Journal of Research Administration
Society of Research Administrators International

# FORECAST THE DAILY INVESTMENT MARKET TREND USING A HYBRID MACHINE LEARNING TECHNIQUE

**Dr. Amit Nautiyal[1], Dr. Som Aditya Juyal[2], K Karpagavalli[3], Dr. A. Suresh Kumar[4]**
[1]Associate Professor, Himalayan School of Management Studies, Swami Rama Himalayan University, Dehradun
[2]Dr. Som Aditya Juyal, Professor, Himalayan School of Management Studies, Swami Rama Himalayan University, Dehradun
[3]Assistant professor, Annamacharya Institute of technology and Sciences, Rajampet
[4]Professor and Head, Department of MBA, VelTech High Tech Engineering College, Avadi

**Abstract -** Forecasting the movement of stock prices is vital for realizing the greatest possible return on an investment in stocks. It attempts to predict the path that stock prices will take in the future. Possible investors capitalize on the possibility of a crisis by redistributing their funds and turning prospective drawbacks into potential gains. But due to some losses in market trends, investors are hesitant to put their money into risky ventures. To overcome this issue, we proposed a Hybridized Random Support Vector Machine (H-RSVM) method. The purpose of H-RSVM method is to predict the daily market trend in investments. We collected the dataset of stock market investment and then collected data is preprocessed using min-max normalization. The preprocessed data is feature extracted using principal component analysis (PCA). As a result, our proposed H-RSVM method provides a superior performance in predicting the prices of stock market in terms of accuracy, precision, recall and f1-score measure.

**Keywords-** Stock market, Machine learning (ML), Investments, Prices, Funds, Hybridized Random Support Vector Machine (H-RSVM), Principle component analysis (PCA), Support vector machine (SVM), Random forest (RA).

## I. INTRODUCTION

The stock market has always been regarded as having one of the highest potential profit margins among the several investing avenues. However, generating a greater rate of return consistently over a longer time horizon requires a thorough knowledge of the market and a clear investment plan. One of the important foundational strategies is stock prediction, which aims to forecast future stock price trends. This methodology has drawn growing interest from great minds [1]. The value of the global stock markets has topped 68.654 trillion US dollars, according to the World Bank in 2018. The rise in interest in investing in shares throughout the last decade or so is mostly attributable to technical advances. Marketers seek tactics and tools that increase revenue while reducing risk. It's not easy to predict the stock market due to its unexpected, changeable, erratic, and inaccurate qualities. A kind of time-series estimation known as SMP swiftly analyses past data and forecasts the values of forthcoming data. The stock market is often used as a gauge for stock prices and volume. However, the stock price generation process exhibits complexity and unpredictability owing to the stock market's complexity, variability, and uncertainty [3]. Experts in finance and economics have long found it difficult to anticipate stocks. The basic strategy used to produce this prediction is to purchase equities with a high possibility of price growth and dispose of stocks with a high likelihood of price drop [4]. For companies, shareholders, and commodity traders, predicting stock prices is a difficult and complex undertaking to forecast potential profits. The stock markets are inherently noisy, random, non-stationary, haphazard, and deterministic processes. It makes it challenging to predict the price exactly and properly [5]. The objective is to predict the daily market trend in investments using H-RSVM.

Phase II of the paper covers related works, Phase III describes the proposed methodology, Phase IV includes the results and discussion, and Part V provides a conclusion for further research.

## II. RELATED WORKS

Reference [6] suggested that research, a rigorous big data analytics technique, was used to forecast the daily return direction of the Standard & Poor's Depositary Receipts (SPDR) Standard & Poor (S&P ) 500 Exchange-Traded Fund (ETF). The whole pre-processed but untransformed dataset was then subjected to Deep Neural Network (DNN) and conventional Artificial Neural Network (ANN) application; as well as both data sets being altered using PCA, the daily direction of imminent index-based stock market returns may be estimated. Reference [7] provided a unique framework for projecting stock closing prices has been developed. The deep hybrid framework's components included the machine learning classifier component, the predictor improvement approach, and the data processing component. Reference [8] proposed a deep neural network model to forecast stock price trends utilizing desensitized transaction records and open market data. In order to choose the appropriate stocks for the purpose of developing the market and trading information, their strategy makes use of the expertise in chart and chart insertion methodologies. In Reference [9], Long Short-Term Memory (LSTM) was used as the engine, and Multi-Layer Preceptor (MLP) is used as the filter in the Generative Adversarial Network (GAN) architecture for stock price prediction. The detector, created using MLP, aimed to differentiate between the actual stock information and the fabricated data, while the generator, which is developed using the LSTM, creates data in a similar manner and mines the stock market for stock information trends. As per reference [10], the issue of forecasting stock price changes and index values for Indian stock markets was discussed. In the article, 4 models for forecasting and 2 input techniques are compared. Reference [11] focused on stock market group future predictions. Four groups from the Tehran Stock Exchange were chosen for exploratory evaluations: integrated financials, fuel, minerals that are not metallic and essential metals. Information for the categories was compiled using a decade's worth of sources from history.

