JOURNAL OF ALGEBRAIC STATISTICS
Volume 13, No. 3, 2022, p. 1405-1414
https://publishoa.com
ISSN: 1309-3452

Stock Price Prediction Using Sentimental Analysis & Deep Learning Optimizer Techniques

P.N.V. Syamala Rao. M[1], N. Suresh Kumar[2]

[1]Research Scholar, Department of CSE,Gitam Institute of Technology, Vishakapatnam,AP, India.
[2]Department of CSE, Gitam Institute of Technology, Vishakapatnam, AP, India
E-mail:[1] syam.medaka@gmail.com, 2nskgitam2009@gmail.com

ABSTRACT

Stock prices in India are extremely volatile for a selection of reasons, which include political verdict outcomes, rumors, budgetary news, community safety events, and so on. Because of its fluctuating nature, stock price prediction is complex and difficult. The proposed research intends to create a novel methodology by employing deep learning techniques to combine sentiment analysis using time-series data on conventional stock market prediction. It takes sentiments from online sources like social media, Twitter, and integrates sentimental duality to improve prediction accuracy. In this study, we examine the sentimental analysis effectiveness of GRU-Gradient Descent with Adam optimizer. We concentrate on analyzing stocks in order to gain a better grasp of market fluctuations and to help us estimate prices more correctly. When compared to GRU with AdaGrad, Adam optimizer, Adadelta, and RMSprop, GRU - Gradient Descent with Adam optimizer produces better results.

Keywords- Sentiment Analysis, LSTM, GRU, Time Series data, Stochastic Gradient descent, Adagram, Adam, Stock price analysis**.**

1. INTRODUCTION

Emotion analysis is an automated approach for detecting the main sentiment regarding a product or company using social media data, developed utilizing modern mining techniques. Stock market prediction has be a prominent application of emotion analysis, and it is undeniably a topic under investigation. Forums and several internet communities are becoming the source for variety and volumes of data. Twitter is a perfect illustration where we can find the flow of information in the form of tweets counting to 600 million tweets per date. Though a single tweet may not be significant all by itself, a large number of them can provide data with useful insight into community opinion on a specific topic. Using Twitter to judge the public's view can be beneficial when developing trading strategies. Many aspects go into making an accurate prediction about stock price fluctuations, and public mood is arguably one of them.

In this study, we look at the TF-IDF embedded algorithm. "Term Frequency — Inverse Document Frequency" is abbreviated as TFIDF. This is a method for calculating the number of words in a collection of documents. We assign each word a score to indicate its prominence in the text and corpus. This approach is often utilized in text mining and information retrieval. This building is so high, for example. The semantics of the words and the phrase make it easier for us to comprehend the sentence. However, how can a computer programme (for example, python) decipher this? Data may be more easily understood by a programming language when it is represented numerically. Because of this, we need to vectorize all of the text so that it can be properly shown.

We may then conduct a variety of additional tasks, such as locating relevant documents, rating, grouping, etc., by vectorizing the documents. Searches on Google are carried out using a similar method (now they are updated to newer transformer techniques). Documents and queries are the terms used to describe the web pages you are searching. All documents are represented in the search engine's database in the same way. Using a query, the search engine determines which pages are most relevant to your search, ranks them according to importance, and displays the top k results. All of this is accomplished via the use of query and document vectorization.

Term Frequency

Counts how many times a term appears in a document. A common term like "was" may be used numerous times in a single text, depending on the length of the document and its generality. It's more likely that the word "was" is used more often in a document with 10,000 words than in a document with 100 words. The lengthier document, however, cannot be compared to the shorter document. It's for this same reason that we do normalisation on occurrence values, by dividing occurrences by their overall word count.

Remember that we still need to vectorize the document. Vectorizing papers is more complicated than just focusing on the words in a given document. If we do so, the vector length for the two papers will be different, making it impossible to determine the degree of similarity between them. In other words, we vectorize texts based on vocabularies. The vocabulary in the corpus is a comprehensive enumeration of all conceivable worlds.

JOURNAL OF ALGEBRAIC STATISTICS
Volume 13, No. 3, 2022, p. 1405-1414
https://publishoa.com
ISSN: 1309-3452

To calculate TF, we require the total number of words in the vocabulary list as well as the document's overall length. If the word does not appear in a given document, the TF value for that document will be 0. If every word in the page is identical, then the TF will be one. The normalized TF value will fall anywhere between [0 and 1]. In the range of 0 to 1.

