



IJRASET

International Journal For Research in
Applied Science and Engineering Technology



INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 11 **Issue:** XI **Month of publication:** November 2023

DOI: <https://doi.org/10.22214/ijraset.2023.56952>

www.ijraset.com

Call:  08813907089

E-mail ID: ijraset@gmail.com

Stock Market Prediction Using Deep Learning

Akash Soni¹, Vaibhav Pandey², Saumya Yadav³

Department of Computer Science and Engineering, Galgotias College of Engineering and Technology, India

Abstract: The stock market represents a dynamic and complex system influenced by various factors, making accurate prediction of stock prices a challenging task. In this investigation, we utilize deep learning techniques to boost the precision of predictions in the stock market. Specifically, we employ Long Short-Term Memory (LSTM) networks to analyze historical stock data and capture intricate patterns that traditional methods may overlook. Our methodology encompasses meticulous data preprocessing, encompassing feature selection and normalization, ensuring the optimal functionality of the chosen deep learning model. The dataset utilized, derived from NSE, provides a robust foundation for training and testing our model. Evaluation metrics, gauge the model's performance.

Keywords: Stock Market Prediction, CNN, LSTM, Feature Learning, Predictive Modeling.

I. INTRODUCTION

The unpredictable nature of stock markets has long captivated the attention of investors, financial analysts, and researchers alike. The quest for effective tools to forecast stock price movements remains a critical endeavor, given the myriad factors that influence market dynamics. In response to this challenge, our project embarks on an exploration of the potential of deep learning methodologies, with a specific focus on advanced models such as LSTM, to enhance the accuracy of stock market predictions.

Traditional methods of stock market prediction often grapple with the inherent complexities of financial markets, where trends can be elusive and influenced by a multitude of factors ranging from economic indicators to global events. Deep learning, with its ability to capture intricate patterns within vast datasets, presents a promising avenue for addressing the limitations of conventional approaches.

The core objective of this project is to leverage the power of deep learning to discern patterns and relationships within historical stock data that may elude traditional methods. Through the meticulous preprocessing of data, involving feature selection and normalization, we aim to optimize the performance of the chosen deep learning model. As we navigate the early stages of this exploration, our preliminary findings underscore the potential of the selected model in capturing nuanced market dynamics.



Fig. 1. An illustration of how social media and finance news has impact on stock market movements.

Our project is poised to enter a phase of expanded analysis, incorporating diverse datasets to enrich the model's understanding of varying market conditions. This expansion not only contributes to the robustness of our predictive framework but also lays the groundwork for future investigations. As we progress, we anticipate refining our model, exploring alternative architectures, and ultimately advancing the state of stock market prediction through the integration of deep learning methodologies.

This introduction sets the stage for our project, outlining the motivation behind our exploration, the challenges we aim to address, and the potential impact of leveraging deep learning in the dynamic realm of stock market prediction.

II. PROBLEM STATEMENT

The forecasting of stock prices remains a challenging task, primarily due to the inherent complexities and dynamic nature of financial markets. Existing approaches, including traditional statistical and econometric models, often fall short when confronted with financial time series data characterized by significant randomness. The limitations of such methods become pronounced, leading to suboptimal forecast accuracies when attempting to capture the intricate patterns and trends present in the stock market.

Moreover, while machine learning models have shown promise, there is a need for more sophisticated and adaptive techniques to navigate the complexities of financial markets effectively. Specifically, the challenge arises when financial time series data exhibits a substantial random component, impacting the predictive capabilities of existing models. The problem at hand is to address the limitations of conventional forecasting methodologies in the context of stock price prediction. This project aims to develop models that can better accommodate the inherent randomness in financial time series data. By leveraging advanced deep learning techniques, such as convolutional neural networks (CNNs) and long- and short-term memory (LSTM) networks, the goal is to enhance the accuracy of stock price forecasts by capturing nuanced features from historical data. Additionally, the execution time of these models is a critical consideration. While achieving high accuracy is a primary objective, the computational efficiency of the models is crucial for practical implementation, ensuring real-time or near-real-time predictions on the selected hardware architecture. Therefore, the problem statement centers around the need for improved stock market prediction models that can effectively handle randomness in financial time series data, offering enhanced forecast accuracies while maintaining reasonable computational efficiency.

