A Hybrid Approach To Predictive Modelling Of Stock Prices Using Machine Learning

Antonio J Cefalo Yaghmour
School of Computing
University of Portsmouth
Portsmouth, United Kingdom
up828398@myport.ac.uk

Alexander Gegov
School of Computing
University of Portsmouth
Portsmouth, United Kingdom
alexander.gegov@port.ac.uk

Mo Adda
School of Computing
University of Portsmouth
Portsmouth, United Kingdom
mo.adda@port.ac.uk

***Abstract*— This paper uses both accuracy and efficiency in a hybrid way as performance evaluation indicators for the following machine learning algorithms: Linear Regression (LR) and Long Short-Term Memory Neural Network (LSTM). For this purpose, these algorithms are simulated to predict stock prices from the historical daily prices of the National Association of Securities Dealers Automated Quotations (NASDAQ Exchange) using the Amazon (AMZN) dataset. The accuracy of the algorithms is evaluated comparatively for both training and testing by means of the Root Mean Square Error (RMSE) using a benchmark and a novel approach. The efficiency of the algorithms is also evaluated comparatively in a similar way by means of Computational Time (CT).**

The proposed novel approach performs better than the benchmark approach in terms of accuracy during training for both algorithms. The novel approach outperforms the benchmark approach during testing for the LSTM algorithm. However, the benchmark approach outperforms the novel approach during testing for the LR algorithm. The simulation results show that accuracy is usually improved by increasing the number of training iterations for the algorithms and the amount of input data from the dataset. Also, this improvement of accuracy is achieved at the expense of only marginal loss of efficiency. The proposed novel approach provides more flexibility to investors on the stock market due to its hybrid nature. This allows investment decisions to be made on the basis of individual preferences with regard to accuracy and efficiency.

Keywords—machine learning, predictive modelling, stock prices, linear regression, long short-term memory neural network

I. INTRODUCTION

This section presents an overview of machine learning and modelling. It also states the aim, objectives, outcomes, benefits of the research and introduces the overall structure of the paper.

Machine learning based modelling is an effective approach that can be used by people who would like to gain a deeper understanding of the stock market. Nowadays, with the popularity of the stock market investments increasing, this approach generates an impact and more attraction to individuals that are looking for new ways of financial development and economic growth. That is why machine learning based modelling has been implemented in many software applications in recent years for the purpose of helping investors by means of stock price prediction. As mentioned in (flatworldsolutions.com)

Machine learning is concerned with the investigation and computer modelling of learning processes in all of their forms. It is currently organised around three main research areas: 1) Task-oriented studies, also known as the engineering method, are the creation and study of learning processes to increase performance in a predetermined series of tasks. 2) Cognitive simulation is the study and simulation of human learning processes using computers. 3) Theoretical analysis—a non-domain-specific theoretical exploration of the space of potential learning methods and algorithms.

In addition to 1), 2) and 3) above, machine learning includes also exploration of alternative learning approaches, such as the discovery of various induction algorithms, the scope and limitations of some methods, the knowledge that must be accessible to the learner, the issue of dealing with incomplete training data, and the development of general techniques applicable in several task domain. All of these areas of exploration for machine learning are equally important as 1), 2), 3) (Marcus Thorström 2017)

Machine learning has become important and useful in recent years due to the wide variety of applications as well as remarkable capacity to adjust and provide solutions to complex problems in a timely, accurate, and productive manner. The iterative aspect of machine learning is important because models will evolve independently as they are exposed to new data. They use previous computations to generate consistent, repeatable decisions and outcomes. It's an old science that's gaining new traction thanks to the increased availability of large data sets (Shelley Elmblad, 2021)

This paper goes through the implementation, design and thought process of creating a system that would handle specific amounts of datasets to determine the prediction of stocks. By creating a software system where several machine learning algorithms are combined and analysed simultaneously, the paper provides meaning to large data to get a better understanding of how the accuracy and efficiency of these algorithms can be affected and how important it is for the specific outcome expected.

These days, potential investors want to get not only an accurate but also an efficient prediction of stock prices depending on their individual circumstances and preferences. Therefore, it is important to test some established machine learning algorithms and see how they work for stock prediction in terms of both accuracy and efficiency. In this context, it is desirable to increase the accuracy of the algorithms without compromising their efficiency.

This paper looks at the increase of prediction accuracy for stock prices making use of large data. However, the latter can compromise the efficiency of the models. For this reason, this project is focused on achieving higher accuracy by using more data without compromising the efficiency of the solution as much.

A. Aim

The aim of this paper is to systematically explore suitability and applicability of several popular machine learning algorithms for financial modelling of the stock market. This aim is achieved through the following objectives.

B. Objectives

1. To evaluate accuracy and efficiency of stock market prediction in a complementary way
2. To visualise results from algorithm training and testing in an integrated way
3. To execute several machine learning algorithms in a simultaneous way

C. Outcomes and Benefits

Although, an exact prediction of the outcomes in a research might be impossible to state this early in the paper section, a quite precise as to the nature and scope of the outcomes and as to who might benefit is attempted.

Expected outcomes are:

- Produce stock price prediction software which meets all the objectives from section 1.3 and benefits investors on the stock market
- Identify weaknesses and strengths of different machine learning algorithms and how they can complement each other

Potential benefits are:

- Attract more investors to help companies grow and make a bigger contribution to the economy
- Help users and employees of the stock market industry to understand better stock price prediction

D. Paper Organization

This paper shows in detail the functionality and performance characteristics of a specific software developed for financial modelling. Different sections are shown throughout the paper where each of the sections to follow next have a sequence of the previous.

Section 1, the introduction section, where it clearly states the problem & background of the research, aims and objectives of the paper as well as outcomes and benefits of the software.

Section 2, the literature review section, focuses on conducting research introducing different machine learning algorithms, software artefacts, a conclusion and a summary section of the section. This section is a less technical description but more of the research.

Section 3, the methodology section, which provides a more detailed description of the research, shows how the management of the project was developed throughout. In more technical details the research part of the algorithms used and software methodology of the approach chosen, conclusion and summary of the section.

Section 4, the requirements section, which includes the research and software strands of the artefact as well as the summary of the section.

Section 5, the design section, shows machine learning charts for the specific algorithms chosen and diagrams of the software implemented for the design model and summary of the section.