For each document and word, TF may be expressed as follows:

tf(t,d) = count of t in d / number of words in d - (1)

Document Frequency
Document importance is measured in relation to the rest of the corpus. Unlike the frequency counter TF, it counts the number of times the word t is used throughout the document set N, rather than only once in the document d. Word use statistics may be found using the abbreviation DF. A occurrence is considered to be the first time a word occurs in the document. We don't care how many times the term is used.

df(t) = occurrence of t in N documents – (2)

Divided by the total number of papers, we can keep this with in a reasonable range. DF is the exact opposite of a term's informativeness, which is our primary purpose. as a result, the DF is inverted.

Inverse Document Frequency
Inverse document frequency (IDF) is a measure of how useful a word is. Since they appear in practically all texts, stop words have a very low IDF value when we compute IDF (and N/df gives that word a very low IDF value). This gives us the relative weighting we're seeking for.

idf(t) = N/df–(3)

There are a few more issues with the IDF; for example, when the corpus size is huge, such as N=10000, the IDF value explodes. To counteract the impact, we use the IDF log. When a term is missing from the vocab, it is simply disregarded throughout the query process. However, in a few circumstances when we employ a set vocab and a few words from the vocab are missing from the text, the df will be 0. We smooth the number by adding 1 to the denominator since we can't divide by 0.

$$idf(t) = \log(N/(df + 1)) \quad -(4)$$

Finally, the TF-IDF score is calculated by multiplying the values of TF and IDF. There are many various versions of TF-IDF, but for now, we'll stick with the most basic one.

$$tf\text{-}idf(t, d) = tf(t, d) * \log(N/(df + 1)) \quad -(5)$$

The paper is set as follows. In Section 2 and 3 we outline similar research and put forward our approach on sentimental analysis and stock prediction respectively. In Section 4, we go over the datasets we used for this paper and the data pretreatment techniques we used. In Section 5, we put forward the sentimental analysis technique that we have created for this work. In Section 6, we forecast the sentiment scores to investigate a process for projecting stock price by examining the correlation between tweets and the stock.

2. Related Work

Hiransha.M. et al. [1] compared four deep neural networks: Recurrent Neural Networks (RNN), Convolution Neural Networks (CNN), and LSTM and come to the conclusion that CNN outperforms the others. A hybrid network that integrates different models can improve the forecast even more. Hoseinzade E. et al. [2] proposed two CNN variations, one for data taken from a single source and another for data collected from a variety of markets. However, because CNN's prediction is independent of previous outcomes, it is not ideal for time series prediction. To overcome this drawback, based on generative adversarial networks (GAN), K. Zhang et al [3] developed a method for predicting the stock's close price, in which the originator is built with LSTMs and the discriminator is built with MLPs. Furthermore, by using a proper optimizer and integrating other factors that affect economic data, this model can be improved.

Shi. L, et. al's work [4] revolved around text-based predictions using DNN, pattern recognition, and finding its accuracy with real-world applications; this technique can be boosted by training and evaluating with various media messages and economic news events. Chen. L, et al. [5] proposed an approach that combines autoencoder, deep learning model, and restricted Boltzmann machine. It turned out to be superior to existing deep learning approaches such as

extreme learning machines, radial basis purpose neural networks, and backpropagation feed forward neural networks. The model can be enhanced by including more aspects such as politics, economy, culture, environment, and so on. Zhang, X., et al. [6], introduced a original data extraction approach for extracting facts from a variety of sources, including the online news and social media events were investigated and a multi-instance learning model was built for classification. To elevate accuracy in predicting values, other sources can be used in addition to Twitter tweets.

Long W. Lu. Z, et al. [7] proposed a deep learning-based feed neural network to extract features from multivariate financial time series data and compared the results to Recurrent neural networks and Convolutional neural networks. To additionally improve results, other factors affecting steadiness and profitability can also be studied. Chen.Y, et al. [8] introduced a hybrid stock index price prediction framework based on machine learning techniques, in which Support Vector Machine (SVM) was used to allocate weights during the training phase, and these attribute weights were used during the testing phase by K-nearest neighbor (KNN) machine learning algorithm. For further development, various methods for assigning weights and hybrid models could be studied. Patel.J, et al. [9] employed four machine learning models: Naive-Bayes, Support Vector Machine , random forest and ANN, with two distinct techniques for input data. They can only predict closing prices of Stocks. We estimated comparative strength, stochastic oscillator, and MACD.

