Stock Price Prediction System Based on Hybrid RNN

Hope Soko, Dr. T.Samraj Lawrence [a, *] and Dr.G.Glorindal [b]

a DMI-St. John The Baptist University P.O. BOX 406, Mangochi, hopesoko8@gmail.com, +265-998629057
b DMI-St. Eugene University, Zambia pg@dmiseu.edu.zm, +265- 99446970

ARTICLE INFO

Article history:
Received
Received in revised form
Accepted

Available online

Keywords:
RNN
Prediction
Complex
Price
Stock

ABSTRACT

The stock market provides grounds for forecasting due to the complex nature of financial data available in the marketplace and an endless supply of external factors that affect the price of stocks. The complex nature of the marketplace makes it difficult to create trends that take into account all the factors that will affect the cost of stock for a given day, let alone the trendline for the stock price as to whether the price will take an upward or downward inflection. There exists a vast amount of historical data regarding trends of all stocks that are being traded daily. However, making rapid decisions and predictions on stocks based on this enormous amount of pre-existing information can prove almost impossible. Thus, algorithms must be implemented to combat the available data's complexity. Recurrent neural networks have proved a successful means by which stock prices can be predicted. But has failed in accuracy. This paper aims to establish a hybrid RNN using the gated recurrent unit with stochastic gradient descent. This approach will provide high levels of accuracy and tackle the problem of complexity of the voluminous amounts of information in our marketplace.

Introduction

Stock prices are volatile in nature as they are heavily dependent on several external factors as such makes them almost impossible to predict. The process of predicting stock is likened to trying to predict a random walk in the park which means one should take in account all the turns and stops an individual is to make along the way or even to say an individual may choose not to take the walk altogether.

Stock markets offer an opportunity for traders to make a large amount of money with little investment. In that sense they mitigate the risk that is there is need to make for high risk of reward from high risk of investment. However, due to the large

number of factors that are affecting the stock market there is high chance that the stock prices can either fall or rise without much notice. As such an individual must take into account all factors when dealing with stocks. But the idea that one individual trader can keep tabs on all factors in regard to a

single trade is far-fetched and cannot be done. Even with the presence of the vast amounts of historical data available for use by the traders it does not make his or her job easier.

The process of the stock market trading is to predict the stock price such that it can be sold before its value decreases so that the traders does not make loses on his initial investment. As well as buy stock before its value increases so that one can easily make profit from the initial investment. This boils down the process of trading in a nutshell without looking deeply into all the step-by-step protocols that go into acquisition of stock. This being the case, it is clear to see that trading offers a risk that need to be mitigated as a lot of money is on the line.

Normally traders that have been trading for a long time make use of their expertise knowledge and years of experience to make profitable trades. However, this is not always enough to ensure the accuracy of trades in the long run. Prediction tools are used in these scenarios to make sure that the bulk of the trades that are made are profitable or at least don't result in loses. The hybrid Recurrent neural network offers one of such of these as it ensures that traders have a greater rate of accuracy when it comes to knowing the trends and prices of future stock prices.

An artificial neural network that employs sequential or time series data is known as a recurrent neural network (RNN). Wellknown programs like Siri, voice search, and Google Translate include these deep learning algorithms. They are often employed for ordinal or temporal issues, such as language translation, natural language processing (nlp), speech recognition, and picture captioning. Recurrent neural networks (RNNs) use training data to learn, much as feedforward and convolutional neural networks (CNNs) do. They stand out because of their "memory," which allows them to affect the current input and output using data from previous inputs. Recurrent neural networks' outputs rely on the previous parts in the sequence, unlike typical deep neural networks, which presume that inputs and outputs are independent of one another. Unidirectional recurrent neural networks cannot account for future occurrences in their forecasts, although they would be useful in deciding the output of a particular sequence (Education).