## III. PROPOSED METHODOLOGY

Data was gathered for this investigation, which was followed by pre-processing. The data is subsequently extracted utilizing Principal Component Analysis (PCA), and our suggested technique is used and described in detail. Fig. 1 denotes the representation of the suggested technique.

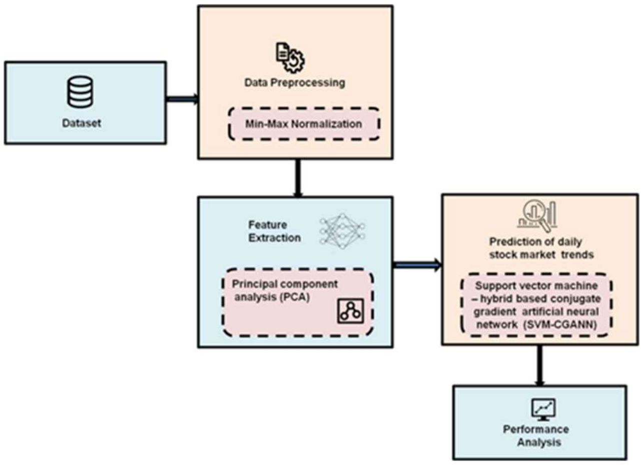
Dataset
Data Preprocessing
Min-Max Normalization
Feature Extraction
Principal component analysis (PCA)
Prediction of daily stock market trends
Support vector machine – hybrid based conjugate gradient artificial neural network (SVM-CGANN)
Performance Analysis

Fig.1 Architecture of Proposed Methodology

### Dataset

The collection of data in our experiment, creating relevant datasets from vast multi-modality sources, is difficult. We gather financial news, market prices, market events, and the knowledge graph for all 3714 publicly traded businesses listed on China A-shares market3 between January 1 and December 31, 2019, to assure fairness. To create events for the list firms, we specifically scan the public announcements and use event extraction techniques [12].

### Data pre-processing using min-max normalization

The normalization procedure ensures that the range of each piece of information in the database's records is uniform. This is especially important when the data is unstructured and has a large range of values. Information/data with high dimensions benefit from MinMax normalization. Electro Encephalo Gram(EEG) signal levels are measured in microvolts and vary significantly from channel to channel. This variance makes it difficult to train a model. A normalization method called MinMax may scale all EEG signal data to have values between 0 and 1.

$$Y_{std} = \frac{(Y-Y.min)}{(Y.max- \quad .min)} \qquad (1)$$

### Feature extraction using Principal Component Analysis (PCA)

PCA is a basic methodology in statistics and machine learning for reducing dimensionality. The original variables are changed into a new set of uncorrelated variables known as principal components. It is used to simplify complicated datasets while maintaining important information. Data compression, feature selection, and visualization are the three main applications of PCA.

The phases involved are as follows;

Phase 1: Row / column vector size $M3$ denotes the set of $N$ images $(A1, A2, A3 \dots AN)$ with size $M^*M$

Phase 2: The description of the training set image average ($\mu$) is

$$\mu = \frac{1}{n}\sum_{m=1}^{\prime N} Am \quad (1)$$

Phase 3: Each trainee image has a distinct average image by a vector (X).

$$Xj = Aj - \mu \qquad (2)$$

Phase 4: Covariance Matrix or Total Scatter Matrix is measured from $\Phi$ as shown below:

$$D = \sum_{m=1}^{\prime N} xmxms = BBS \quad (3)$$

Where $B = [X1X2X3 \dots Xn]$

Phase 5: Determine the covariance matrix D's eigenvalues $\lambda K$ and eigenvectors VK.