III. LITERATURE REVIEW

Numerous algorithms are available for forecasting stock market values, with Neural Networks, Genetic Algorithms, Association, Decision Trees, Fuzzy Systems, and Recurrent Neural Networks being widely adopted and commonly employed. A review of studies comparing algorithms indicates a prevalent trend in the practical application of technical indicators for stock price prediction over the past two decades.

Kim and Han examined algorithmic approaches in neural systems with the aim of predicting stock price values. The application of Genetic Algorithm extends beyond improving the training algorithm; it is also employed to mitigate the inherent complexity in the feature space. The genetic algorithm optimizes both the association weights between layers and thresholds for feature discretization simultaneously. The weights that evolved in genetic form overcome the general limitations of the gradient descent algorithm. Therefore, a genetic-based model surpasses conventional models.

Salim Lahmiri proposed a comparative analysis of Probabilistic Neural Networks (PNN) and the Support Vector Machines (SVM) algorithm for stock market analysis, incorporating both economic and technical information. The study revealed the exceptional performance of PNN in technical indicators, whereas SVM excelled in handling economic information.

Ming-Chang Lee and Chang proposed utilizing both backpropagation and support vector machine algorithms for predicting stock market values. Their study demonstrated that the support vector machine algorithm exhibited greater accuracy and correctness compared to backpropagation.

Alaa F. Sheta, Sara Elsir M. Ahmed, and Hossam Faris conducted a comparative analysis involving three algorithms—Regression, Artificial Neural Networks, and Support Vector Machines—for predicting the stock market index. Their study concluded that the designed SVM model with an RBF kernel exhibited superior prediction capabilities compared to the regression and ANN models. The results were validated using the number of criteria.

Sapkal et al., compared four Stock Market Prediction Algorithms such as Forecasting Algorithm, Moving Averages Algorithm, Regression Algorithm and Neural Network Algorithm and gave the best prediction decision based on these algorithms as artificial neural network (ANN).

Yang, Min, and Lin suggested the application of fuzzy neural systems for stock exchange value estimation. Their approach involved employing Genetic Algorithm to discover fuzzy rules, as discussed in Gupta, A., & Sharma, D. S. D. (2014). This paper presents a method to improve the discovery model by addressing issues that exist in its application. The change in information is accomplished primarily by applying Genetic Algorithm in fuzzy framework to find rules, dispense with errors caused by noisy data, and hence form valid set of principles. Likewise, fuzzy thinking approach is used based on the rule sets to anticipate value of stock market.

Mahdi, Hamidreza, and Homa investigated the prediction of stock exchange values using neural networks. In this paper, two sorts of neural systems, a feed forward multilayer perception (MLP) and an Elman recurrent network, are used to predict stock value dependent on its stock share value history. The use of MLP neural networks proves more effective in predicting stock price changes compared to Elman recurrent networks and the linear regression method. However, the direction of changes of stock value can be predicted better by Elman recurrent network and linear regression than MLP.

Shweta, Rekha and Vineet proposed foreseeing future trends in stock market by decision tree rough set based hybrid framework with Hierarchical Hidden Markov Model (HHMM). It presents a hybrid framework based on a decision tree rough set for forecasting trends in the Bombay Stock Exchange. This hybrid framework incorporates the Hierarchical Hidden Markov Model. It also illustrates forthcoming trends grounded in price-earnings and dividend.

Vivek Rajput and Sarika Bobde attempted stock market forecasting techniques by incorporating sentiment analysis from social media and various other data mining techniques to predict stock market values. However, following a detailed study, they concluded that stock prediction is a highly complex task, emphasizing the importance of considering various essential parameters in the process.

IV. METHODOLOGY

Stock prediction in my paper is based on 3 main algorithms, on analysis of which best algorithm for the prediction of the stock market can be evaluated.

A. Back Propagation

The Back Propagation Algorithm (Figure 1a) is applied to both classification and regression problems. In this context, predicting stock prices is framed as a classification problem. We implemented the Back Propagation algorithm for stock price prediction using the Numpy and Pandas libraries. Back propagation (backward propagation of errors) is a general supervised learning method to train artificial neural networks. Applied in conjunction with an optimization method like gradient descent. Back propagation calculates the error (gradient) of the network regarding the network's weights that can be modified. Back propagation is a multilayer feed-forward network. A Multilayer feed-forward neural network is a network that consists of an input layer, the hidden layer may be one or more than one and one output layer. A neural network that has no hidden layers is referred to as a Perceptron. In Back Propagation, there is a connection between the input layer and the hidden layer, and the hidden layer is connected to the output layer by interconnection of weights. The computational complexity of the neural network rises with an increase in the number of layers. This may result in the time taken for convergence and to minimize the error to be very high.