An ensemble of different machine learning approaches can improve the prediction even more. The Auto-Regressive Integrated Moving Average model in short ARIMA, a combination of AutoRegressive (AR) and Moving Average (MA) models, was used by Idrees. S. M, et al. [10] to construct a linear model to predict future prices. However, this model can only forecast univariate time-series data. The ARIMA model was enhanced with a composite model to reduce the error rate.

In [11], to predict stock price, a method was presented that consolidated theme based sequence resetting with Convolutional neural network. To improve accuracy, sentiments from other real time sources can be incorporated. Nousi P and Tsantekidis A [12] proposed the Numerical-based Attention (NBA) methodology, which utilizes a combination of news and numerical information, for market estimation. AI and profound learning models were utilized to dissect the NBA. In [13], Two-stream Gated Recurrent Unit (TGRU) was proposed, which utilizes both feeling and monetary information. However, the intricacy of TGRU is double that of GRU, which takes more computational resources and training time. [14] proposed an SVM-based estimation approach that incorporates investor psychology by using sentiment data acquired from news and information Deep learning-based algorithms can manage large volumes of data.

3. Proposed Methodology

For further developed securities exchange forecasts, the recommended strategy incorporates two kinds of data. The first is information gleaned from Twitter and current events. The crude information winnowed from online media stages is overflowing with commotion. As a result, pre-processing is required to eliminate extraneous data. Natural language processing algorithms recognize the brain science of financial backers and the effect of information occasions from preprocessed information, resulting in polarity ratings for news and web-based media information. Monetary time series information acquired via web crawling is the second input. LSTM is an appropriate deep learning algorithm for processing time series data.

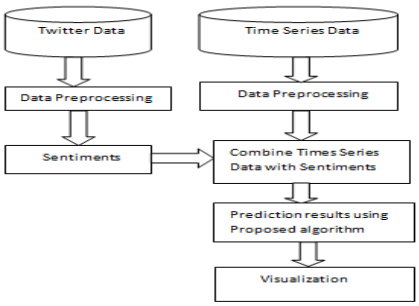


Figure 1: Proposed Model

GRU is considered to evaluate the time series data to avoid vanishing and exploding gradient descent problem which was found in the case of LSTM. The goal now is to apply sentiments gathered from online news websites and community media events to the prediction of economic time series data. To link feelings with time-series data prediction, a basic strategy has be proposed. Figure 1 depicts the detailed flow of activities.

4. Data Collection

For this study, we gathered data from TCS(Yahoo Finance). From September 2013 to September 2021, the every day stock price dataset contains final stock prices of the Tata Consultancy Services. We also gathered news stories for TCS firms from twitter & other social media sites between September 2013 and September 2021. A total of 65463 items have

JOURNAL OF ALGEBRAIC STATISTICS
Volume 13, No. 3, 2022, p. 1405-1414
https://publishoa.com
ISSN: 1309-3452

been collected. Making use of accessible data to make an learnedjudgment is the most difficult component of stock price prediction. For many businesses, a large amount of data is generated, and if this data were to be handled manually, it would take a long time. It would be difficult to make a choice in a timely manner. As a result, deep neural network models are used to process data as it is generate. To collect statistics, textual in a row, and stock prices from the net, we constructed a web scraper. This unprocessed data is supplied into a data tube, which process it and feeds it to a deep learning engine, which detects feelings in the provide texts and their impact on stock values, as well as forecasting future stock prices.

Table 01 : TCS Tweets

TCS
Oct-26-21 11:02AM
RT TCS makes first digital acquisition buys design studio W12 Technology News ETtech
10:24AM
TCS makes its first digital acquisition all you need to know about TCS buyout of London s W12 Studio
09:44AM
Stocks Radar Axis Bank Hindustan Petroleum Indraprastha Gas TCS
08:47AM
RT PLANNING YOUR RACE DAY OUTFIT Include our LovePartiesHateWhips accessories and share your pic with the LovePartiesHateWhips h
08:31AM
Se algu m que voc n o conhece lhe diz oi o que voc diz Oi
08:20AM
Tcs acusa Bolsonaro de ditador fascista e agora o Bolsonaro chama um magistrado reconhecendo as
08:13AM
RT Personendrohnen in Jetgr sse selbstfahrende Autos f hrerstandlose Z ge Welchen Einfluss die Digitalisierung auf di
08:00AM
RT na Check out Pepper the humanoid robot welcoming people to the TCSNYCMarathon
07:53AM
RT project ygo CS 12 22 Akkun CS 12 23 Tcs
07:40AM
RT The new Tax Collection at Source is here to ensure better and centralised compliance lesser revenue leakages for the gove
07:14AM
RT RT this post and follow us for your chance to twirl into winter and win a Ballet Theatre and Ice Skating Friends We ve g
06:55AM