Section 6, the implementation section, shows screenshots of the graph results, a software artefact part explaining the code and summary of the section.

Section 7, the testing and evaluation section, shows a benchmark approach of the results of each algorithm and evaluates requirements met from a previous section as well as the summary of the section.

Section 8, the conclusion section, mentions contributions being accomplished, dissemination, further research that can take place, wider context, general reflections.

The paper finishes with references containing relevant information sources used and appendices section with supplementary material for the project.

II. Literature Review

A. Machine Learning Algorithms for Stock Prediction

There are different types of machine learning artefacts available for stock prediction on the market (Hiransha et al., 2018). However, depending upon preferences and goals different software artefacts use particular algorithms. This section discusses on a general scale, what types of algorithms there are out there based on research that people have used for their stock prediction software in order to decide what algorithms to choose for the development of the software in this paper. Stock price prediction is usually considered in the context of two main groups that are highlighted in subsections *B* and *C* below whether it is short term or long term. As machine learning algorithms are fairly generic and used for both short and long term prediction, they are then described in sections 2.2.3-2.2.5 within a general overview.

B. Short-Term Prediction

Short-Term predictions are linked to short-term investments which are held for less than a year. Short-term investments are also named by many people as temporary investments. Many short-term investments are typically sold by many after a period of only 3-12 months as people with not a lot of experience expect quicker returns in a short time. Some common examples of short term investments include money market accounts, high-yield savings accounts, government bonds, and treasury bills. Normally, short term investments are of highly liquid assets (Troy Segal, 2021).

Referring to financial assets, short-term investments have a few additional requirements that are owned by a company. Recorded in a separate account and listed in a particular asset section of the organization, short-term investments in this context are investments that a company has made available for individuals, employees or investors in general that are expected to be sold within one year as (Katelyn Peters, 2021) stated in investopedia.com

C. Long-Term Prediction

Long-Term predictions are linked to long-term investment which are considered investments held for more than 1 year and 1 day. A long-term investment is an account on the asset side of a company's balance sheet that represents the company's investments, including stocks, bonds, real estate and cash (Claire Boyte-White, 2021)

The long-term investment has an obvious and big difference from the short-term investment. As mentioned, short-term investments will be sold quickly, whereas long-term investments will be held for a longer period of time, sometimes could even never be sold. Long-term investors are willing to consider risks that come with it, just for an uncertain potential higher return. You must be patient for a longer period of time. It is advisable of the individual having enough capital to afford a quantity that will not be touched for a while as (Alexandra Twin, 2021) stated.

D. Regression Analysis

Regression Analysis (RA), is a generally useful methodology for expectation and forecasting, where its implementation has a generous cover within the machine learning field. Second, in certain circumstances regression analysis can be utilized to deduce causal connections between the dependent and independent factors. Critically, regressions on their own just uncover connections between a dependent variable and independent in a fixed dataset. To utilize regression for prediction or to derive causal connections, respectively, a researcher must carefully justify why existing relationships have predictive power for a new context or why a relationship between two variables has a causal interpretation (Nirbhey Singh, 2017). Some regression analysis methods are discussed in the following paragraphs.

Linear Regression (LR), was developed in the statistics field, but it is used in machine learning as well. Linear regression builds a model that assumes a linear relationship between the input variables (x) and the output variable (y) along with a corresponding coefficient for each input variable. Machine learning uses gradient descent to update these coefficients to reduce the error in forecasting the output variable. (Nirbhey Singh, 2017)

Polynomial Regression (PL), is a slightly more advanced regression algorithm than linear regression. It models the relationship between a dependent(y) and independent variable (x) as nth degree polynomial. If we apply a linear model on a linear dataset, then it provides us a good result as we have seen in Simple Linear Regression, but if we apply the same model without any modification on a non-linear dataset, then it will produce a drastic output. Due to which loss function will increase, the error rate will be high, and accuracy will be decreased. So for such cases, where data points are arranged in a non-linear fashion, we need the Polynomial Regression model (Nirbhey Singh, 2017)

Random Forest Regression (RF), is a supervised learning algorithm for classification and regression purposes that consists of a large number of decision trees that operate as an ensemble. An ensemble method combines the predictions from multiple decision tree algorithms together to make more accurate predictions than any individual model would. Random forest constructs a multitude of decision trees with a training set and outputs the class for a classification problem or prediction for a regression problem. (Katherine Gail Nowadly and Sohyun Jung, 2020)

E. Artificial Neural Network

Artificial Neural Network (ANN), is a machine learning algorithm that resembles the workings of the human brain. ANN discovers a new pattern out of investigating a relationship 42 between the inputs and outputs. ANN has three important elements in its structure: input layer, hidden layer, and output layer. The input layer receives the information into the model, the hidden layer computes the patterns among the input data, and the output layer obtains the result. In a neural network, there are multiple parameters and hyperparameters that affect the performance of the model. Each node in the network is assigned weights that can be interpreted as the impact that node has on the node of the next layer. Depending on the weighted sum value, an activation function defines whether a given node should be activated or not. Finally, based on the result, the model adjusts the weights of the neural network to optimize the following various cost minimization functions. (Murkute Amod, Tanuja Sarode, 2015). Some common types of artificial neural networks are discussed in the following paragraphs.

Feed Forward Neural Network (FFNN), the information only moves in one direction from the input layer, through the hidden layers, to the output layer. The information moves straight through the network and never touches a node twice. Feed-forward neural network has no memory of the input it receives and is bad at predicting what is coming next. Because a feed-forward network only considers the current input, it has no notion of order in time. It simply can not remember anything about what happened in the past except its training. (Niklas Donges, 2019)

Recurrent Neural Network (RNN), is a class of neural networks that is helpful in modelling sequence data. Derived from feedforward networks, RNN exhibits similar behaviour to how human brains function. Recurrent neural networks are of big use to produce predictive results in sequential data that other algorithms can not do (Murkute Amod, Tanuja Sarode, 2015).

