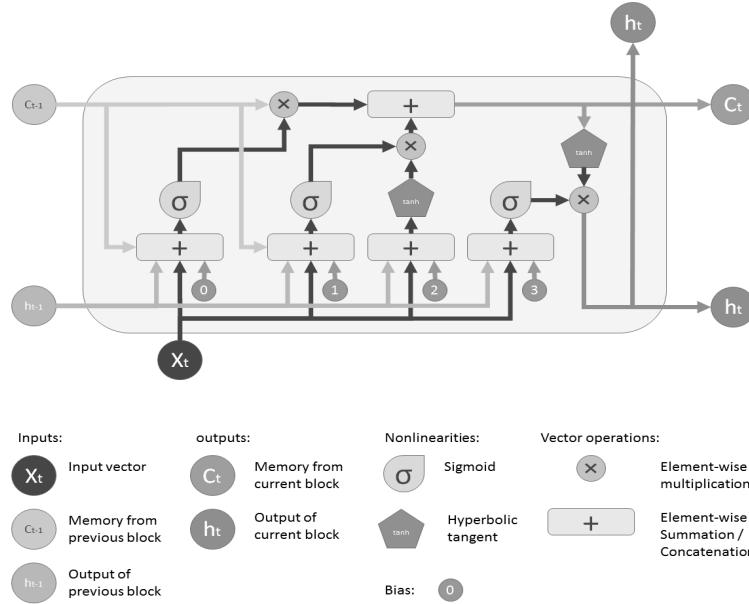


Figure 3: A conceptual design of an LSTM cell bank - from Medium article by Shi Yan: Understanding LSTM and its diagrams[21]

The network takes in three initial inputs, input of current time step, output from the previous LSTM unit if any, and the memory of the previous unit. Outputs,  $H_t$  - output of current network, and  $C_t$  - the memory of the current unit. [21]

The various steps of the network decide what information is thrown away from the cell state, through use of a 'forget gate' which is influenced by the calculations of sigmoid memory gates which influence how much of old and new memory is used and  $X_t$ , and merged together based upon importance. The section of the cell that controls the outflow memory  $H_t$  and  $C_t$  determines how much of the new memory should be used by the next LSTM unit. *For a more in-detailed explanation of exactly how the calculations are made see [20],[21] and [22].*

As mentioned in the foremost section of related work an LSTM network, the use of one would be optimal for the given problem domain over the use of machine learning algorithms.