Over the years a series of prediction methods have been utilized in order to forecast the nature of stocks namely statistical and Artificial intelligent methods. However, the time series method technique offers one of the most widely utilized prediction techniques in most real time applications. It utilizes continuous data of a period of time to predict the result of the next period of time. It has been replaced by the recurrent neural networks and long short-term memory due to their level of accuracy and ability to model the complexity of the financial system.

1. Literature

In this section, relevant works are discussed. We evaluated the pertinent studies of two distinct domains: technical and financial.

According to Weng B, Lu L, Wang X, Megahed FM, Martinez (2018). They used ensemble techniques and internet data sources to forecast short-term stock values. The main contribution of this work is the development of an R-based platform for investors that do not need users to submit their data but instead uses an API to download the data from an internet source simply. From a research standpoint, they only considered the future price forecast for 1 to 10 days and did not consider durations longer than two trading weeks or less than one day. Their study's main flaw was that they only looked at 20 stocks with U.S. origins; as a result, the model may not apply to foreign stock markets and has to be further validated to rule out overfitting issues. They employed the neural network regression ensemble (NNRE), a Random Forest with unpruned regression trees as base learners (RFR), and an AdaBoost with unpruned regression trees as base learners (BRT), and a support vector regression ensemble as their machine learning models (SVRE).

According to Long W, Lu Z, Cui developed features based on deep learning for predicting stock price movement(2018). The authors' use of a unique hybrid model created by combining several types of neural networks to start it gives this study its power. It serves as an example of creating hybrid neural network architectures. The degree of accuracy is not stated. stochastic gradient descent in a multi-filter neural network (MFNN) (SGD) The originality of the used model.

The hybrid financial trading support system Thakur M. and Kumar D.A. developed uses the random forest and multicategory classifiers(2018). The authors suggested a useful model made for actual investing activity. They omitted to mention how long and computationally hard their efforts were. However, the fundamental problem with their job was a lack of foundation in the financial industry. The authors suggested a hybrid model that merged random forest (RF) techniques with a time- and computational-intensive weighted multicategory generalized eigenvalue support vector machine (WMGEPSVM).

Based on an initial set of attribute reductions, Lei L created a Wavelet neural network prediction technique of stock price trends(2018). The model was evaluated based on several stock market indexes, and the results were compelling about generality; nevertheless, if the same model were to be applied to

a particular stock, its performance may change. computational complexity of Rough Set (RS), Wavelet Neural
Network (WNN).

Pimenta A, Nametala CAL, Guimares FG, and Carrano EG created a multi-objective genetic programming-based automated investment system for the stock market (2018). The assessment section of this study was innovative. When doing validation, they considered a historical era that was a pivotal point in Brazilian politics and economy. This strategy strengthened the generalizability of their suggested model. They also established criteria to guarantee greater asset liquidity while choosing the sub-dataset for assessment. The comparison's starting point was too simple and fundamental, and the authors did not attempt to compare their findings to those of other models already in use. The paradigm and
technique of genetic programming (GP)

Cont R and Sirignano J created Perspectives from deep learning on universal aspects of price generation in financial markets(2018). Due to the implicit programming of the deep learning technique, it is uncertain whether any worthless characteristics are tainted while feeding the data into the model trained on a universal feature set of financial markets. The dataset utilized contained purchase and sell records of all transactions.

Fischer T. and Krauss C. created deep learning with extended short-term memory networks for financial market forecasts(2018). This work's main strength is that the authors performed predictions using the most recent deep learning technology. Due to their ignorance of the financial industry, they relied on the LSTM approach. Although the LSTM performed better than the conventional DNN and logistic regression techniques, the author neglected to disclose the work required to train an LSTM with long-term dependencies. Deep learning approach LSTM.

S. McNally, J. Roche, and S. Caton use machine learning to predict bitcoin prices (2018). This work's feature engineering and optimization sections are the most useful since we could use the same techniques to pre-process our data. Overfitting is their work's fundamental flaw. The research challenge of forecasting the Bitcoin price trend parallels that of predicting the price of stocks. Threats to this effort include hidden characteristics and sounds in the pricing data.The Boruta algorithm-optimized RNN and LSTM model's prediction of the Bitcoin price trend show some parallels to stock market pricing.