Long Short-Term Memory Neural Network (LSTM Neural Network), is very powerful in sequence prediction problems because it is able to store past information. This is important because the previous price of a stock is crucial in predicting its future price. (D. M. Q. Nelson et. al, 2017)

F. Other Machine Learning Methods

Support Vector Machine (SVM), is a supervised learning algorithm that is used for classification. It is based on the structural risk minimization principle, which sets a hyperplane that maximizes the margin of separation between different classes and minimizes the expected error of a learning machine. Through training the data, one unique hyperplane, called optimize hyperplane, is created to separate the data. Since most real data is not linearly distributed, a kernel function is created to classify the non-linearly distributed data. The kernel function is applied on each data instance to map the original nonlinear instances into a high dimensional space, where they become separable. By calculating the distances from 41 the kernel function to the instances, SVM can be used as a regression method to predict the future values too. (Vapnik, 1998)

Distributed Decision Tree (DDT), is a machine learning method that finds patterns inside of data and makes a rule to classify said data. This method classifies instances into branch-like segments, hence its nomenclature. The decision tree structure has internal nodes representing a feature and branches representing a decision rule, with each leaf node representing the outcome. The decision tree algorithm starts

by selecting the best attribute using attribute selection measures to split the data. The attribute selected becomes a decision node and breaks the dataset into smaller subsets. By repeating this process recursively for each instance, a tree is formed. At each node of the decision tree, the information entropy gain generated by the split is used to evaluate whether the variable is meaningful or not (S. Patil et al, 2019)

Some people use a combination of different machine learning algorithms for stock prediction. In accordance with different approaches in order to try and combine the advantages that each of these algorithms can offer (Marcus Thorström, 2017), (Mehtab, S. and Sen, J, 2021)

III. Research Methodology

Two different machine learning methodologies for stock prediction from many possible on the market were selected. Those are: Linear Regression (LR) and Long short-term memory neural network (LSTM). Where each of them will be explained in more detail in the following subsections. Finally, a detailed discussion of those algorithms in terms of accuracy and efficiency is also presented.

A. Linear Regression

This algorithm was the first kernel to be used in a Support Vector Machine (SVM) and has no impact on the dimensionality of the data as opposed to the other kernels. The linear kernel is related to a linear equation. The definition of the linear kernel is: K(xi , xj) = x T i xj

In linear regression, a linear equation is fitted to a data set and minimizes the squared error between estimates and actual values. In this implementation, the ordinary least squares (OLS) solution is used where a matrix X is computed to give the best estimate. The form of the regression can be seen in (Equation 1), where the shape of $\boldsymbol{b}$ is a $\boldsymbol{1 \times n}$ matrix, the shape of $\boldsymbol{a}$ is a $\boldsymbol{n \times p}$ matrix and the $\boldsymbol{X}$ is a 1 × n matrix (Mehtab, S. 2021)

(Equation 1), $\boldsymbol{b = aX}$

The objective of ordinary least squares (OLS) is to minimize the euclidean distance of the estimation of X as can be seen in (Equation 2), where $||...||^2$ is the euclidean distance.

(Equation 2), $\boldsymbol{min(||b - aX||^2)}$

A very common equation used for linear regression is the $y = \beta_0 + \beta_1 x_1$, 'Fig 1' shows how the algorithm can perform in a graphical way based on data fitted into the algorithm.

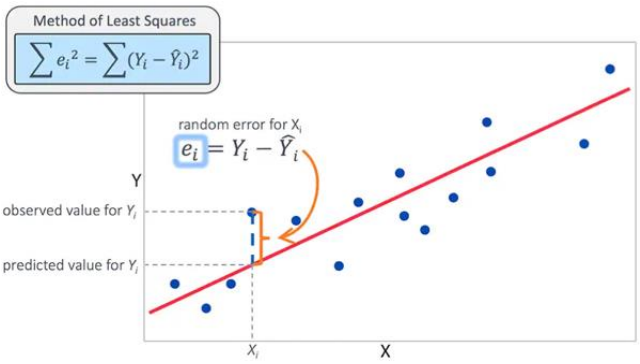

Fig. 1. Graph plot - Linear Regression

B. Long Short-Term Memory Neural Network

In a Long Short-Term Memory Neural Network (LSTM) the nodes are recurrent but they also have an internal state as working memory space which means information can be stored and retrieved over many time steps. Each LSTM cell unit as shown in 'Fig. 2.' maintains a cell memory. For each time step the next LSTM cell unit can choose to read from it, write to it or reset the cell using an explicit gating mechanism.

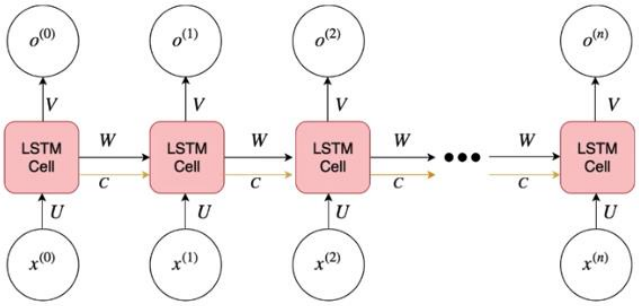

Fig. 2. Cell diagram - Long Short-Term Memory Neural Network

Each cell unit has three gates, the input gate which controls whether the memory cell is updated using the formula presented in the equation 3.6, the forget gate which controls if the memory cell is reset to 0 using the formula presented in the equation 3.7 and the output gate which controls whether the information of the current cell state is made visible using the formula presented in the equation 3.8

(Equation 3), $$i^{(t)} = \sigma\left(W^i\left[h^{(t-1)}, x^{(t)}\right] + b^i\right)$$

(Equation 4), $$f^{(t)} = \sigma\left(W^f\left[h^{(t-1)}, x^{(t)}\right] + b^f\right)$$

(Equation 5), $$o^{(t)} = \sigma\left(W^o\left[h^{(t-1)}, x^{(t)}\right] + b^o\right)$$

C. Research Method Attributes

Accuracy and efficiency are very critical performance indicators for stock price prediction, it is important for investors to get accurate predictions to know where to invest but also to get quick predictions as stocks is a very dynamic market. Is also key to consider these attributes in correlation with each other, complementing one another depending on investors preferences. In some cases efficiency may be more important than accuracy. If someone does not want to win too much but wants to win straight away, they are not interested so much in accuracy but they are interested in the efficiency of the predicted value. On the other hand others might want to win more regularly but not straight away leading more towards accuracy focus.