Table 02: Time Series Data from TCS with Polarity scores

Date	open	High	Low	Close	Adj Close	Volume	Sentiment
2019-08-26	51.47	51.80	51.26	51.62	51.12	104174400	Positive
2019-08-27	51.97	52.14	50.88	51.04	50.54	103493200	Positive
2019-08-28	51.03	51.43	50.83	51.38	50.88	63755200	Neutral
2019-08-29	52.13	52.33	51.67	52.25	51.74	83962000	Positive
2019-08-30	52.54	52.61	51.80	52.19	51.67	84573600	Negative

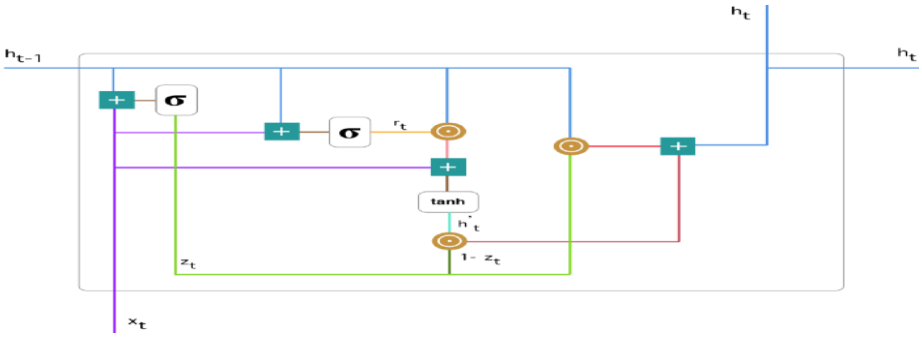
5. Experiments

GRU

GRUs are enhanced versions of traditional recurrent neural networks. But what distinguishes them and makes them so effective? GRU employs the U-Gate and R-Gate to overcome the vanishing gradient problem in standard Recurrent Neural Network. Essentially, these are a couple of vectors that figure out what data should be passed to the output. They help in holding data from a long time ago without vanishing it away through time or removing data that is immaterial to the prediction. The gates figure out which information is significant and will be valuable later on, and which information should be neglected. The gates are

1. Update gate
2. Reset gate

JOURNAL OF ALGEBRAIC STATISTICS
Volume 13, No. 3, 2022, p. 1405-1414
https://publishoa.com
ISSN: 1309-3452



ADAGRAD
When it comes to optimizers, one drawback is that the learning rate doesn't alter with time. Improves the pace at which new information is absorbed. In each parameter and time step "t," it alters the learning rate. Basically, it's a second-order algorithm. It is derived from an error function's derivative.
ADADELTA
AdaGrad is a programme that addresses the issue of slowing learning rates. Prior gradients are only gathered in a window of w rather than all previously squared gradients. In this case, an exponential moving average is used instead of the sum of all gradients.
RMSprop
The RMSprop optimization approach is based on gradients. It was suggested by Geoffrey Hinton, the originator of back-propagation. Gradients tend to evaporate or increase when input passes through very sophisticated processes like neural networks (refer to vanishing gradients problem). As a stochastic mini-batch learning method, Rmsprop was developed. To avoid the vanishing gradient issue, RMSprop normalises the gradient using a moving average of squared gradients. This normalisation reduces the step size (momentum) for large gradients to prevent bursting and increases it for small gradients to avoid disappearing. To summarise, RMSprop treats learning rate as a variable parameter rather than a hyperparameter, implying that learning rate varies with time.
Modified -TFIDF, GRU - Gradient Descent with ADAM
GD is the most fundamental and widely utilized method of optimization. Algorithms for regression and classification often use it. The gradient descent method is often used in neural network back propagation. The first order derivative of the loss function is used in GD, which is a first-order optimization technique. To simplify the function, it shows how the weights should be altered. To reduce the loss, the model's parameters, often referred to as weights, are adjusted when the loss is transmitted from one layer to the next.
Adaptive Moment Estimation is a GD optimization approach that is very successful when dealing with complex problems containing a significant quantity of data or parameters. It is more efficient and consumes less memory. The 'GD with momentum' and the Root Mean Square Propagation algorithms are a natural fit. This approach considers the gradients' 'exponentially weighted average,' which speeds up the gradient descent procedure. The technique converges to the minima faster when averages are used.