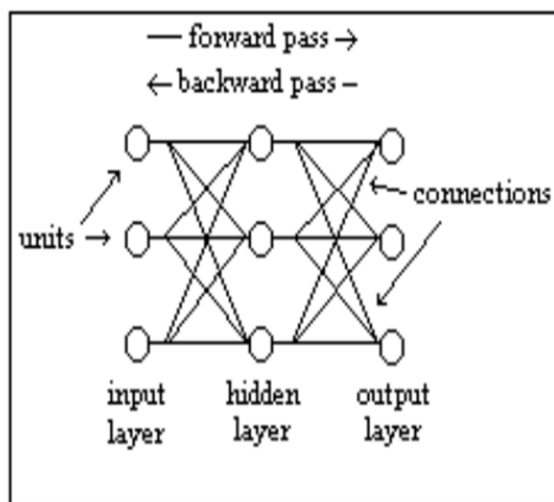


Fig.2 - Back Propagation

B. Long Short Term Memory (LSTM)

Long Short-Term Memory (LSTM) is a structural block or architecture within a neural network. LSTM blocks are employed to construct a recurrent neural network. Recurrent Neural Network (RNN) is a type of neural network (Ican & Celik, 2017) where the output of a block is fed as input to the next iteration. The main components of LSTM are:

- cell
- input gate
- output gate
- forget gate

The cell operates by retaining values at regular or arbitrary time intervals. Among these 3 gates, any of them can be called a conventional artificial neuron, as in a multi-layer feed-forward network. That means they compute an activation of a weighted sum. There are connections between cell and gate some of which are recurrent and some are not. As noted earlier, forecasting stock market movements is a time series problem. LSTM does not have the vanishing gradient problem that a traditional RNN has.

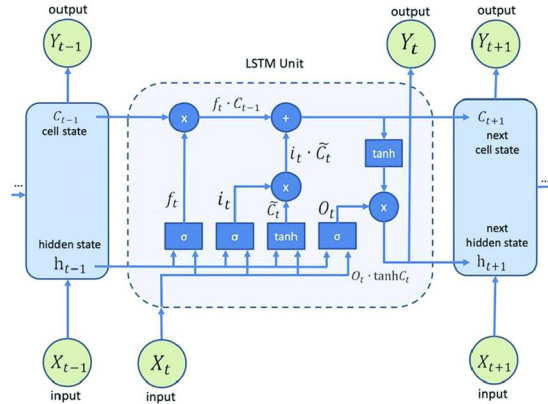


Fig.3. Long Short Term Memory

C. Support Vector Machine (SVM)

Support Vector Machine (SVM) is one of the prominent algorithms employed for various classification problems (Figure 3). It is one of the regulated learning models with related learning algorithms that examine the information utilized for arrangement and relapse analysis. From the given arrangement of training algorithm (Hegazy, O., Soliman, O. S., & Salam, M. A., 2014), the SVM training algorithm constructs a model that assigns out new models to either class, making it a non-probabilistic paired direct classifier. The SVM model is also referred to as the general representation of examples, mapped with the goal that the models of the different classifications are partitioned by clear gap i.e. as wide as could be allowed. New examples are additionally mapped into a similar space and anticipated to have a place with the class dependent on which side of the gap they fall. SVMs can also proficiently play out a non-straight arrangement utilizing the kernel trick, verifiably mapping their contributions to high-dimensional component spaces. As for the given problem, our data was not linearly separable, so we implemented the “RBF” kernel (Deepak, R. S., Uday, S. I., & Malathi, D., 2017) which gives better results for nonlinear kernels. For our problem, we implemented SVM using the Scikit Learn (sklearn) Library. Using Python code, we imported libraries, tried to run SVM on the training dataset and then predicted values on that machine for the test dataset.