IV. Research Experiment

This section introduces the requirements for the research experiment with the chosen machine learning algorithms. Each of these algorithms should be simulated using three independent modules which are discussed in this section. These modules should interact in a way whereby each module performs appropriately for the achievement of specific tasks for the algorithms. The first module deals with the selection of the input features for the machine learning algorithms. The second module focuses on the processing of the features by

these algorithms. Finally, the third module deals with the presentation of the modelling results from the algorithms.

A. *Input Module Specification*

This module should allow investors to select input variables such as number of days in the past with actual stock values and number of training iterations for the chosen machine learning algorithms. These input variables are important as they correlate directly with the actual stock values for the next day in the past. The upper limit of the variation ranges for these input variables should be 100 for the number of days in the past and 20 for the number of iterations. These upper limits are higher than the short term prediction works reviewed in section 2. The purpose of this is to explore the impact on the computational accuracy and efficiency of the chosen machine learning algorithms.

B. *Processing Module Specification*

This module should process the input variables from the previous module in section 4.2.1 in order to mapped them into output variables. These output variables are the actual and predicted stock values for the next day in the past as well as the training and testing errors for the chosen machine learning algorithms. Each machine learning algorithm should be executed simultaneously during a single run to facilitate the analytical calculation of the output variables. The purpose is to simulate the computational accuracy and efficiency of these algorithms.

C. *Output Module Specification*

This module should map the analytical values of the output variables from the previous module in section 4.2.2 into a graphical presentation. This presentation should be done simultaneously for the chosen machine learning algorithms in regards to the actual and predicted stock values for the next day in the past as well as the training and testing errors for these algorithms. The results for each machine learning algorithm should be visualised simultaneously during a single run to enable investors to make better investment decisions. These decisions are based on the evaluation of the computational accuracy and efficiency of these algorithms.

V. Research Design

The chosen machine learning algorithms are explained in terms of their specific design in order to simulate these algorithms within the software. Those designs are presented in a flowchart format which shows a general design type of level in terms of block segments. Each block segment has a task assigned that processes information from the previous block. The behaviour of each algorithm depends on the first block segment which relies on the user input in order to behave in a certain way. The following subsection presents those designs for the machine learning algorithms chosen, Linear Regression (LR) and Long Short-Term Memory.

The design for the Linear Regression (LR) algorithm is described in the flowchart diagram in 'Fig. 3.'. Each block segment presented in this diagram correlates to one another as an accurate implementation of the algorithm behaviour in the software simulation. There are three different methods that can be implemented when using the Linear Regression (LR) model, Bisquare, Least Square or Least Absolute Residual method; this will depend upon the experimental data points used that differs significantly from the rest of the data points on the graph. In the development of this software, using Linear Regression algorithm for stock prediction, plenty of data in the past is required, therefore the Least Square method for the linear fit is used to find the slope and intercept by minimizing residue for even closer prediction results. The weight referred to in the diagram is the array for (X, Y) observations. The general formula for Linear Regression can be found and explained in section 3.3.1

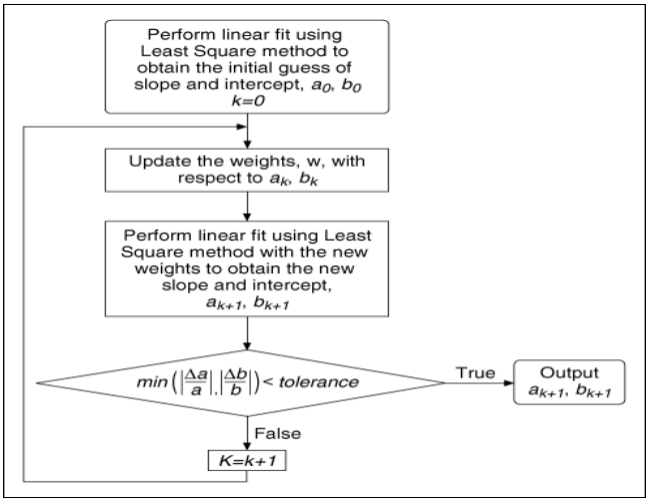

Fig. 3. Flowchart diagram - Linear Regression

A. *Long Short-Term Memory Neural Network*

The design for the Long Short-Term Memory Neural Network (LSTM) algorithm is described on the flowchart diagram in figure 5.4. The block segment presented in this diagram is an accurate implementation of the algorithm behaviour in a software simulation. The decisions made by this model are based on its three different inputs: the current input, previous output and previous memory shown in the figure 5.4 legend. These inputs are all used in the node calculations, the result of the calculations are used not only to provide an output but also update the memory. The gate parameters control the flow of information within the node, in particular how much the safe state information is used as an input to the calculation. These gate parameters are weights and biases which means the behaviour depends on the inputs

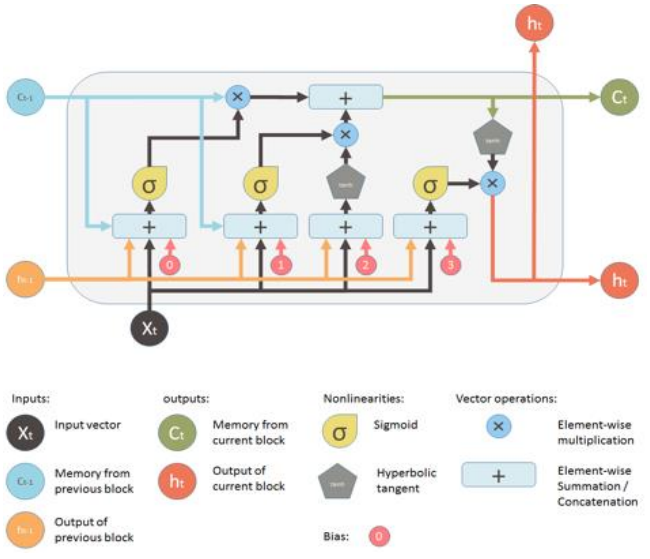

Fig. 4. Flowchart diagram – Long Short-Term Memory Neural Network

VI. Research Results

This section presents screenshots of the results for each of the machine learning algorithms chosen in terms of actual and predicted stock values for different combinations of numbers of training iterations and days in the past as well as a table with partial data points for each algorithm. Sections *A* and *B* present the results for Linear Regression (LR) and Long Short-Term Memory Neural Networks (LSTM), respectively. The benchmark approach, based on existing research, is implemented with 10 training iterations and 50 days in the past and the novel approach with 20 training iterations and 100 days in the past.