Proposed Algorithm

1. Collect Raw Data

2. Data preprocessing apply: Tokenize the text in to the terms, stop words etc….

3. tf(t,d) = count of t in d / number of words in d - (Term Frequency)

4. df(t) = occurrence of t in N documents - (Document Frequency)

5. idf(t) = N/df - (Inverse Document Frequency)

6. idf(t) = 1-log(N/(df + 1))

1. tf-idf(t, d) = tf(t, d) * (sum(I to N) 1- log(N/(df + 1)))
(t: term, D: Document, N= Documents, I=1…n)

7. Apply Modified -TFIDF, GRU - Gradient Descent with ADAM

JOURNAL OF ALGEBRAIC STATISTICS
Volume 13, No. 3, 2022, p. 1405-1414
https://publishoa.com
ISSN: 1309-3452

$$m_t = \beta_1 m_{t-1} + (1-\beta_1)\left[\frac{\delta L}{\delta w_t}\right] v_t = \beta_2 v_{t-1} + (1-\beta_2)\left[\frac{\delta L}{\delta w_t}\right]^2$$

5.Results and Analysis

GRU With Adadelta optimizer:

Table 03 and Fig. 01 show the Prediction of stock prices using time series. The figure depicts the real and Predicted values outcome of the stock's closing price prediction. The figures show the value of prediction is higher than the original stock price.

Table3. Actual and predicted values of close stock

Date	Real	Predicted
2020-08-19	2222.709717	2552.397705
2020-08-20	2219.016113	2450.072998
2020-08-21	2214.829834	2533.031250
2020-08-24	2214.337402	2475.502441
2020-08-25	2208.969482	2487.713867

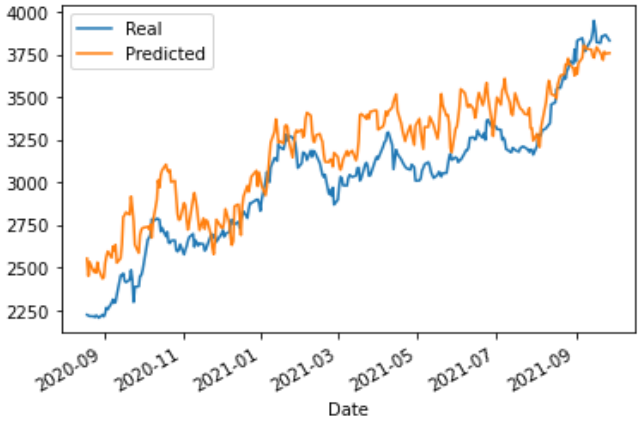


Fig: 02 Real and Predicted of time series data using GRU With Adadelta optimizer

Table4. Performance Metrics on GRU with Adadelta Optimizer

GRU with Adadelta Optimizer
Root Mean Squared Error: 0.12243151003483592
R-squared : 0.7049384563013845

GRU With Adagrad Optimizer

The Predicting of stock prices using time series data is demonstrated in Tab.05 and Fig.02 The figure represents the training and testing of the outcome of the stock's closing price. The figures show the value of prediction is higher than the original stock price.

Table: 05 Actual and predicted closing value of stocks

Date	Real	Predicted
2020-08-19	2222.709717	2435.484779
2020-08-20	2219.016113	2435.484779
2020-08-21	2214.829834	2435.484779
2020-08-24	2214.337402	2435.484779
2020-08-25	2208.969482	2435.484779

JOURNAL OF ALGEBRAIC STATISTICS
Volume 13, No. 3, 2022, p. 1405-1414
https://publishoa.com
ISSN: 1309-3452