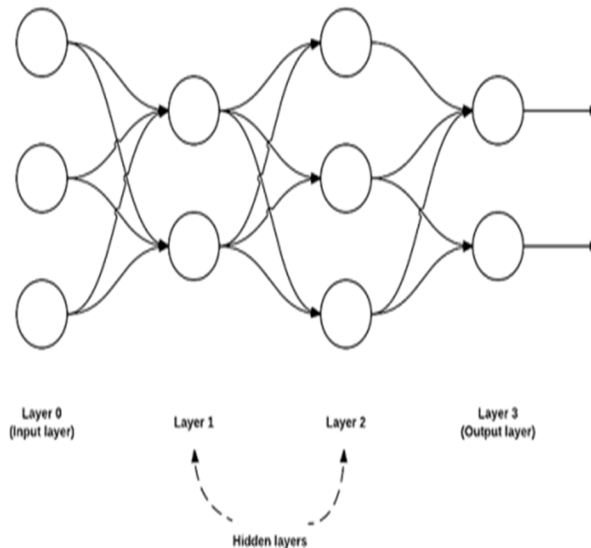


Fig.3- Support Vector Machine

V. COMPARISON AND RESULT

To ensure the consistency of one algorithm's performance and its superiority over others, we ran the same algorithm multiple times. Each algorithm was executed 30 times (Figure 4), and for each run, the prediction accuracy was calculated on the test data. For each run, different training and testing datasets are used.

```
Accuracy: 64.98450641876937
Accuracy: 65.30070195408841
Accuracy: 65.26275848985013
Accuracy: 65.28805413267564
Accuracy: 65.19951938278632
Accuracy: 65.21216720419908
Accuracy: 64.9718585973566
Accuracy: 65.31334977550117
Accuracy: 65.27540631126288
Accuracy: 65.28805413267564
Accuracy: 65.19951938278632
Accuracy: 65.04774552583318
Accuracy: 65.21216720419908
Accuracy: 65.32599759691394
Accuracy: 65.14892809713527
Accuracy: 65.18054765066718
Accuracy: 65.37658888256497
Accuracy: 64.78214127616518
Accuracy: 65.12363245430974
Accuracy: 65.16789982925441
Accuracy: 65.0414216151268
Accuracy: 65.28173022196927
Accuracy: 65.6295453108202
Accuracy: 64.99715424018213
Accuracy: 65.29437804338203
Accuracy: 65.20584329349269
Accuracy: 65.2690824005565
Accuracy: 65.31334977550117
Accuracy: 65.11730854360336
Accuracy: 65.19319547207994
n_epoch: 30
Mean_Accuracy: 65.20015177385694
Standard_Deviation: 0.15193494335888166
total time : 719.4723827838898
```

Fig.4- SVM

```
Accuracy:=68.026307
Accuracy:=68.114842
Accuracy:=60.412319
Accuracy:=64.579776
Accuracy:=67.185227
Accuracy:=64.877000
Accuracy:=68.114842
Accuracy:=68.114842
Accuracy:=61.752988
Accuracy:=66.805793
Accuracy:=67.754379
Accuracy:=67.337001
Accuracy:=62.050212
Accuracy:=68.121166
Accuracy:=62.227281
Accuracy:=66.382091
Accuracy:=66.065895
Accuracy:=67.836590
Accuracy:=68.089547
Accuracy:=66.110163
Accuracy:=67.994688
Accuracy:=63.599570
Accuracy:=68.114842
Accuracy:=68.076899
Accuracy:=68.114842
Accuracy:=64.301524
Accuracy:=64.428002
Accuracy:=62.764814
Accuracy:=66.628723
Accuracy:=68.114842
n_epoch: 30
learning rate: 0.5
Mean_Accuracy: 66.06990029300785
Standard_Deviation: 2.3292274542540126
total_time : 731.2970454692841
```

Fig.5- Back Propagation

```

Accuracy:63.81001548786783
Accuracy:67.95732231973842
Accuracy:67.75081741524694
Accuracy:67.47547754259163
Accuracy:65.90948201686456
Accuracy:66.02994321115126
Accuracy:67.95732231973842
Accuracy:65.87506453278266
Accuracy:64.54999139562898
Accuracy:64.60161762175186
Accuracy:67.63035622096025
Accuracy:67.87127860953365
Accuracy:63.99931165031836
Accuracy:68.1638272242299
Accuracy:65.84064704870075
Accuracy:67.38943383238686
Accuracy:67.33780760626398
Accuracy:67.50989502667355
Accuracy:67.8884873515746
Accuracy:67.66477370504217
Accuracy:67.80244364136982
Accuracy:64.75649630012046
Accuracy:65.8578557907417
Accuracy:66.27086559972466
Accuracy:65.47926346584065
Accuracy:66.54620547237997
Accuracy:64.9802099466529
n_epoch: 30
Mean_Accuracy: 66.55825159180864
Standard Deviation: 1.3068310850096652