Using Sirignano J and Cont Deep learning insights on the common characteristics of price creation in financial markets(2018). Other than the stocks in the training data, their universal model could generalize and cover them. The price of the training was steep.

Due to the LSTM deep learning algorithm's implicit programming, it features stochastic gradient descent (SGD) LSTM units and, in the end, a feed-forward layer with rectified linear units (ReLUs) (SGD).

Using M. Omair Shafiq, Jingyi Shen Built a prediction model of the target stocks utilizing a comprehensive deep learning system to anticipate short-term stock market trends(2020). Described data sets lack FE, RFE, and PCA period.

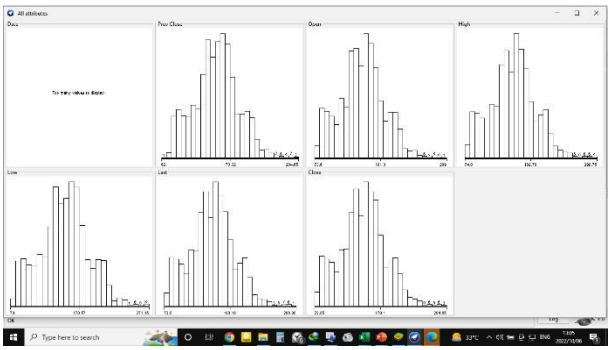
2. Problem Statement

A stock exchange market depicts savings and investments that are advantageous to increase the effectiveness of national economic. The future stock returns have some predictive relationships with the publicly available information of present and historical stock market indices. The investors decide the better time to sell/buy/hold a share in stock market based on the former relationship. Every investor is interested in predicting the future stock prices, whether the investor may be a long-term investor or a day-trader. This possesses a major challenge to design and develop an effective and efficient predictive model that assists the investors to take appropriate decisions.

3. Dataset Structure and Description

In this section, the details of the data set being used for the research are described. As already stated, we have taken a bulk of the data from yahoo finance, and the data from the site is broken down into the date of the stock opening price, highest price, lowest price, closing price, adjusted closing price, volume of the stock for that particular day. And we will note from the data that stock trading does not happen daily.

However, we only require the date and the closing stock price for that particular day. Splitting the data into only the fields needed for our supervised learning problem will be necessary. We apply a function that only considers the date and closing stock price. Upon having this information, we have enough information to model our supervised learning model.



Previous close, opening value and high value

4. Methodology.

In this section we describe the methodology used and how the data set will be used in accordance our algorithm in question.

Researchers have been using a variety of models for predicting stock price trends, but how do we decide which is the best? In this work, we will compare our approach with the outperformed machine learning models in the evaluation part and find the solution for this research question.

First and foremost, we have to grab our required data set from finance yahoo where we can acquire a consolidated list of data sets that have been collected over the years. The website has a wide array of financial history that spans over 30 years for several financial years. Our intended company details will span for a period of about 20 years. What is crucial is to import our data set as a csv file and import it as a panda file. The next step is to set the required parameters for our data set namely we will be mainly concentrating on the date, opening stock price, highest peak of the stock for that day, the low point the stock for that day, adjusted price for that day and the closing price for that particular day. However, our hybrid RNN aims to look at only two parameters from the data set which means we have to so rightly discard the other parameters. This means we have to only include the date of the transaction and the closing stock price of that particular day.

Supervised Learning Approach

In order to correctly solve the problem of prediction using the RNN we need to convert this problem into a supervised learning problem which is essentially "the training process for artificial intelligence (AI) algorithms is designed to be largely automated innately. There are often thousands, millions, or even billions of data points, and the algorithms must process them to search for patterns. In some cases, though, AI scientists are finding that the algorithms can be more accurate and efficient if humans are consulted occasionally during the training.