Phase 6: This feature space may be used to classify images. Measure the weight vectors.

$$\Omega S = [x1, x2, \dots, xN' \qquad (4)$$

Wherein

$$Gl = VlS\,(A - \mu), l = 1,2, \dots, N' \quad (5)$$

### A. *Hybridized Random support vector machine (H-RSVM)*

**Random Forest (RF)**

The RF classification is a collective approach that employs bagging, averaging, and bootstrapping to continuously train a number of decision trees. In different parts of the training samples, numerous independent decision tree structures may be concurrently built by employing various subsets of the given features. By guaranteeing that every decision tree inside the RF is distinct, bootstrapping lowers the RF variance. The final judgment in RF classification is combined from several tree judgments, which gives the classifier a great generalization. The RF classifier seeks to consistently perform better than almost all other classifier algorithms in terms of accuracy without issues with unbalanced datasets and overfitting. Equation (6) may be used to determine an RF's mean square error (MSE).

$$MSE = \frac{1}{M}\sum_{r=0}^{m}\binom{m}{r}(Lj - Xj)p^2 \qquad (6)$$

In equation (6), M is the total number of different data points, Lj denotes the model Xj's output, and the exact value of point value is j.

**Support Vector Machine (SVM)**

The SVM is a supervised kind of machine learning that addresses problems with the performance of various classifiers in classification. Once training examples for each segment have been provided, SVM features may categorize new material. Assume that we have 2-dimensional residual column vectors in normal class $\Omega_{mq}$ as $y_1, y_2, \dots . y_{m_1}$ and defective class $\Omega_{le}$ as $y_{m_1} + 1, y_{m_1} + 2, \dots . y_{m_1} + m_2$. Give the vectors in Group $\Omega_{no}$ a label of 1 and Group Ωfa a label of -1. SVM seeks to obtain an ideal classifier for these vectors and labels (the difference between the data in the various classes is as small as possible). The form of the SVM classifier is:

$$\mathrm{T(y)} = \mathrm{u}^{*^{\mathrm{D}}}\varphi(\mathrm{y}) + \mathrm{p}^*, \quad (7)$$

Where the residual vector in two dimensions, y, has to be categorized. u* is the best b-dimensional column vector to solve, p* is the best threshold, and $\varphi(y)$ is a simultaneous b-dimensional mapping from input space to feature space of the sample y. The data in $y_w$ are categorized by the classifier using:

$$\begin{cases} y_w \in \Omega_{no}, if T(y_w) > 0, \\ y_w \in \Omega_{fa,} otherwise. \end{cases} \tag{8}$$

Here, $u^*$ and $p^*$ are solutions of the problem:

$$\begin{matrix} min \\ u,p \end{matrix} \ \frac{1}{2}\|u\|^2 + V \sum_{j=1}^{m_1+m_2} \xi_j$$

$$s.t. x_j T(y_j) \geq 1 - \xi_j \,, \ \ j = 1,2,\dots,m_1 + m_2,$$
$$\xi_j \geq 0, \qquad\qquad j = 1,2,\dots,m_1 + m_2 \,, \quad (9)$$

Where $V$ is a parameter known as the coefficient of penalty., $x_j$ is the sample label $y_j$, and $\xi_j$ is the slack sample variable $y_j$. The issue is described as follows once this problem of optimization has been converted to the dual space:

$$\begin{matrix} max \\ \Lambda \end{matrix} \ \Lambda^D 1 - \frac{1}{2}\Lambda^D Z \Lambda$$

$$s.t. \ \begin{matrix} 0 \leq \Lambda \leq \mathrm{V}, \\ \Lambda^D X = 0, \end{matrix} \tag{10}$$

Where $\Lambda^D = (\alpha_1, \alpha_2, \dots., \alpha_{m1+m2})$ is the vector of the Lagrangian multiplier required to be modified, $\Lambda^D = (1,1,\dots,1)$denotes a $(m_1 + m_2)$-dimensional unit vector, $X^D = (x_1, x_2, \dots., x_{m1+m2})$ represents the label vector, and " $\leq$ " in this context signifies element-wise comparison. $Z$ represents a matrix of squares with the dimension of $(m_1 + m_2) \times (m_1 + m_2)$--× (n1 + n2). The elements of Z are:

$$Z_{r,f} = x_r x_f(y_r, y_f), R(y_r, y_f) = \varphi(y_r)\varphi(y_f), r, f = 1,2,\dots,m_1 + m_2 \tag{11}$$

Where $R(y_r, y_f)$ is the function of Kernel of $y_r$ and $y_f$.