```

Fig.6- LSTM

A. SVM Result

As illustrated in Figure 4, over 30 runs of the SVM algorithm, we obtained a mean accuracy of approximately 65.20%, with a standard deviation of 0.15. This indicates that SVM exhibits consistent performance over 30 runs, highlighting the robust nature of the SVM algorithm. The algorithm will continue training until it can classify the maximum amount of testing data. This results into almost same network at the end resulting same close accuracy values on test data.

B. Back propagation Result

As depicted in Figure 5, across 30 runs of the Back Propagation algorithm, we achieved a mean accuracy of approximately 66.0%, with a standard deviation of 2.32. As we can see Back propagation performs well as compared to SVM but it has huge fluctuation in accuracy so, it may cause an issue when we want steady accuracy.

C. LSTM Result

As illustrated in Figure 6, across 30 runs of the LSTM algorithm, we achieved a mean accuracy of approximately 66.55%, with a standard deviation of 1.30. Notably, LSTM performs well compared to SVM and Back Propagation. The minimal fluctuation in accuracy suggests an overall performance improvement compared to other algorithms.

Table: 1 Comparison Results

| | <i>SVM</i> | <i>Back Propagation</i> | <i>LSTM</i> |
|--------------------|----------------|-------------------------|----------------|
| Mean Accuracy | 65.20 | 66.00 | 66.55 |
| Standard Deviation | 0.15 | 2.32 | 1.30 |
| Variation | 0.023(approx.) | 6.243(approx.) | 1.851(approx.) |

D. T-Test

After obtaining accuracy scores for each algorithm, it is essential to perform a T-test to determine which algorithm outperforms the others.

$$T-0.95(58) = \pm 1.701$$

The T-test value for a 95% confidence level accuracy for a 30-epoch run fell within the range of ± 1.701 . If the value for each pair is within that range, it suggests a 0.05% chance of improvement with the other pair of algorithms. In the case of Back Propagation and SVM implementations, the T-test result is closest to the range.

Therefore, the likelihood of obtaining a better result from Back Propagation compared to SVM is very low, given the significant variations observed for each run in Back Propagation, as indicated in Table 1 and Table 2.

Table: 2 T-Test Results

| Comparison | T-Test Result |
|---------------------------|---------------|
| Back propagation and SVM | 1.885 |
| Back Propagation and LSTM | 1.133 |
| LSTM and SVM | 5.64 |

VI. CONCLUSION

In the scope of this project, we have presented a machine learning approach, specifically deep learning, for predicting stock market trends using various neural network architectures. The results showcase how historical data has been effectively utilized to predict stock movements with a notable level of accuracy. Through meticulous T-test result analysis, we can derive valuable insights, leading us to the conclusion that Long Short-Term Memory (LSTM) outperforms both Back Propagation and Support Vector Machines (SVM). Our findings underscore the potential of incorporating diverse factors influencing stock performance and feeding them into neural networks, coupled with rigorous data preprocessing and filtering. The trained neural network models exhibit a capacity to offer more accurate and precise predictions of stock momentum. Such predictive models have the potential to significantly contribute to firms' decision-making processes, leading to an augmented profit ratio compared to current practices. Furthermore, the integration of advanced predictive models contributes to increased transparency in the stock market. Firms can more effectively analyze losses and strategize for success, fostering a more informed and efficient financial ecosystem. As we navigate the landscape of stock market prediction, the amalgamation of sophisticated deep learning methodologies with comprehensive data analysis holds promise for shaping a more transparent, accurate, and profitable future for financial enterprises.