## Kera and TensorFlow

### Optimisers

ADAGrad optimizer, RMSprop, Adam

**Loss**

**Regularisation - Dropouts**

**Machine Learning**

**Naive Bayes**

**Other Techniques**

**Ngrams**

**Scoring and Validation**

F-score, Precision, Accuracy, Validation Datasets, Cross-Validation

## **Solution Approach**

### **Solution Summary**

A system will be created that will utilise

### **Data flow Overview**

### **Packages, Tools and Techniques**

# **System Design and Implementation**

## **Data collection**

### **Price Time-series Data**

Historical data of Bitcoin prices can be obtained through may means,

## **Data processing**

### **Preprocessing**

#### **Tweet Filtering**

#### **Text Cleaning**

#### **Ngram based Language detection filtering**

#### **Spam Filtering**

#### **Tweet Processing**

### **Naive Bayes model**

## **Sentiment Analysis**

### **VADER**

### **Recurrent Neural Network**

### **Training and Testing Model**

### **Validation**

### **Future Prediction Forecasting**

## Testing: Verification and Reflection

## **Discussion: Contribution and Reflection**

### **Limitations**

## **Conclusion and Future Improvements**

### **Conclusion**

### **Future Improvements**

Shifting the intial data by and hour and sequencing over previous data

## References

- [1] V. S. Pagolu, K. N. Reddy, G. Panda, and B. Majhi, "Sentiment analysis of twitter data for predicting stock market movements," in *2016 international conference on signal processing, communication, power and embedded system (SCOPES)*, IEEE, 2016, pp. 1345–1350. [Online]. Available: <https://arxiv.org/pdf/1610.09225.pdf>.
- [2] J. L. Evita Stenqvist, "Predicting bitcoin price fluctuation with twitter sentiment analysis," Diva, 2017. [Online]. Available: <http://www.diva-portal.org/smash/get/diva2:1110776/FULLTEXT01.pdf>.
- [3] Intel-Corporation, "Stock predictions through news sentiment analysis," Code Project, 2017. [Online]. Available: <https://www.codeproject.com/Articles/1201444/Stock-Predictions-through-News-Sentiment-Analysis>.
- [4] S. C. Sean McNally Jason Roche, "Predicting the price of bitcoin using machine learning," in *2018 26th Euromicro International Conference on Parallel, Distributed and Network-based Processing (PDP)*, IEEE, 2018, pp. 344–347. [Online]. Available: <https://ieeexplore.ieee.org/abstract/document/8374483>.
- [5] Twitter, "Search tweets," Twitter Developers, 2018. [Online]. Available: <https://developer.twitter.com/en/docs/tweets/search/overview>.
- [6] ———, "Consuming streaming data," Twitter Developers, 2018. [Online]. Available: <https://developer.twitter.com/en/docs/tutorials/consuming-streaming-data.html>.
- [7] J. Roesslein, "Streaming with tweepy," Tweepy, 2009. [Online]. Available: [http://docs.tweepy.org/en/v3.4.0/streaming\\_how\\_to.html](http://docs.tweepy.org/en/v3.4.0/streaming_how_to.html).
- [8] S. N. Mehrnoush Shamsfard, "Using linked data for polarity classification of patients experiences," in *Journal of Biomedical Informatics*, Elsevier, 2015, pp. 6–19. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S1532046415001276>.
- [9] L. P. T. Chedia Dhaoui Cynthia M. Webster, "Social media sentiment analysis: Lexicon versus machine learning," in *Journal of Consumer Marketing, Volume 34. Issue 6*, Emerald Insight, 2017. [Online]. Available: <https://www.emeraldinsight.com/doi/pdfplus/10.1108/JCM-03-2017-2141>.
- [10] C. Hutto and E. Gilbert, "Vader: A parsimonious rule-based model for sentiment analysis of social media text," in *Eighth International Conference on Weblogs and Social Media (ICWSM-14)*, Ann Arbor, MI, 2014. [Online]. Available: <https://www.aaai.org/ocs/index.php/ICWSM/ICWSM14/paper/download/8109/8122>.
- [11] W. Kenton, "Wisdom of crowds," Investopedia, 2018. [Online]. Available: <https://www.investopedia.com/terms/w/wisdom-crowds.asp>.
- [12] Skymind, "A beginner's guide to neural networks and deep learning," in *A.I. Wiki*, Skymind, 2018. [Online]. Available: <https://skymind.ai/wiki/neural-network>.
- [13] J. DeMuro, "What is a neural network," in *World of tech*, techradar, 2018. [Online]. Available: <https://www.techradar.com/uk/news/what-is-a-neural-network>.
- [14] F. Bach, "Supervised dictionary learning," in *Advances in neural information processing systems*, NIPS Proceedings, 2009, pp. 1033–1040. [Online]. Available: <http://papers.nips.cc/paper/3448-supervised-dictionary-learning>.
- [15] D. E. Rumelhart, G. E. Hinton, and R. J. Williams, "Learning internal representations by error propagation," California Univ San Diego La Jolla Inst for Cognitive Science, 1985. [Online]. Available: <https://apps.dtic.mil/docs/citations/ADA164453>.
- [16] Skymind, "A beginner's guide to lstms and recurrent neural networks," in *A.I. Wiki*, Skymind, 2018. [Online]. Available: <https://skymind.ai/wiki/lstm>.

- [17] N. Donges, “Recurrent neural networks and lstm,” Towards Data Science, 2018. [Online]. Available: <https://towardsdatascience.com/recurrent-neural-networks-and-lstm-4b601dd822a5>.
- [18] P. Jason Brownlee, “A gentle introduction to exploding gradients in neural networks,” Machine Larning Mastery, 2017. [Online]. Available: <https://machinelearningmastery.com/exploding-gradients-in-neural-networks/>.
- [19] S. D. S. Team, “Recurrent neural networks (rnn) - the vanishing gradient problem,” Super Data Science, 2018. [Online]. Available: <https://www.superdatascience.com/blogs/recurrent-neural-networks-rnn-the-vanishing-gradient-problem>.
- [20] S. Hochreiter and J. Schmidhuber, “Long short-term memory,” in *Neural computation, Volume 9. 8*, MIT Press, 1997, pp. 1735–1780. [Online]. Available: <https://www.bioinf.jku.at/publications/older/2604.pdf>.
- [21] S. Yan, “Understanding lstm and its diagrams,” Medium, Mar 13, 2016. [Online]. Available: <https://medium.com/mlreview/understanding-lstm-and-its-diagrams-37e2f46f1714>.
- [22] C. Olah, “Understanding lstm networks,” 2015. [Online]. Available: <https://colah.github.io/posts/2015-08-Understanding-LSTMs>.