The result creates hybrid intelligence that marries the relentless, indefatigable power of machine learning (ML) with human intelligence's insightful, contextsensitive abilities. The computer algorithm can plough through endless training data files, and humans correct the course or guide the processing. To a large extent, supervised ML is for domains where automated machine learning does not perform well. Scientists add supervision to bring the performance up to an acceptable level. It is also an essential part of solving problems where no readily available training data contains all the details that must be learned. Many supervised ML problems begin with gathering a team of people who will label or score the data elements with the desired answer. For example, some scientists built a collection of images of human faces and then asked other humans to classify each face with a word like "happy" or "sad." These training labels made it possible for an ML algorithm to start to understand the emotions conveyed by human facial expressions." (Wayner)

We need to do this by establishing target values that need to be used as input for our model. This will allow us to treat the problem as a normal regression problem.

We convert all the values we have of the date and closing stock and convert them into arrays for our tensor flow model. "The training process for artificial intelligence (AI) algorithms is designed to be largely automated innately. There are often thousands, millions, or even billions of data points, and the algorithms must process them to search for patterns. In some cases, though, AI scientists are finding that the algorithms can be more accurate and efficient if humans are consulted occasionally during the training. The result creates hybrid intelligence that marries the relentless, unwavering power of machine learning (ML) with human intelligence's insightful, context-sensitive abilities. The computer algorithm can plough through endless training data files, and humans correct the course or guide the processing.

To a large extent, supervised ML is for domains where automated machine learning does not perform well. Scientists add supervision to bring the performance. "TensorFlow is an open-source end-to-end platform for creating Machine Learning applications. It is a symbolic math library that uses dataflow and differentiable programming to perform various tasks focused on the training and inference deep neural networks. It allows developers to create machine learning applications using various tools, libraries, and community resources. Currently, the world's most famous deep learning library is Google's TensorFlow. Google product uses machine

learning in all of its products to improve the search engine, translation, image captioning, or recommendations. In this TensorFlow tutorial, you will learn:

To give a concrete example, Google users can experience a faster and more refined search experience with AI. If the user types a keyword in the search bar, Google provides a recommendation about what could be the next word. Google wants to use machine learning to take advantage of its massive datasets to give users the best experience. Three groups use machine learning:

They can all use the same toolset to collaborate and improve efficiency. Google does not just have any data; they have the world's most massive computer, so Tensor Flow was built to scale. TensorFlow is a library developed by the Google Brain Team to accelerate machine learning and deep neural network research. It was built to run on multiple CPUs or GPUs and even mobile operating systems, and it has wrappers in several languages like Python, C++, and Java. A couple of years ago, deep learning started to outperform all other machine.

learning algorithms when given a massive amount of data. Google saw it could use these deep neural networks to improve its services: They built a framework called Tensor flow to let researchers and developers work together on an AI model. Once developed and scaled, it allows lots of people to use it. It was first made public in late 2015, while the first stable version appeared in 2017. It is open source under the Apache Open-Source license. You can use it, modify it and redistribute the modified version for a fee without paying anything to Google. Next, in this TensorFlow Deep learning tutorial, we will learn about TensorFlow architecture and how TensorFlow work. TensorFlow enables you to build dataflow graphs and structures to define how data moves through a graph by taking inputs as a multi-dimensional array called Tensor.

It allows you to construct a flowchart of operations that can be performed on these inputs, which goes at one end and comes at the other end as output. Tensor flow architecture works in three parts: It is called Tensor flow because it takes input as a multi-dimensional array, also known as tensors. You can construct a sort of flowchart of operations (called a Graph) that you want to perform on that input. The input goes in at one end, and then it flows through this system of multiple operations and comes out the other end as output. This is why it is called TensorFlow because the Tensor goes in it, flows through a list of operations, and then comes out the other side. TensorFlow hardware, and software requirements can be classified into Development Phase: This is when you train the mode. Training is usually done on your Desktop or laptop. Run Phase or Inference Phase: Once training is done, TensorFlow can be run on many different platforms. You can run it on You can train it on multiple machines, then you can run it on a different machine once you have the trained model. The model can be trained and used on GPUs as well as CPUs.