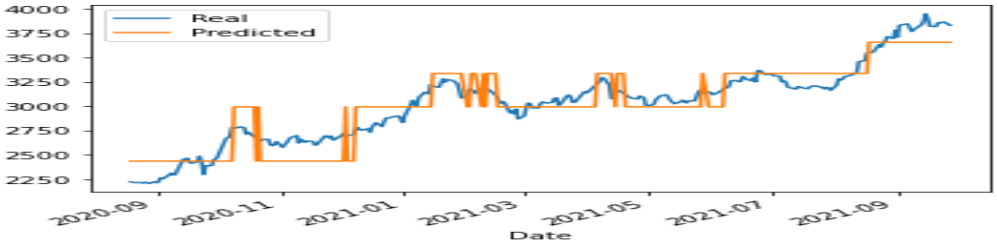


Fig:03 Real and Predicted values of time series data using GRU With Adam Optimizer

Table:06 Performance Metrics on GRU With Adam Optimizer

GRU with AdamPerformance Metrics
Root Mean Squared Error: 0.088030452443548
R-squared : 0.8474570762332728

GRU WITH ADAM OPTIMIZER

The Predicting of stock prices using time-series information is depicted in Tab.07 and Fig.03 Figure depicts the training and testing of the stock's closing price prediction outcome. The figures show the value of prediction is lower than the original stock price.

Table: 07 Actual and predicted closing value of stocks

Date	Real	Predicted
2020-08-19	46.39	51.604271
2020-08-20	46.13	50.025772
2020-08-21	46.22	50.789448
2020-08-24	48.95	49.647217
2020-08-25	50.38	50.632050

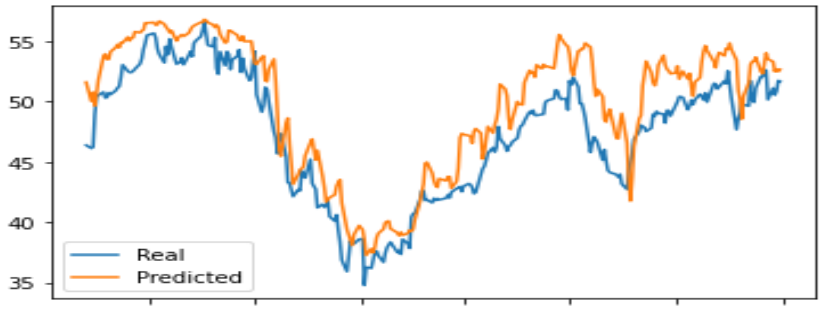


Fig: 04 Real and Predicted values of time series data using GRU With RMSProp

Table: 08 Performance Metrics on GRU With RMSprop Optimizer

GRU With RMSprop
Root Mean Squared Error: 0.1333872761480421
R-squared : 0.6800711327825214

GRU - Gradient Descent with Adam

Tab:9 and Fig:4 illustrate the predicting of closing prices using time series. Figure 5 depicts the training and testing of the stock's final price forecast. The figures show the value of prediction is higher than the original stock price.

Table: 09 Actual and predicted closing value of stocks

Date	Real	Predicted
2020-08-19	2222.709717	2234.730469
2020-08-20	2219.016113	2403.244141
2020-08-21	2214.829834	2262.406494
2020-08-24	2214.337402	2287.150146
2020-08-25	2208.969482	2201.097900

JOURNAL OF ALGEBRAIC STATISTICS
Volume 13, No. 3, 2022, p. 1405-1414
https://publishoa.com
ISSN: 1309-3452

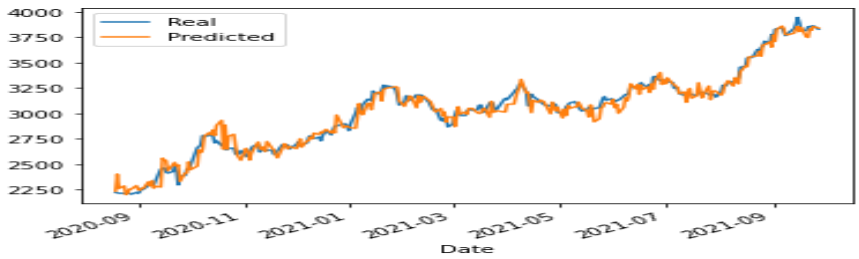


Fig: 05 Real and Predicted values of time series data using GRU Gradient Descent with Adam

Table: 10 Performance Metrics on GRU Gradient Descent with Adam

Proposed Algorithm Performance Metrics
Root Mean Squared Error: 0.03792352160569744
R-squared : 0.9716897622157885