## **Appendices**

### **Appendix A - Project Initiation Document**

Displayed on the following pages below.

## Individual Project (CS3IP16)

Department of Computer Science  
University of Reading

# Project Initiation Document

### PID Sign-Off

<b>Student No.</b>	24005432
<b>Student Name</b>	Andrew Sotheran
<b>Email</b>	andrew.sotheran@student.reading.ac.uk
<b>Degree programme (BSc CS/BSc IT)</b>	BSc CS
<b>Supervisor Name</b>	Kenneth Boness
<b>Supervisor Signature</b>	
<b>Date</b>	

## SECTION 1 – General Information

### Project Identification

1.1	<b>Project ID</b> (as in handbook) N/A
1.2	<b>Project Title</b>  Cryptocurrency market and value prediction tracking
1.3	<b>Briefly describe the main purpose of the project in no more than 25 words</b>  To provide a means to predict the value of cryptocurrencies that will aid in investor decision making in investment of the market

### Student Identification

1.4	<b>Student Name(s), Course, Email address(s)</b> e.g. Anne Other, BSc CS, a.other@student.reading.ac.uk
	Andrew William Sotheran BSc CS Andrew.sotheran@student.reading.ac.uk

### Supervisor Identification

1.5	<b>Primary Supervisor Name, Email address</b> e.g. Prof Anne Other, a.other@reading.ac.uk
1.6	<b>Secondary Supervisor Name, Email address</b> Only fill in this section if a secondary supervisor has been assigned to your project

### Company Partner (only complete if there is a company involved)

1.7	<b>Company Name</b> N/A
1.8	<b>Company Address</b> N/A
1.9	<b>Name, email and phone number of Company Supervisor or Primary Contact</b> N/A

## SECTION 2 – Project Description

2.1	<p><b>Summarise the background research for the project in about 400 words. You must include references in this section but don't count them in the word count.</b></p>
	<p>To create a tool that aims to predict the price of cryptocurrencies that aids in investor decisions. Research will need to be conducted into the following topics that surround data mining, machine learning and artificial neural networks.</p> <p>This research will consist along the lines of;</p> <p>Natural Language processing and analysis – To analyse and process fed in data gathered through RSS data feeds and social media feeds, through the underlying tasks of Natural language processing.</p> <p>Content categorisation (search and indexing, duplication detection), Topic discovery and modelling (Obtain meanings and themes within the data and perform analytic techniques), sentiment and semantic analysis (which will identify the mood and opinions within the data), summariser (to summarise a block of text and disregard the rest).</p> <p>Machine learning algorithms: The three types of machine learning (Supervised, Unsupervised and Reinforced)</p> <p>The types of common algorithms used, each of these will be researched to identify the most suitable for this project and only one will be used: (Linear Regression, Logistic Regression, Decision Tree, SVM, Naive Bayes, kNN, K-Means, Random Forest, Dimensionality Reduction Algorithms, Gradient Boosting algorithms (GBM, XGBoost, LightGBM, CatBoost).</p> <p>Artificial Neural Networks: To identify the drawbacks and benefits of using them or other computational models within machine learning. Recurrent Neural networks and 3rd generation Neural Networks.</p> <p>Data mining: To investigate the different techniques and algorithms used (Same as the ones listed above for machine learning including C4.5, Apriori, EM, PageRanks, AdaBoost and CART) these will be researched and the most appropriate identified.</p> <p>To investigate techniques: for storing and processing large amount of data, such as Hadoop, Elasticsearch utilities, Graphing and data modelling and visualisation.</p> <p>To identify appropriate libraries for python or C for each of the topics above to aid in the creation of this project. Libraries such as:</p> <p>Natural Language Toolkit (NLTK) – python</p> <p>Pandas - python</p> <p>Sklearn - python</p> <p>Numpy – python - scientific computation for working with arrays</p> <p>Matplotlib - python - data visualisation</p> <p>Investigate into types of databases. Sql and nosql for a storage medium between receiving data and feeding it into the machine learning algorithm.</p> <p>Investigate into the use of REST API and other web-service based technologies (GRPC, Elasticsearch)</p> <p>Investigate into frameworks for the thin client, such as Angular vs React, Nodejs, Leaflet.js, charts.js</p> <p>Additionally Web scraping may be needed if certain website that don't either have an API or JSON for the data needed.</p> <p><a href="https://www.sas.com/en_gb/insights/analytics/what-is-natural-language-processing-nlp.html">https://www.sas.com/en_gb/insights/analytics/what-is-natural-language-processing-nlp.html</a> <a href="https://blog.algorithmia.com/introduction-natural-language-processing-nlp/">https://blog.algorithmia.com/introduction-natural-language-processing-nlp/</a> <a href="https://gerardnico.com/data_mining/algorithm">https://gerardnico.com/data_mining/algorithm</a> <a href="https://www.analyticsvidhya.com/blog/2017/09/common-machine-learning-algorithms/">https://www.analyticsvidhya.com/blog/2017/09/common-machine-learning-algorithms/</a> <a href="https://www.kdnuggets.com/2015/05/top-10-data-mining-algorithms-explained.html">https://www.kdnuggets.com/2015/05/top-10-data-mining-algorithms-explained.html</a> <a href="https://www.datasciencecentral.com/profiles/blogs/artificial-neural-network-ann-in-machine-learning">https://www.datasciencecentral.com/profiles/blogs/artificial-neural-network-ann-in-machine-learning</a> <a href="http://scikit-learn.org/stable/index.html">http://scikit-learn.org/stable/index.html</a> <a href="https://grpc.io/docs/">https://grpc.io/docs/</a></p>