GPUs were initially designed for video games. In late 2010, Stanford researchers found that GPU was also very good at matrix operations and algebra so it makes them very fast for doing these kinds of calculations. Deep learning relies on a lot of matrix multiplication. TensorFlow is very fast at computing matrix multiplication because it is written in C++. Although it is implemented in C++, TensorFlow can be accessed and controlled by other languages, mainly, Python. Finally, a significant feature of TensorFlow is the Tensor Board. The Tensor Board enables to monitor graphically and visually what TensorFlow is doing." (Guru99) up to an acceptable level. It is also an essential part of solving problems where no readily available training data contains all the details that must be learned. Many supervised ML problems begin with gathering a team of people who will label or score the data elements with the desired answer.

For example, some scientists built a collection of images of human faces and then asked other humans to classify each face with a word like "happy" or "sad". These training labels made it possible for an ML algorithm to start to understand the emotions conveyed by human facial expressions." (Wayner).

5. Proposed Solution

When developing a model for making predictions, it is necessary to select a specific machine learning algorithm.

Because stock prices are time-series data, a Recurrent Neural Network was chosen for this study.

The recurrent characteristics of such a network work better than any other machine learning algorithm for predicting time-series data. This was one of the reasons why this study was conducted. The structure of the model can be explained using the flowchart presented in the following: To train the data, the data set is first scaled using MinMaxScaler.

Afterward, the data is trained. Before beginning to build up the RNN, a batch function must first be defined. To achieve the best possible outcome, critical parameters need to have their settings optimized. Following the RNN's initialization, the placeholders were created to store the values that are desired, but an LSTM cell is defined to cope with short-term memory storage and term memory, long term memory, dynamic RNN, and loss are all described to determine the term memory.

Error Squared Mean of Squares (MSE). Following an analysis of the MSE, the model will determine whether it ought to run the session to obtain the conclusive result for the projected fee or run once again. For fine-tuning the configuration of the RNN's parameters. In the end, after the required number of iterations, the model can make very exact and accurate predictions of the stock price.

6. Algorithm Description

Since stock price prediction is time series data, it is only right to employ the univariate forecasting. This is because the study focuses on how the closing price will change over time. We establish our target values in order to come up with the predicted stock price for that day.

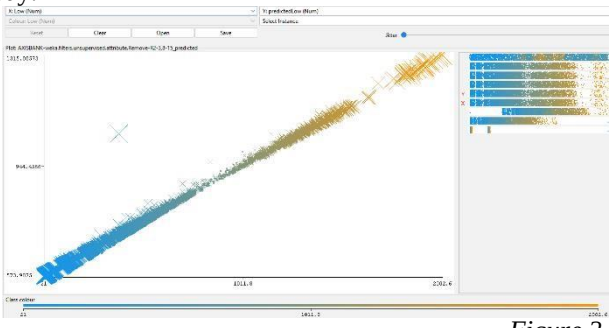


Figure 2.

a Gaussian process is a stochastic process (a collection of random variables indexed by time or space), such that every finite collection of those random variables has a multivariate normal distribution, i.e. every finite linear combination of them is normally distributed. The distribution of a Gaussian process is the joint distribution of all those (infinitely many) random variables, and as such, it is a distribution over functions with a continuous domain.