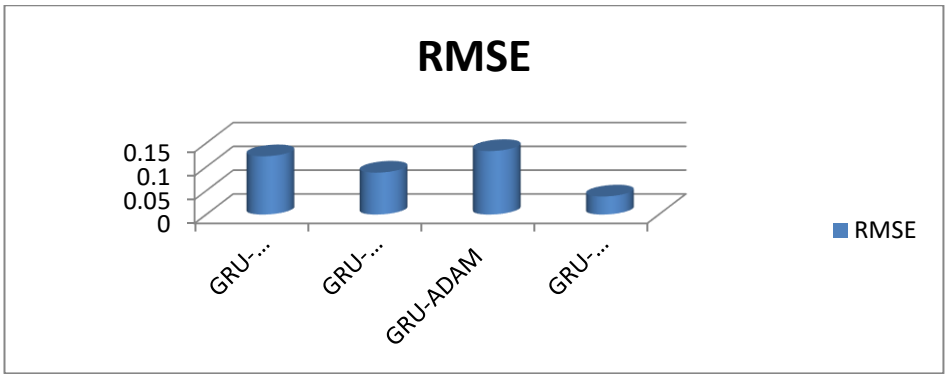


Fig: 06 RMSE Values for all Models

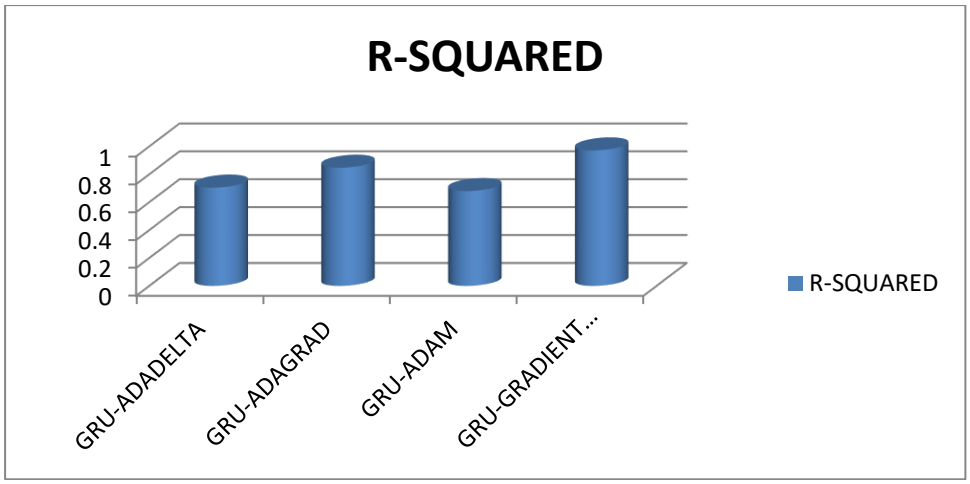


Fig: 07 R-Squared values for all Models

JOURNAL OF ALGEBRAIC STATISTICS
Volume 13, No. 3, 2022, p. 1405-1414
https://publishoa.com
ISSN: 1309-3452