The decision function, for instance, y in the dual space, has the following form:

$$T(y) = \sum_{j=1}^{m_1+m_2} x_j \alpha_j^* R(y_j, y) + p^* \qquad (12)$$

Where $\alpha_j^*$ for $j = 1, 2, \cdots, n_1 + n_2$ are the components of the ideal vector solution to equation (4). Support vectors are examples of a non-zero Lagrangian multiplier. The decision function for SVM is made up of the Lagrangian multipliers, labels for support vectors, threshold p*, and support vectors.

## IV. RESULTS AND DISCUSSION

A statistical metric called accuracy is used to assess how well a classification model is doing. Occurrences accurately predicted (both true positives and true negatives) as a percentage of all occurrences in a dataset are what this term denotes. In other words, accuracy assesses how well a model assigns data points to the appropriate categories. In mathematics, precision is often represented as a ratio or a percentage:

$$Accuracy = (Number\ of\ Correct\ Predictions) \ / \ (Total\ Number\ of\ Predictions) \qquad (13)$$

In this article, we achieved an accuracy of 95 % using the H-RSVM method. While other existing methods achieved accuracy, such as DWT and RNN achieved 82.5%, GA-XGBoost achieved 70%, CNN and RNN achieved 88%.

Fig. 2 denotes the contrast of accuracy with conventional and suggested techniques. Table I denotes the numerical outcomes of accuracy.

**TABLE I**
**COMPARISON OF ACCURACY**

| Methods | Accuracy (%) |
|---|---|
| DWT+RNN [13] | 82.5 |
| GA-XGBoost [14] | 70 |
| CNN+RNN [15] | 88 |
| H-RSVM [Proposed] | 95 |

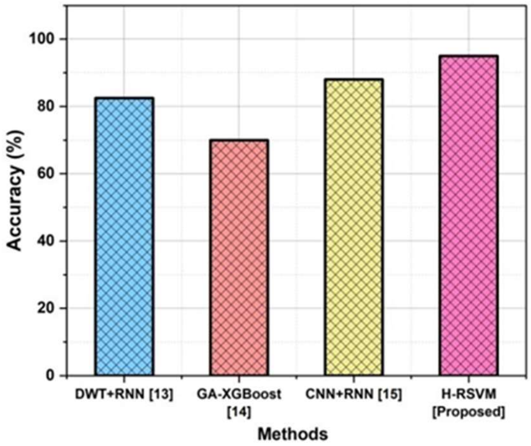

Fig.2 Contrast of accuracy with conventional and suggested approach

In binary classification and information retrieval tasks, precision is a performance parameter that is used to assess how accurately a model predicts positive outcomes. It focuses on the percentage of positive predictions that were accurately classified as genuine positives and offers information on the model's capacity to prevent false positives.

The precision formula is given by:

$$Precision = True\ Positives \,/\, (True\ Positives + False\ Positives) \tag{14}$$

True Positives are the number of positive instances that were really predicted to be positive. False positives are the quantity of negative events that were wrongly identified as positive. In short, it is the proportion of the model's predictions that are accurate. The model is excellent at producing positive predictions without making many false positive mistakes, as shown by a high precision. In this article, we achieved a precision of 93% using the H-RSVM method. While other existing methods achieved precision, such as DWT and RNN achieved 72.15%, GA-XGBoost achieved 82.75%, CNN and RNN achieved 86%.

Fig. 3 denotes the contrast of precision with the conventional and suggested approach. Table II depicts the numerical outcomes of precision.

**TABLE II**
**COMPARISON OF PRECISION**

| No. of data | Precision (%) | | | |
|---|---|---|---|---|
| | DWT+RNN [13] | GA-XGBoost [14] | CNN+RNN [15] | H-RSVM [Proposed] |

| | | | | |
|---|---|---|---|---|
| 1 | 88 | 81 | 83 | 96 |
| 2 | 81 | 82 | 87 | 95 |
| 3 | 89 | 87 | 90 | 92 |
| 4 | 87 | 82 | 85 | 90 |
| 5 | 90 | 83 | 85 | 93 |