2.2	<p><b>Summarise the project objectives and outputs in about 400 words.</b></p> <p>These objectives and outputs should appear as tasks, milestones and deliverables in your project plan. In general, an objective is something you can do and an output is something you produce – one leads to the other.</p>
	<p>To produce a thin web client that provides a dashboard that provides tangible and useful information to users such as; Their current price (Updated every 5 minutes), exchange rates, network hashrates, historical price data. It will also display statistics about sentiment analysis conducted on social media about the currency, graphical predictions on what the price may be, in a given time, and will also compare this to other currencies for aid in investment.</p> <p>To produce significant research into the topics in and around data mining, machine learning and Artificial Neural network and the underlying tasks and algorithms used, the efficiency, drawbacks and advantages of each to identify the most suitable for the use in this project.</p> <p>To produce a system that analyses a data set obtained through social media feeds and posts on news sites regarding crypto currencies. It should perform sentiment analysis using Natural Language processing and analysis techniques to identify features and identifies the type of sentiment in the data and categorises it for machine learning.</p> <p>To utilise machine learning techniques and algorithms to produce a system that learns from historical data to predict to an extent the possible future price of a given currency. To compare this with the use of an Artificial Neural Network and to analyse the drawbacks of both.</p>
2.3	<p><b>Initial project specification - list key features and functions of your finished project.</b></p> <p>Remember that a specification should not usually propose the solution. For example, your project may require open source datasets so add that to the specification but don't state how that data-link will be achieved – that comes later.</p>
	<p>The finished project should provide a thin client single page application. This will provide a means to users the ability to view various statistics on crypto currencies on a dashboard that incorporates text analysis through natural language analysis, and will utilise various machine learning and data mining techniques to provide price predictions to the users. The nature and level of this will depend on the research conducted into the areas of data mining, machine learning, natural language processing and artificial neural networks, along with the algorithms used.</p> <p>The data set will be created from scratch for this project as it will require the gathering of data from numerous sources and performing text analysis on them to for the data needed. Data sets for the characteristic and data for the currencies can be obtained from pre-existing data sets such as:</p> <p><a href="https://www.kaggle.com/sudalairajkumar/cryptocurrencypricehistory">https://www.kaggle.com/sudalairajkumar/cryptocurrencypricehistory</a>  <a href="https://www.kaggle.com/jessevent/all-crypto-currencies">https://www.kaggle.com/jessevent/all-crypto-currencies</a></p> <p>Web scraping may be included if certain news/social media websites do not provide an API or RSS feed for the analysis engine to perform text analysis on</p> <p>Additionally, there will be a server between the analysis/prediction engine and the thin client that will maintain a database, either SQL or NoSQL, that will hold statistics about the currencies and data about the price predictions about the currencies. It will not hold any of the data used in the analysis engine, as this database will only hold data available to the end users.</p>