FIGURE 1.
Preprocessing of data

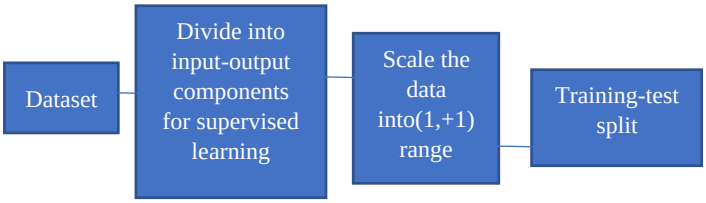
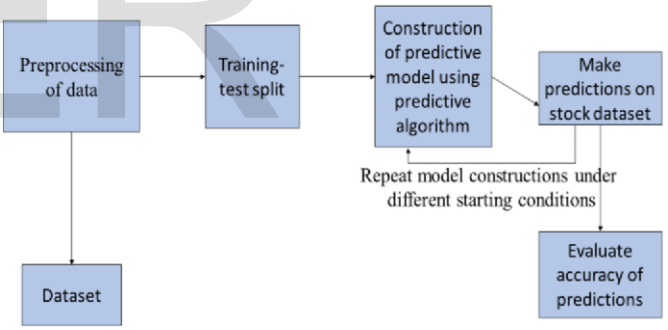


FIGURE 2.
Overall Architecture



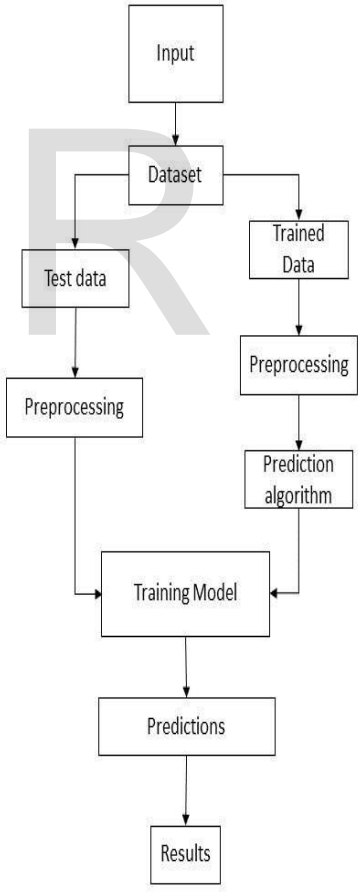
1: Input stock price data from portal. 2: Compute the Moving Average for 20, 50 and 200 days of stock price.
3: Moving Average = Stock Price1 + Stock Price2. . . Stock PriceN/Total Number of Days 4: Combine the moving average to identify the uptrend in stock price.
5: if (Stock Price > 20 Days & Stock Price > 50 Days & Stock Price> 200 Days)
6: then Uptrend in Stock Price
7: Else
8: Down Trend in Stock Price
9: Uptrend Stock Price are given input to PRE method.
10: Rule are generated using prediction rule ensembles method.
11: F(p) = x0 + åKk xkFk(p)
12: Selected the decision tree with the lowest RMSE score.
13: Later, Uptrend stock data given input to the RNN. 14: Constructed RNN model.
15: Compute H = d(WH1 + B)
16: Fine-tune the hyperparameters of the RNN method, such as the number of layers, learning rate, neurons, and number epoch in the model. 17: Average results of the PRE and RNN prediction model are combined.
18: Validate the results using 10 cross fold validation.
19: Evaluate the performance of model using RMSE and MAE metric.

Evaluation metrics are used to measure the predictive performance of ML models. Although accuracy is the most preferred metric in performance evaluation, it does not solely provide sufficient information to decide whether a model is good enough. Accuracy can also cause misleading results in cases of imbalanced data, which is a concept used to define the datasets where the distribution between classes is not close. Assessment metrics such as F-measure can calculate how well a classifier can distinguish between different classes even in the case of class imbalance (Gunduz et al. 2017a).

The principle of our hybrid prediction method for stock price is that these theories are the theoretical formation of our forecasting methods. The following first introduces the flowchart, the basic structure, and the process of the hybrid stock index prediction method based on the ensemble empirical mode decomposition and the longshort term memory neural network. Our proposed hybrid prediction method first uses the EEMD to decompose the stock index sequences into a few simple stable subsequences. Then, the predict result of each subsequence is predicted by the LSTM method. Finally, the LSTM obtains the final prediction result of the original stock index sequence by fusing all LSTM prediction results of several stock index subsequences.