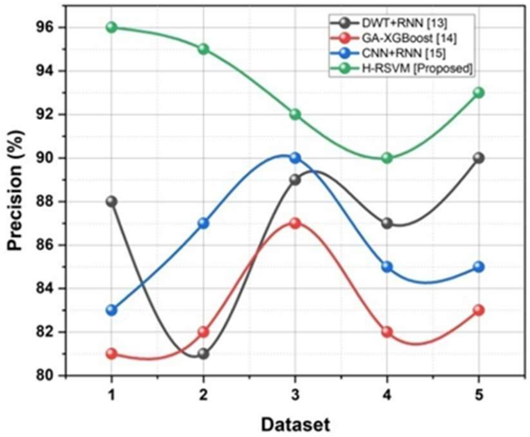

Fig.3 Contrast of precision with conventional and suggested approach

Recall gauges how well a model can recognize each relevant occurrence in a dataset. Taking into account both true positives and false negatives, it is determined as the proportion of true positive predictions to all of the dataset's actual positive cases. When it is expensive to overlook favorable events, recall is more crucial.

$$Recall = True\ Positives\ /\ (True\ Positives + False\quad Negatives) \tag{15}$$

Due to the fact that they provide complimentary insights into a model's performance, these two indicators are often employed in tandem. By altering the categorization threshold, accuracy and recall may be balanced more effectively. Precision may go up, but recall could go down if the criterion is raised (made stricter), and vice versa. In this article, we achieved a recall of 97.38% using the H-RSVM method. While other existing methods achieved recall, such as DWT and RNN achieved 87.35%, GA- XGBoost achieved 80.26%, CNN and RNN achieved 79.13%.

Fig. 4 denotes the contrast of recall with conventional and proposed techniques. Table III denotes the numerical outcomes of recall.

## TABLE III
## COMPARISON OF RECALL

| Methods | Recall (%) |
|---|---|
| DWT+RNN [13] | 83.25 |

| | |
|---|---|
| GA-XGBoost [14] | 89.15 |
| CNN+RNN [15] | 71.35 |
| H-RSVM [Proposed] | 94.85 |

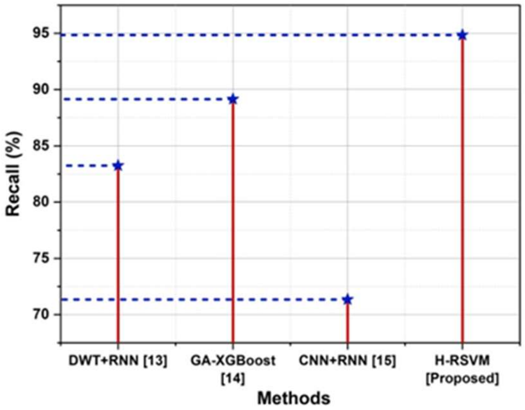

Fig.4 Contrast of recall with conventional and suggested approach

The F1 score, sometimes referred to as the F1 measure or F1 value, is a well-liked metric in machine learning and statistics for evaluating the efficacy of classification models, particularly in binary classification tasks. It provides a balance between these two crucial assessment criteria by combining accuracy and recall into a single scalar number.

$$F1 - score = Two * (P * R) / (P + R) \quad (16)$$

Where P is precision, which is the ratio of correct positive forecasts to all positive guesses made by the model. It gauges how well forecasts turn out. R is recall, also known as sensitivity or the true positive rate; it is the ratio of real positive predictions to all genuine positives in the dataset. It examines the model's capability to accurately detect every good case. In this article, we achieved an F1-Score of 94.85 % using the H-RSVM method. While other existing methods achieved F1-Score, such as DWT and RNN achieved 83.25%, GA-XGBoost achieved 89.15%, CNN and RNN achieved 71.35%.

Fig. 5 denotes the contrast of the f1-score with the conventional and suggested technique. Table IV depicts the numerical outcomes of the f1-score.

**TABLE IV**
**COMPARISON OF F1-SCORE**

| Dataset | F1-Score (%) | | | |
|---|---|---|---|---|
| | DWT+RNN [13] | GA-XGBoost [14] | CNN+RNN [15] | H-RSVM [Proposed] |
| 1 | 91 | 94 | 92 | 95 |
| 2 | 83 | 93 | 90 | 95 |

| | | | | |
|---|---|---|---|---|
| 3 | 89 | 95 | 94 | 96 |
| 4 | 82 | 89 | 92 | 94 |
| 5 | 92 | 95 | 90 | 97.38 |