VII. ACKNOWLEDGMENT

I would like to express my sincere gratitude to all those who have contributed to the successful completion of this project on stock market prediction using deep learning methodologies. First and foremost, I extend my heartfelt thanks to my project supervisors, Dr. Shashi Kant Singh, and project coordinator Manish Kumar Sharma and Sanjay Kakhil for their invaluable guidance, unwavering support, and insightful feedback throughout this research. Their expertise and encouragement played a pivotal role in shaping the direction of this project. I am deeply appreciative of the resources and facilities provided by Galgotias College of Engineering & Technology, which have been instrumental in the execution of this study. The collaborative and enriching academic environment has significantly contributed to the quality of the research conducted.

REFERENCES

- [1] J. Sen and T. Datta Chaudhuri conducted a study titled "A comparative analysis of the Indian consumer durable and small-cap sectors, investigating an alternative framework for time series decomposition and forecasting, and evaluating its importance in the context of portfolio selection." This research was published in the Journal of Economics Library, volume 3, number 2, with pages ranging from 303 to 326 in the year 2016.
- [2] J. Sen and T. Datta Chaudhuri authored a paper titled "Gaining insights into the sectors of the Indian economy to inform portfolio selection." This research was published in the International Journal of Business Forecasting and Marketing Intelligence, volume 4, number 2, with pages ranging from 178 to 222 in the year 2018.
- [3] S. Mehtab and J. Sen conducted a study titled "Stock price prediction using convolutional neural network on a multivariate time series," presented at the Proceedings of the 3rd National Conference on Machine Learning and Artificial Intelligence (NCMLAI' 2020) in New Delhi, India, in 2020.
- [4] S. Mehtab and J. Sen authored a report titled "A time series analysis-based stock price prediction using machine learning and deep learning models," identified as Technical Report No: NSHM_KOL_2020_SCA_DS_1 at NSHM Knowledge Campus, Kolkata, India. The report includes a DOI: 10.13140/RG.2.2.14022.22085/2.
- [5] Z. Yan, Z. Huang, and M. Liang authored a paper titled "Stock prediction via linear regression and BP regression network," published in Communications in Mathematical Finance, volume 8, number 1, pages 1-20 in 2019.
- [6] T. Vantuch and I. Zelinka presented a paper titled "Evolutionary based ARIMA models for stock price forecasting" at the Interdisciplinary Symposium on Complex Systems (ISCS' 14) in 2014. The proceedings were published with page numbers 239-247.



- [7] Navale, G. S., Dudhwala, N., Jadhav, K., Gabda, P., & Vihangam, B. K. (2016). The research involves the prediction of the stock market through the application of data mining and artificial intelligence, indicating an exploration of advanced techniques to analyze and forecast market trends. *International Journal of Engineering Science*, 6539.
- [8] Deepak, R. S., Uday, S. I., & Malathi, D. (2017). Machine Learning Approach in Stock Market prediction. *International Journal of Pure and Applied Mathematics*, 115(8), 71-77.
- [9] A. A. Ariyo, A. O. Adewumi, and C. K. Ayo, "Stock price prediction using the ARIMA model", In *Proceedings of the 16th International Conference on Computer Modeling and Simulation (UKSim-AMSS)*, Cambridge, pp. 106-112, 2014.
- [10] W. Bao, J. Yue, and Y. Rao authored a paper titled "A deep learning framework for financial time series using stacked autoencoders and long-and-short-term memory," published in *PLOS ONE*, volume 12, number 7 in 2017.
- [11] M. R. Vargas, B. S. L. P. de Lima, and A. G. Evsukoff presented a paper titled "Deep learning for stock market prediction from financial news articles" at the *IEEE International Conference on Computational Intelligence and Virtual Environments for Measurement Systems and Applications (CIVEMSA)* in Annecy, France, in 2017.
- [12] L. D. S. Pinheiro and M. Dras contributed to a paper titled "Stock market prediction with deep learning: A character-based neural language model for event-based trading," presented at the *Australasian Language Technology Association Workshop* in Brisbane, Australia, with proceedings published on pages 6-15 in 2017.
- [13] Yahoo Finance Website: <http://in.finance.yahoo.com>
- [14] S. Mehtab, J. Sen and A.Dutta, "The research explores stock price prediction through the application of machine learning and LSTM-based deep learning models," In *Proc. of the 2nd Symposium on Machine Learning and Metaheuristic Algorithms and Applications*, Chennai, India, October 2020. (Accepted).



10.22214/IJRASET



45.98



IMPACT FACTOR:
7.129



IMPACT FACTOR:
7.429



INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Call : 08813907089  (24*7 Support on Whatsapp)