FIGURE 3.

Structure Diagram



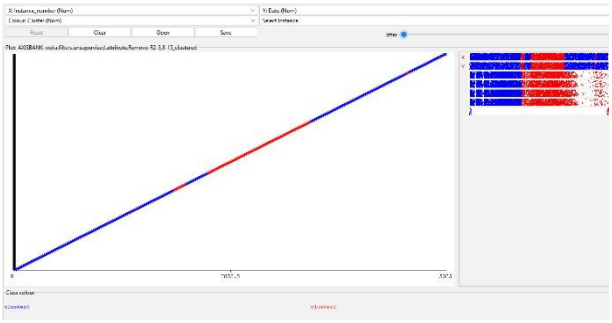
7. Result & Accuracy

This section describes the accuracy of the hybrid RNN prediction model. It achieves this by training the previously existing data set, validation, and crossvalidation of the data. The

precision of the model is vital for the general success of the prediction model. The training data functions as the measure of accuracy for the model as we can test the model using already existing data and check whether it will offer the correct results with precision. The test showed the model was accurate up to 60 percent for the stocks supplied into the model as input. This was based on the cross-validation test.

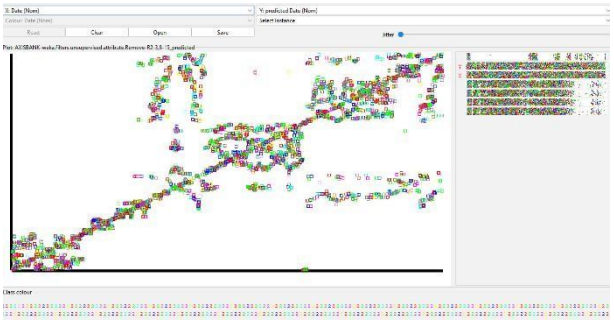
Epochs	Accuracy	MSE	RMSE
10	93.00717	207.6578	14.41034
20	94.01166	156.3873	12.50549
30	95.64188	105.3248	10.26279
40	95.59026	99.17409	9.958619
50	96.99466	62.24641	7.88964
epochs	Accuracy	100 98.28213337528945	
200	97.63336589796519		
300	96.94409289369247		
400	97.35454469043535		

Epochs for Google Dataset



We are comparing instant number and date

Date and predicted date



8. Conclusion

The purpose of this research was to build a model that could accurately forecast future stock values by utilizing a hybrid Recurrent Neural Network for processing time series data. The estimated value of the stock is not too far off from the actual value. To validate the performance of the model, the absolute value of the error %, cross validation of the data was done. The computed findings indicated a very significant correlation between the actual stock price and the forecasted price. To test the statistical hypothesis, the beta probability density function was used to create a sample histogram for KDE. Overall, we can infer that the created hybrid RNN model can successfully forecast the stock price one day in advance.

References

[1] Education, IBM Cloud. What Are Recurrent Neural Networks? | IBM. www.ibm.com, 14 Sept. 2020,https://www.ibm.com/cloud/learn/recurrent neural-networks.

[2] Wayner,Peter.VentureBeat.venturebeat.com,1Aug .2022,https://venturebeat.com/2022/08/01/whatis-supervised-machine-learning/.

[3] Guru99. www.guru99.com, 5 Jan. 2020, https://www.guru99.com/what-istensorflow.html.

[4] Zhu, Yongqiong. (2020). Stock price prediction using the RNN model. Journal of Physics: Conference Series. 1650. 032103. 10.1088/17426596/1650/3/032103.

[5] Trafalis, T. B., and Ince, H. (2000). Support vector machine for regression and applications to financial forecasting. In Neural Networks, 2000. IJCNN 2000, Proceedings of the IEEE-INNSENNS International Joint Conference on (Vol. 6, pp. 348-353). IEEE.