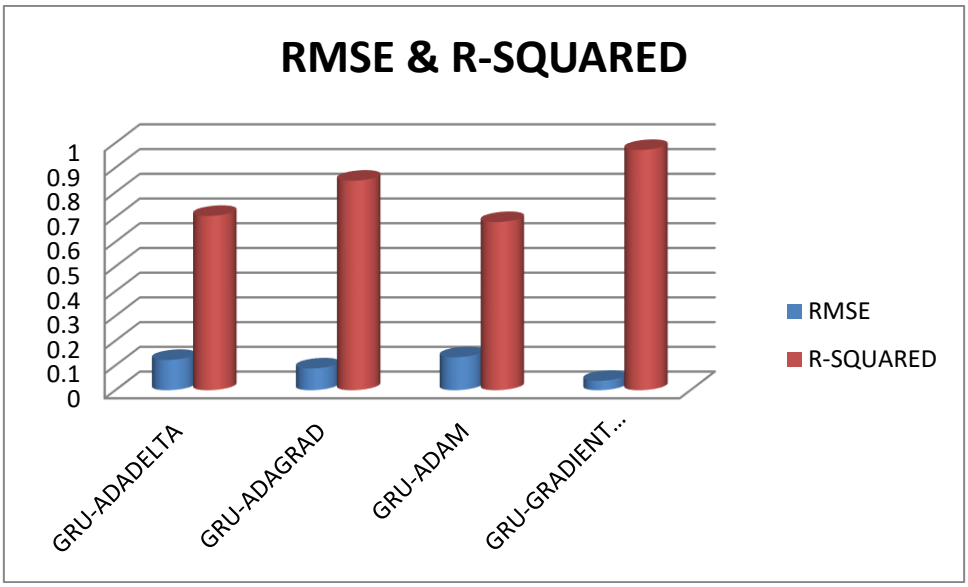


Fig: 08 Performance Metrics for all Models

6. Conclusion

In the context of the Indian economy, the suggested work proposes an accurate technique for stock market prediction. It extricates extremity scores from news and online media articles using sentiment analysis, then combines the derived sentiments with previous stock time-series data to predict the closing price. Because events and investor thinking have straightforward effects on the stock, the projected method produces accurate findings. The proposed method has a percentage error of roughly 2.9, which is lower than other current methods. As a result, it assists investors in making more informed judgments throughout various stock marketplace situations. In the outlook, sentiment from other online resources can also be added to pick up the suggested framework execution As well as Fundamental and Technical Features also. Furthermore, by combining more stocks from different spaces and establishing the relationship between them, a better prediction can be made.

References

1. Hiransha. M, Gopalakrishnan. E. A, Menon. V. K and Soman.K. P, “NSE stock market prediction using deep-learning models”, Procedia computer science, Vol. 132, pp. 1351-1362, 2018.
2. oseinzade. E and Haratizadeh. S, “CNNpred: CNN-based stock market prediction using a diverse set of variables”, ExpertSystems with Applications, Vol. 129, pp. 273-285, 2019.
3. Zhang K., Zhong, G., Dong, J., Wang, S. and Wang, Y., “Stock Market Prediction Based on Generative Adversarial Network”, Procedia computer science, Vol. 147, pp. 400-406, 2019.
4. W. Tai, T. Zhong, Y. Mo and F. Zhou, "Learning Sentimental and Financial Signals With Normalizing Flows for Stock Movement Prediction," in IEEE Signal Processing Letters, vol. 29, pp. 414-418, 2022
5. Chen. L, Qiao. Z, Wang. M, Wang. C. Du. R, and Stanley. H. E,“Which artificial intelligence algorithm better predicts the Chinese stock market?”, IEEE Access, Vol. 6, pp.48625-48633, 2018.
6. Zhang. X, Qu. S, Huang. J, Fang. B, and Yu. P, “Stock market prediction via multi-source multiple instance learning”, IEEEAccess, Vol. 6, pp. 50720-50728, 2018.
7. Long. W., Lu. Z and Cui. L, “Deep learning-based featureengineering for stock price movement prediction”, KnowledgeBased Systems, 164, pp.163-173, 2019.
8. Chen. Y and Hao. Y, “A feature weighted support vector machine and K-nearest neighbor algorithm for stock market indices prediction”, Expert Systems with Applications, Vol. 80, pp.340-355, 2017.
9. B. Shaikh, A. Iyer, M. Koneti and S. Iyengar, "Stock Price Prediction with Sentimental Analysis," 2022 4th International Conference on Smart Systems and Inventive Technology (ICSSIT), 2022, pp. 1632-1638
10. Idrees. S. M, Alam. M. A and Agarwal. P, “A Prediction Approach for Stock Market Volatility Based on Time SeriesData”, IEEE Access, Vol. 7, pp. 17287-17298, 2019.
11. A. Goel, "Stock price prediction using MLR on Sentiments and Fundamental Profile," 2021 12th International Conference on Computing Communication and Networking Technologies (ICCCNT), 2021, pp. 1-6
12. Liu G, Wang X., “A numerical-based attention method for stockmarket prediction with dual information”, IEEE Access, Vol. 7, pp. 7357-67, Dec 2018.

13. U. Shah, B. Karani, J. Shah and M. Dhande, "Stock Market Prediction Using Sentimental Analysis and Machine Learning," 2021 2nd Global Conference for Advancement in Technology (GCAT), 2021, pp. 1-4
14. Y. E, P. S, N. A and C. S, "Product Aspect Ranking Using Sentimental Analysis," 2021 International Conference on System, Computation, Automation and Networking (ICSCAN), 2021, pp. 1-4