<b>2.4</b>	<p><b>Describe the social, legal and ethical issues that apply to your project. Does your project require ethical approval? (If your project requires a questionnaire/interview for conducting research and/or collecting data, you will need to apply for an ethical approval)</b></p>
	<p>The project will not be handling any user related data, therefore it does not need ethical approval.</p>
<b>2.5</b>	<p><b>Identify and lists the items you expect to need to purchase for your project. Specify the cost (include VAT and shipping if known) of each item as well as the supplier.</b> e.g. item 1 name, supplier, cost</p>
	<p>None Needed</p>
<b>2.6</b>	<p><b>State whether you need access to specific resources within the department or the University e.g. special devices and workshop</b></p>
	<p>Possibly a server to host the database and analysis engine on to perform the computation necessary, and a server to host the thin client.</p>

## SECTION 3 – Project Plan

<b>3.1</b>	<b>Project Plan</b>		
	Split your project work into sections/categories/phases and add tasks for each of these sections. It is likely that the high-level objectives you identified in section 2.2 become sections here. The outputs from section 2.2 should appear in the Outputs column here. Remember to include tasks for your project presentation, project demos, producing your poster, and writing up your report.		
<b>Task No.</b>	<b>Task description</b>	<b>Effort (weeks)</b>	<b>Outputs</b>
<b>1</b>	<b>Background Research</b>		
1.1	Investigate into RPC frameworks and REST APIs	0.3	To identify the type of API/RPC framework that would be most suitable
1.2	Research into Natural Language processing and analysis techniques	0.5	To get an understanding of how NLP works and how it could be used
1.3	Research into the use of machine learning – types and algorithms	0.5	To grasp how ML paradigms work and how this project will use it
1.4	Research into the application of Neural Networks – drawbacks and advantages of using them	0.3	To identify whether there will be a need for a neural network or ML paradigms can be used instead
1.5	Research techniques for storing and processing large amount of data, such as Hadoop, spark or Elasticsearch utilities.	1	To understand the uses, application and whether the use of these are more viable solution than standard ML practices
1.6	Identify appropriate libraries for data modelling and visualisation, NLP and Machine Learning	1	To identify what libraries will aid in the construction of this project
1.7	Investigate into frameworks for the front-end thin clients	0.3	To identify what frameworks the thin client should be used with, along with drawbacks and advantages
1.8	Research web scraping techniques	0.3	To understand the application of these techniques and learn how to apply them
<b>2</b>	<b>Analysis and design</b>		
2.1	Resolve issues discovered by background research	0.2	...
2.2	Identify limitations discovered from research and what is not feasible	0.1	...
2.3	UML Diagrams/ XUML	0.2	
2.4	Wire frames for frontend	0.1	
2.5	Data Flow	0.1	
2.6	User Flow	0.1	
<b>3</b>	<b>Develop prototype</b>		
3.1	Develop thin client	2	
3.2	Develop analysis Engine	4	
3.3	Develop Prediction Engine	3	
3.4	Develop Unit tests	2	
<b>4</b>	<b>Testing, evaluation/validation</b>		
4.1	Unit testing	1	
4.2	Acceptance Testing	0.8	
4.3	User testing	0.8	
<b>5</b>	<b>Assessments</b>		
5.1	write-up project report	2	Project Report
5.2	produce poster	0.5	Poster
5.3	Log book	0.5	

<b>TOTAL</b>		<b>Sum of total effort in weeks</b>	<b>21.9</b>	
--------------	--	-------------------------------------	-------------	--

#### SECTION 4 - Time Plan for the proposed Project work

For each task identified in 3.1, please *shade* the weeks when you'll be working on that task. You should also mark target milestones, outputs and key decision points. To shade a cell in MS Word, move the mouse to the top left of cell until the cursor becomes an arrow pointing up, left click to select the cell and then right click and select 'borders and shading'. Under the shading tab pick an appropriate grey colour and click ok.