[6] Kohara, K., Ishikawa, T., Fukuhara, Y., and Nakamura, Y. (1997). Stock price prediction using prior knowledge and neural networks. Intelligent systems in accounting, finance and management, 6(1), 11-22.

[7] S. Selvin, R. Vinayakumar, E. A. Gopalkrishnan, V. K. Menon and K. P. Soman - Stock price prediction using LSTM, RNN and CNN-sliding window model - 2017.

[8] Murtaza Roondiwala, Harshal Patel, Shraddha Varma, "Predicting Stock Prices Using LSTM" in Undergraduate Engineering Students, Department of Information Technology, Mumbai University, 2015.

[9] Xiongwen Pang, Yanqiang Zhou, Pan Wang, Weiwei Lin, "An innovative neural network approach for stock market prediction", 2018

[10] Manh Ha Duong Boriss Siliverstovs June 2006 - The Stock Market and Investment.

[12] Dharmaraja Selvamuthu, Vineet Kumar and Abhishek Mishra Department of Mathematics, Indian Institute of Technology Delhi, Hauz Khas, New Delhi 110016, India - Indian stock market prediction using artificial neural networks on tick data.

[13] Lufuno Ronald Marwala A dissertation submitted to the Faculty of Engineering and the Built Environment, University of the Witwatersrand, Johannesburg, in fulfilment of the requirements for the degree of Master of Science in Engineering - Forecasting the Stock Market Index Using Artificial Intelligence Techniques.

[14] Xiao Ding, Kuo Liao, Ting Liu, Zhongyang Li, Junwen Duan Research Centre for Social Computing and Information Retrieval Harbin Institute of Technology, China - Event Representation Learning Enhanced with External Common-sense Knowledge.

[15] Huicheng Liu Department of Electrical and Computer Engineering Queen's University, Canada - Leveraging Financial News for Stock Trend Prediction with Attention-Based Recurrent Neural Network.

[16] Hyeong Kyu Choi, B.A Student Dept. of Business Administration Korea University Seoul, Korea = Stock Price Correlation Coefficient Prediction with ARIMA-LSTM Hybrid Model.

[17] M. Nabipour Faculty of Mechanical Engineering, Tarbiat Modares University, 14115-143 Tehran, Iran; Mojtaba.nabipour@modares.ac.ir - Deep Learning for Stock Market prediction.

[18] Lavanya Ra SRM Institute of Science and Technology | SRM · Department of Computer Science - Stock Market Prediction.

[19] M. Mekayel Anik · M. Shamsul Arefin (B) Department of Computer Science and Engineering, Chittagong University of Engineering and Technology, Chittagong, Bangladesh - An Intelligent Technique for Stock Market Prediction.

[20] Osman Hegazy, Omar S. Soliman and Mustafa Abdul Salam Faculty of Computers and Informatics, Cairo University, Egypt - A Machine Learning Model for Stock Market Prediction

[21] Sharanya Banerjee, Neha Dabeeru, R. Lavanya SRM Institute of Science and Technology, Chennai in Computer Science and Engineering - Stock Market Prediction

[22] Fahad Almudhaf, Yaser A. Alkulaib from Kuwait University - Are Civets Stock Markets Predictable?

[23] Mohammad Mekayel Anik, Mohammad Shamsul Arefin and M. Ali Akber Dewan, Department of Computer Science and Engineering - An Intelligent Technique for Stock Market Prediction

[24] Asep Juarna, Departemen of Informatics, Gunadarma University, Jakarta-Indonesia - ONE YEAR STOCK PRICE PREDICTION AND ITS VALIDITY USING LEAST SQUARE METHOD IN MATLAB

[25] Giovanis, Eleftherios - Applications of Least Mean Square (LMS) Algorithm Regression in Time-Series Analysis