A. Linear Regression

This algorithm works very well for all scenarios based on the closeness of the two curves on the corresponding screenshots that represent actual and predicted stock prices. The rough visual observation of the screenshots suggests that the variation of the number of training iterations and days in the past does not have a substantial impact on the very good prediction accuracy of the algorithm. The general conclusion from the above observation is confirmed analytically on the corresponding tables for the first five days from the dataset used. The orange curve on the graph represents the prediction values and the blue curve shows the actual values from the dataset even though this curve is not very visible.

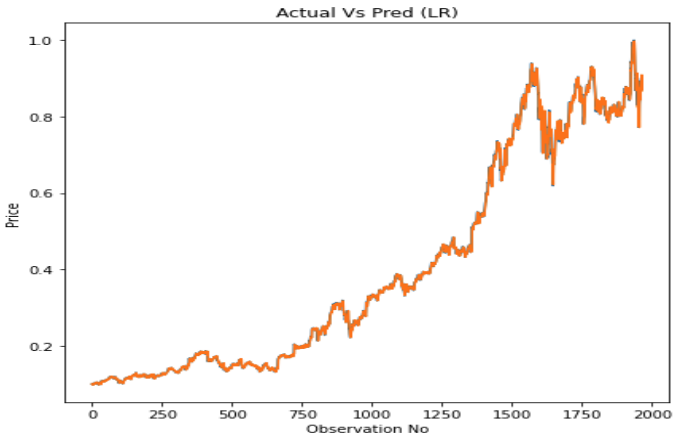

Fig. 5. Benchmark approach - Actual price vs Linear Regression Prediction

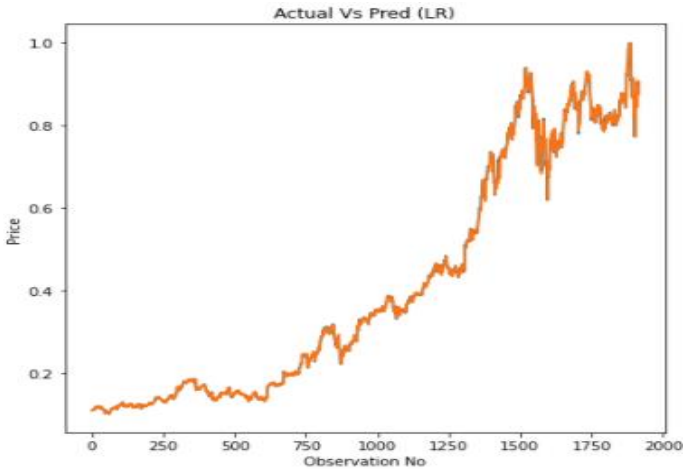

Fig. 6. Novel approach - Actual price vs Linear Regression Prediction

B. Long Short-Term Memory Neural Network

This algorithm works well for all scenarios based on the closeness of the two curves on the corresponding screenshots that represent actual and predicted stock prices. The rough visual observation of the screenshots suggests that the variation of the number of training iterations and days in the past do not have a substantial impact on the good prediction accuracy of the algorithm. The general conclusion from the above observation is confirmed analytically on the corresponding tables for the first five days from the dataset used. The orange curve on the graph represents the prediction values and the blue curve shows the actual values from the dataset.

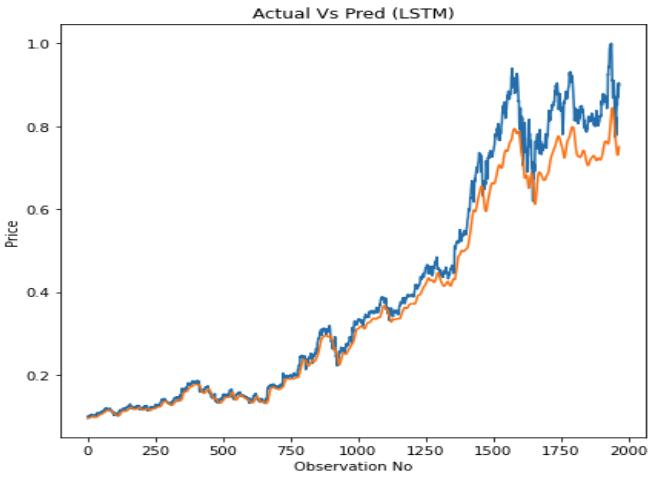

Fig. 7. Benchmark approach – Actual price vs Long Short-Term Memory Neural Network Prediction

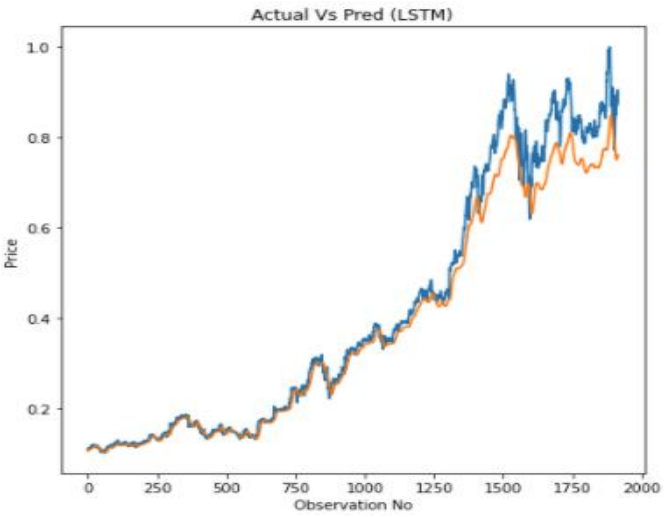

Fig. 8. Novel approach - Actual price vs Long Short-Term Memory Neural Network Prediction

VII. Research Evaluation

This section presents graphical results in terms of Root Mean Square Error (RMSE) using the dataset introduced in section 6.1. This dataset is split into training and testing sets whereby training is used to fit the machine learning algorithms to the training set and testing is used to validate the algorithms with the testing set. Representation of the

training and testing graph curves is shown in section 7.2.1 for every simulation run for each algorithm. Comparative evaluation of these training and testing sets results in terms of accuracy and efficiency is presented in section 7.2.2 in terms of the proposed novel approaches and the benchmark approach for the machine learning algorithms.