Project stage	START DATE: 10/2018 <enter the project start date here>												
	0-3	3-6	6-9	9-12	12-15	15-18	18-21	21-24	24-27	27-30	30-33	33-36	36-39
<b>1 Background Research</b>													
Investigate into RPC frameworks and REST APIs													
Research into Natural Language processing													
Research into the use of machine learning –													
Research into the application of Neural													
Research techniques for storing and													
Identify appropriate libraries for data													
Investigate into frameworks for the front-													
Research web scraping techniques													
<b>2 Analysis/Design</b>													
Resolve issues discovered by background													
Identify limitations discovered from													
UML Diagrams/ XUML													
Wire frames for frontend													
Data Flow													
User Flow													

<b>3 Develop prototype.</b>												
Develop thin client												
Develop analysis Engine												
Develop Prediction Engine												
Develop Unit tests												
<b>4 Testing, evaluation/validation</b>												
Unit testing												
Acceptance Testing												
User testing												
<b>5 Assessments</b>												
write-up project report												
produce poster												
Log book												

### RISK ASSESSMENT FORM

Assessment Reference No.			Area or activity assessed:		
Assessment date					
Persons who may be affected by the activity (i.e. are at risk)	Andrew Sotheran				

**SECTION 1: Identify Hazards** - Consider the activity or work area and identify if any of the hazards listed below are significant (tick the boxes that apply).

1.	Fall of person (from work at height)	<input type="checkbox"/>	6.	Lighting levels	<input type="checkbox"/>	11.	Use of portable tools / equipment	<input type="checkbox"/>	16.	Vehicles / driving at work	<input type="checkbox"/>	21.	Hazardous fumes, chemicals, dust	<input type="checkbox"/>	26.	Occupational stress	<input type="checkbox"/>
2.	Fall of objects	<input type="checkbox"/>	7.	Heating & ventilation	<input type="checkbox"/>	12.	Fixed machinery or lifting equipment	<input type="checkbox"/>	17.	Outdoor work / extreme weather	<input type="checkbox"/>	22.	Hazardous biological agent	<input type="checkbox"/>	27.	Violence to staff / verbal assault	<input type="checkbox"/>
3.	Slips, Trips & Housekeeping	<input checked="" type="checkbox"/>	8.	Layout , storage, space, obstructions	<input type="checkbox"/>	13.	Pressure vessels	<input type="checkbox"/>	18.	Fieldtrips / field work	<input type="checkbox"/>	23.	Confined space / asphyxiation risk	<input type="checkbox"/>	28.	Work with animals	<input type="checkbox"/>
4.	Manual handling operations	<input type="checkbox"/>	9.	Welfare facilities	<input type="checkbox"/>	14.	Noise or Vibration	<input type="checkbox"/>	19.	Radiation sources	<input type="checkbox"/>	24.	Condition of Buildings & glazing	<input type="checkbox"/>	29.	Lone working / work out of hours	<input type="checkbox"/>
5.	Display screen equipment	<input checked="" type="checkbox"/>	10.	Electrical Equipment	<input checked="" type="checkbox"/>	15.	Fire hazards & flammable material	<input type="checkbox"/>	20.	Work with lasers	<input type="checkbox"/>	25.	Food preparation	<input type="checkbox"/>	30.	Other(s) - specify	<input checked="" type="checkbox"/>

**SECTION 2: Risk Controls** - For each hazard identified in Section 1, complete Section 2.

Hazard No.	Hazard Description	Existing controls to reduce risk	Risk Level (tick one)			Further action needed to reduce risks (provide timescales and initials of person responsible)
			High	Med	Low	
3	Tripping over wires	Cable management is at a minimum, none are currently properly cable managed and kept out of way			x	Sufficient cable management needed, cables tied together and moved out of way of feet
5	Eye strain from looking at a monitor	Current screen contrast and brightness is acceptable		x		To have periodic breaks from the screen
Name of Assessor(s)			SIGNED			
Review date						

**Health and Safety Risk Assessments** – continuation sheet

<b>Assessment Reference No</b>	
<b>Continuation sheet number:</b>	

**SECTION 2 continued: Risk Controls**

Hazard No.	Hazard Description	Existing controls to reduce risk	Risk Level (tick one)			Further action needed to reduce risks <i>(provide timescales and initials of person responsible for action)</i>
			High	Med	Low	
<b>Name of Assessor(s)</b>			<b>SIGNED</b>			
<b>Review date</b>						

## **Appendix B - Log book**

The log book for this project is a physical book and was handed to the School of Computer Science. Due to being a physical book, it cannot be inserted here.