This section shows graphically Root Mean Square Error (RMSE) for every simulation run for each algorithm, first on individual graphs and then integrated together on a single graph. A precise percentage for each set is used. For training the software uses 65% of the data to fit the algorithms and 35% of the data is used to test the algorithms. Each algorithm is described with four graphs, one for the benchmark approach and three for the novel approaches. According to the graphs, the orange curve refers to the training error while the green curve refers to the testing error. The actual values of the datasets used are presented in a blue curve, whose visibility varies in some graphs.

A. Linear Regression

This algorithm works very well for all scenarios based on the variation range (0,1) of the two parts of the curves on the corresponding screenshots that represent training and testing Root Mean Square Error (RMSE) for stock prices. The visual observation of the screenshots confirms the conclusion from the rough observation in section 6.2.1 that the number of training iterations and days in the past does not have a substantial impact on the very good prediction accuracy of the algorithm.

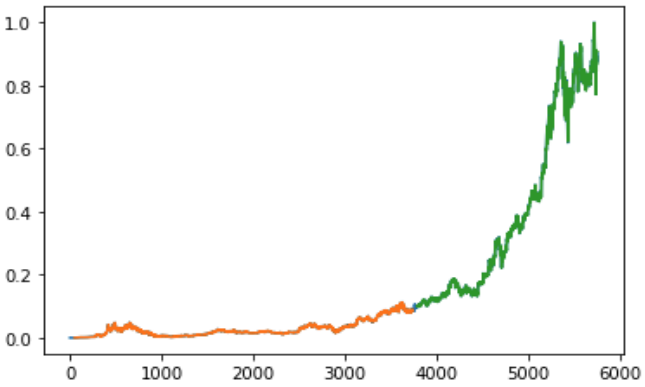

Fig. 9. Benchmark approach – Training and Testing for Linear Regression

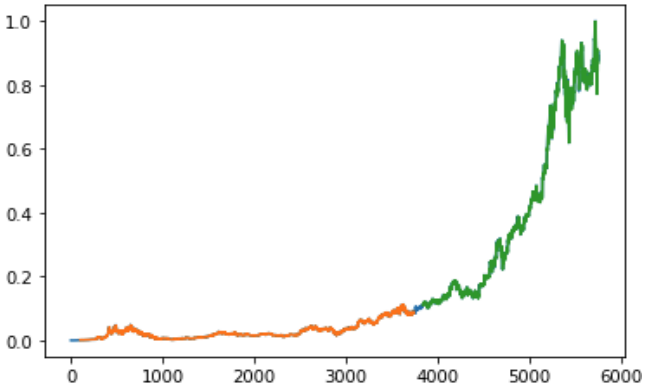

Fig. 10. Novel approach – Training and Testing for Linear Regression

B. Long Short-Term Memory Neural Network

This algorithm works well for all scenarios based on the variation range (0,1) of the two parts of the curves on the corresponding screenshots that represent training and testing Root Mean Square Error (RMSE) for stock prices. The visual observation of the screenshots confirms the conclusion from the rough observation in section 6.2.4 that the number of training iterations and days in the past does not have a substantial impact on the good prediction accuracy of the algorithm.

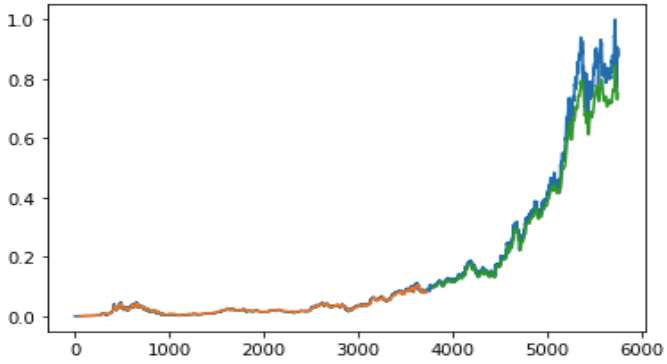

Fig. 11. Benchmark approach - Training and Testing for Long Short-Term Memory Neural Network

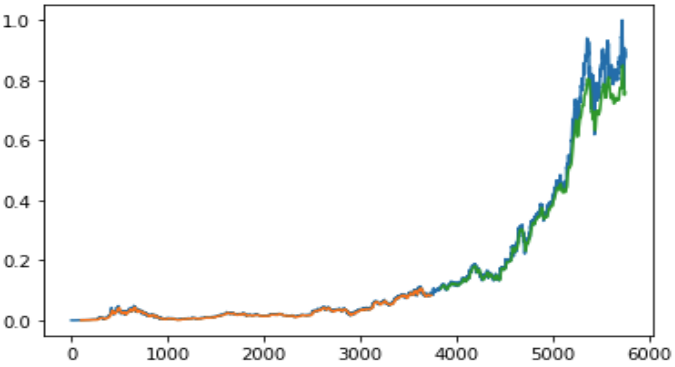

Fig. 12. Novel approach - Training and Testing for Long Short-Term Memory Neural Network

C. Algortihms Comparison

This section presents three tables of 3x3 with Root Mean Square Error result evaluations and time taken performed by the machine learning for the different runs. These tables are presented in terms of training error, testing error and computational time. The best algorithm performance for training is Polynomial Regression (PR) and for testing is Linear Regression (LR) as they both maintained a great overall accuracy in their results. The worst algorithm performance for training is LSTM Neural Network and for testing is Polynomial Regression (PR) as they did not maintain a great overall accuracy in their results, these results are compared based on the benchmark run 1 with run 2. However, LSTM Neural Network is the only algorithm which made an improvement from run 1 up to run 4 for training and testing results. The best efficiency in terms of computational time was achieved in run 3.

a) Training: Root Mean Square Error (RMSE) for training is a way of representing a performance evaluation metric for accuracy during the learning process. The figures shown in table 7.1 present two clear observations. The first observation, which indicates run 1 for the benchmark approach, has the same values as run 3, similarly for run 2 and run 4 except for the LSTM Neural Network algorithm. Second observation, indicates how the accuracy changes with the number of days in the past for each algorithm between run 1 and run 2. Algorithms such as Linear Regression (LR) and Polynomial Regression (PR) increased in their accuracy results in run 2 compared to run 1, whilst Random Forest (RF) and LSTM Neural Network decreased in their accuracy results in run 2 compared to run 1. However, LSTM Neural Network increased its overall accuracy results in run 4 compared to runs 1, 2 and 3. In conclusion, based on these observations, the number of training iterations does not change the algorithms accuracy in most runs, however the increase of the number of days in the past changes the accuracy for all of the algorithms. For clarity, table 7.1 shows the best results for each algorithm highlighted in bold.

Trainning: Root Mean Square Error (RMSE) values		
Simulation scenarios / Machine Learning Algorithms	Run 1: Benchmark 10 number of trainning iterations 50 days in the past	Run 2: Novel 20 number of trainning iterations 100 days in the past
Linear Regression (LR)	0.0011333929498220	**0.0011254118145283**
LSTM Neural Network	0.0025748614649406	**0.0024926140940913**

Fig. 13. Training error values

b) Testing: Root Mean Squared Error (RMSE) for testing is a way of representing a performance evaluation metric for accuracy during the validation process. The figures shown in table 7.2 present two clear observations. The first observation, indicates run 1 for the benchmark approach, having the same values as run 3, similarly for run 2 and run 4 except for the LSTM Neural Network algorithm. Second observation, indicates how the accuracy changes with the number of days in the past for each algorithm between run 1 and run 2. Algorithms such as Linear Regression (LR), Polynomial Regression (PR) and Random Forest (RF) decreased in their accuracy results in run 2 compared to run 1, however LSTM Neural Network increased its accuracy results in runs 1, 2 and 4. In conclusion, based on these observations, the number of training iterations does not change the algorithms accuracy in most runs, however the increase of the number of days in the past changes the accuracy for all of the algorithms. For clarity, table 7.2 shows the best results for each algorithm highlighted in bold.

Testing: Root Mean Square Error (RMSE) values		
Simulation scenarios / Machine Learning Algorithms	Run 1: Benchmark 10 number of trainning iterations 50 days in the past	Run 2: Novel 20 number of trainning iterations 100 days in the past
Linear Regression (LR)	**0.0094332258511968**	0.009686760360715
LSTM Neural Network	0.0557649141216961	**0.033478711736450**

Fig. 14. Testing error values

c) Computational Time: Computational time Is a specific way of representing a performance evaluation metric for efficiency. The figures shown in table 7.3 are approximate times for the simulation runs performed by the software. The main observation from this table indicates how the efficiency changes with the number of days in the past between run 1 to run 2 and run 3 to run 4. In conclusion, based on this observation, the increase in the number of days in the past has a bigger impact in the time efficiency than the number of training iterations. For each run of the 4 algorithms the computational time is divided by 4 to obtain a raw estimate of computational time for each individual algorithm. The computational time in this table includes both training and testing times. These times can not be measured separately because the software gives both results for every algorithm simultaneously. For clarity, table 7.3 shows the best results for each algorithm highlighted in bold.

Computational time values (measured in minutes)		
Simulation scenarios / Machine Learning Algorithms	Run 1: Benchmark 10 number of trainning iterations 50 days in the past	Run 2: Novel 20 number of trainning iterations 100 days in the past
Linear Regression (LR)	1:42.63 minutes	4:27.32 minutes
LSTM Neural Network	1:42.63 minutes	4:27.32 minutes

Fig. 15. Computational time values

VIII. Conclusion

A. Achievements

The aim of this research focused engineering project has been successfully achieved. In particular, this paper has systematically explored the suitability and applicability of several popular machine learning algorithms for financial modelling of the stock market by means of a research focused engineering project. This aim has been fully met through the achievement of all planned objectives.

Objective 1, to evaluate accuracy and efficiency of stock market prediction in a complementary way, has been achieved by exploring the relationship between these two performance indicators, as shown in section 7.2.2.

Objective 2, to visualise results from algorithm training and testing in an integrated way, has been achieved by presenting the Root Mean Square Error for these two stages in the same graph, as shown in section 7.2.1

Objective 3, to execute several machine learning algorithms in a simultaneous way, has been achieved by running these algorithms at the same time and showing their results together, as shown in sections 6.2.1-6.2.4 and section 7.2.1.

B. Contributions

This research focused engineering project has made several contributions in terms of academic novelty and performance evaluation. Each of the four machine learning algorithms considered for this project has been evaluated comparatively. In particular, each algorithm has been simulated and tested for four different simulation runs whereby one of these runs is a benchmark approach and the other three runs are novel approaches. In this context, the benchmark approach uses a low number of training iterations and days in the past in line with most other works in the field

of short term prediction for stock prices. In contrast, the novel approaches use a higher number of training iterations and/or days in the past in order to take advantage of the ability of machine learning algorithms to perform better by using longer learning and larger datasets. Finally, the prediction accuracy of these machine learning algorithms has been evaluated in relation to their computational efficiency in order to provide more flexibility to individual investors depending on their preferences.

There are a number of future studies that can be undertaken for this research focus project which would be beneficial for anyone that wants to expand this project.

1. To evaluate more days in the past from different companies datasets for a performance comparison.
2. To run the software with more than 2 different numbers of training iteration for an even closer accuracy evaluation.
3. To add more company data sets from the NASDAQ stock market and compare the results of each company.
4. To combine datasets from different stock exchanges.
5. To use different machine learning algorithms for stock prediction such as Support Vector Regression (SVR), Support Vector Machines (SVM) and Back Propagation Neural Network (BPNN)
6. To use different financial indicators such as currency exchange rate.

C. *Wider Context*

The use of machine learning algorithms is very useful and has a lot of potential in the financial sector, predicting more general financial indicators, not just for individual investors but also for general economic business. It is also widely used in multiple other fields. Healthcare sector for medical diagnosis and disease. Agriculture sector to help farmers increase the productivity of new weeds and monitoring soil. Transportation sector to manage the traffic circulation and implementation of self-driving vehicles. Those are just a few of the common sectors where machine learning has added a significant value in their productivity.

ACKNOWLEDGMENTS

The first author would like to thank Elvira Yaghmour for the overall support and help as well as Leslie Botha and Heidi Botha for the moral support and guidance, during the work on this research paper. A special mention to Mohammed Fakruddin for the mentoring and guidance provided for the completion of this research.

REFERENCES

[1] Abran, A., Khelifi, A., Suryn, W., & Seffah, A. (2003). Usability meanings and interpretations in ISO standards. *Software quality journal*, *11*(4), 325-338.

[2] Arévalo, A., Niño, J., Hernández, G., & Sandoval, J. (2016). High-Frequency Trading Strategy Based on Deep Neural Networks. Intelligent Computing Methodologies Lecture Notes in Computer Science, 424436. doi:10.1007/978-3-319-42297-8_40

[3] Arora, A., et al.: Deep Learning with H2O (2015) learning process 20. Zeiler, M.D.: ADADELTA: An Adaptive Learning Rate Method, 6 (2012)

[4] Aumann, D. (2019, September 9). *R-stock Prediction*. github. https://github.com/daumann/r-stockPrediction

[5] Bollen, J., & Mao, H. (2011). Twitter Mood as a Stock Market Predictor. Computer,44(10), 91-94. doi:10.1109/mc.2011.323

[6] Chuqing Zhang, Han Zhang, Xiaoting Hu. (2019) A Contrastive Study of Machine Learning on Funding Evaluation Prediction. *IEEE Access* 7, pages 106307-106315

[7] Claire Boyte-White (2021, April). "Understanding Long-term vs ShortTerm Capital Gains", Retrieved from: https://www.investopedia.com/articles/personal-finance/101515/comparing -longterm-vs-shortterm-capital-gain-tax-rates.asp

[8] Coleman, D., Ash, D., Lowther, B., & Oman, P. (1994). Using metrics to evaluate software system maintainability. *Computer*, *27*(8), 44-49.

[9] Ding, X., Zhang, Y., Liu, T., & Duan, J. (2014). Using Structured Events to Predict Stock Price Movement: An Empirical Investigation. Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP). doi:10.3115/v1/d14-1148

[10] Horne, J. C., & Parker, G. G. (1967). The Random Walk Theory: An Empirical Test. Financial Analysts Journal, 23(6), 87-92. doi:10.2469/faj.v23.n6.87

[11] Huseyin Ince & Theodore B. Trafalis (2008) Short term forecasting with support vector machines and application to stock price prediction, International Journal of General Systems, 37:6, 677-687, DOI: 10.1080/03081070601068595

[12] Jingyi Shen, M. Omair Shafiq. (2020) Short-term stock market price trend prediction using a comprehensive deep learning system. *Journal of Big Data* 7:1

[13] Kaustubh Khare, Omkar Darekar, Prafull Gupta and V. Z. Attar, "Short term stock price prediction using deep learning", *Recent Trends in Electronics Information & Communication Technology (RTEICT) 2017 2nd IEEE International Conference on*, pp. 482-486, 2017.

[14] Li, X., Huang, X., Deng, X., & Zhu, S. (2014). Enhancing quantitative intra-day stock return prediction by integrating both market news and stock prices information. Neurocomputing, 142, 228-238. doi:10.1016/j.neucom.2014.04.043

[15] Marcus Thorström (2017). Applying machine learning to key performance indicators. Chalmers University of Technology and University of Gothenburg. 4, 25-36

[16] Mark L. Mitchell and J. Harold Mulherin The Journal of Finance Vol. 49, No. 3, Papers and Proceedings Fifty-Fourth Annual Meeting of the American Finance Association, Boston, Massachusetts, January 3-5, 1994 (Jul., 1994), pp. 923-950

[17] Matharu, G. S., Mishra, A., Singh, H. and Upadhyay, P., 2015. Empirical study of agile software development methodologies: A comparative analysis. ACM SIGSOFT Software Engineering Notes, 40 (1), 1–6.

[18] Mehtab, S. and Sen, J. (2021) "A time series analysis-based stock price prediction using machine learning and deep learning models", International Journal of Business Forecasting and Marketing Intelligence (IJBFMI), Inderscience Publishers. (In Press)

[19] Murkute Amod, Tanuja Sarode, "Forecasting market price of stock using artificial neural networks", International Journal of Computer Applications, 124 (12) (2015), pp. 11-15

[20] Nelson, D. M. Q., A. C. M. Pereira and R. A. de Oliveira, "Stock market's price movement prediction with LSTM neural networks," 2017 International Joint Conference on Neural Networks (IJCNN), 2017, pp. 1419-1426, doi: 10.1109/IJCNN.2017.7966019.

[21] Nirbhey, S., Neeha K., Deepali V., Vidhi S (April 2017). Stock Prediction using Machine Learning a Review Paper. Journal of Computer Applications (0975 – 8887) Volume 163 –

[22] Patil, S and Kulkarni, U. "Accuracy Prediction for Distributed Decision Tree using Machine Learning approach," 2019 3rd International Conference on Trends in Electronics and Informatics (ICOEI), 2019, pp. 1365-1371, doi: 10.1109/ICOEI.2019.8862580.

[23] Royce, W. W. (1987, March). Managing the development of large software systems: concepts and techniques. In *Proceedings of the 9th international conference on Software Engineering* (pp. 328-338).

[24] Saqib Aziz, Michael M. Dowling, Helmi Hammami, Anke Piepenbrink. (2019) Machine Learning in Finance: A Topic Modeling Approach. *SSRN Electronic Journal*.

[25] Schumaker, R. P., & Chen, H. (2009). A quantitative stock prediction system based on financial news. Information Processing & Management, 45(5), 571- 583. doi:10.1016/j.ipm.2009.05.001

[26] Singh, Aishwarya. (26 July 2019) “Predicting the Stock Market Using Machine Learning and Deep Learning.” Analytics Vidhya, article. www.analyticsvidhya.com/blog/2018/10/predicting-stock-price-machine-le arningnd-deep-learningtechniques-python/.

[27] Vijh, M., Chandola, D., Tikkiwal, V.A., Kumar, A.Hide details, Stock Closing Price Prediction using Machine Learning Techniques, Procedia Computer Science, Volume 167, 2020

[28] Weng, B., Ahmed, M. A., & Megahed, F. M. (2017). Stock market one-day ahead movement prediction using disparate data sources. Expert Systems with Applications,79, 153-163. doi:10.1016/j.eswa.2017.02.041