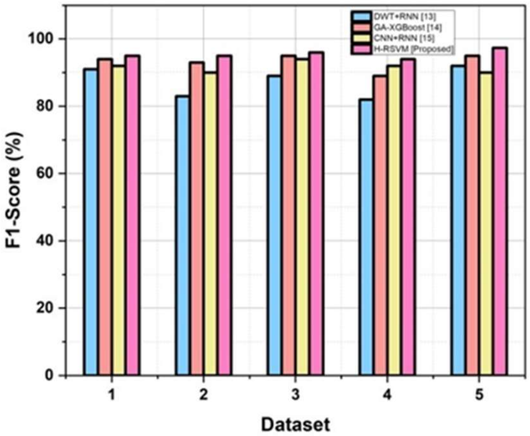

Fig.5 Contrast of f1 score with conventional and suggested approach

The accuracy of a predictive model or the mistakes in a collection of predictions or estimations is often measured using RMSE metric in statistics and data analysis. It is notably prevalent in the signal processing, regression analysis, and machine learning domains. The overall magnitude of the differences between the actual ones recorded and the estimated ones is measured by the RMSE. Mathematically, The square root of the average of the squared discrepancies between the anticipated ones (commonly indicated as $z'$) and the real observed ones (typically denoted as z) is known as RMSE:

$$RMSE = sqrt(\Sigma(z - z' )^2 / m) \qquad (17)$$

Where:
z: Real observed ones
$z'$: Anticipated or estimated ones
Σ: Summation notation (summing over all data points)
m: Total number of data points or observations

A smaller RMSE value indicates a better fit between the model and the data. A larger RMSE number, on the other hand, indicates bigger errors in the model's predictions and a possible poor fit with the data. In this article, we achieved an RMSE of 0.1 using the H-RSVM method. While other existing methods achieved RMSE, such as DWT and RNN achieved 0.9, GA-XGBoost achieved 0.7, CNN and RNN achieved 0.8. Fig. 6 denotes the contrast of the RMSE with the conventional and suggested technique. Table V depicts the numerical outcomes of the RMSE.

TABLE V
COMPARISON OF RMSE

| Methods | RMSE |
|---|---|
| DWT+RNN [13] | 0.4 |
| GA-XGBoost [14] | 0.7 |
| CNN+RNN [15] | 0.8 |
| H-RSVM [Proposed] | 0.1 |

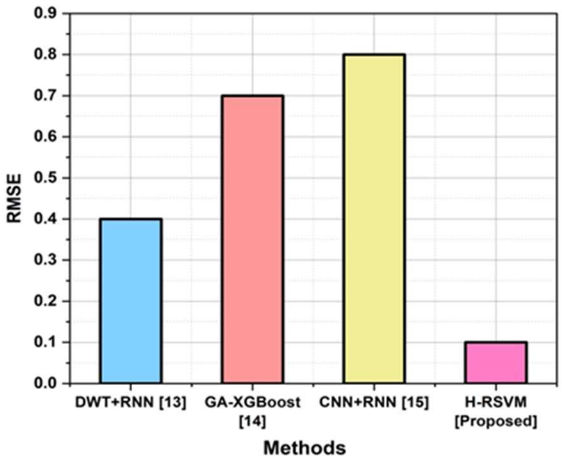

Fig.6 Contrast of RMSE with conventional and suggested approach

The accuracy of a predictive model or the mistakes in a group of forecasts or estimations is often assessed using the Mean Absolute Percentage Error (MAPE) statistic. It is especially useful for calculating how well a model's predictions match the actual results or their relative validity. MAPE is used to compute the average percentage variance between the actual observed values and the projected values.

Mathematically, MAPE is defined as:

$$MAPE = (1/m) * \Sigma(|(z - z') / z|) * 100\% \qquad (18)$$

Where:
z: Actual observed values
ź: Predicted or estimated values
Σ: Summation notation (summing over all data points)
m: Total number of data points or observations

A smaller MAPE means that the model and the data are more closely matched since the model's average predictions are closer to the actual values. In contrast, a larger MAPE score indicates more percentage prediction mistakes, which may suggest a less accurate model. In this article, we achieved a MAPE of 0.2 using the H-RSVM method. While other existing methods achieved MAPE, such as DWT and RNN achieved 0.6, GA-XGBoost achieved 0.3, CNN and RNN achieved 0.5. Fig. 7 denotes the contrast of the MAPE with the conventional and suggested technique. Table VI depicts the numerical outcomes of the MAPE.

TABLE VI
COMPARISON OF MAPE

| Methods | MAPE |
|---|---|
| DWT+RNN [13] | 0.6 |
| GA-XGBoost [14] | 0.3 |
| CNN+RNN [15] | 0.5 |
| H-RSVM [Proposed] | 0.2 |

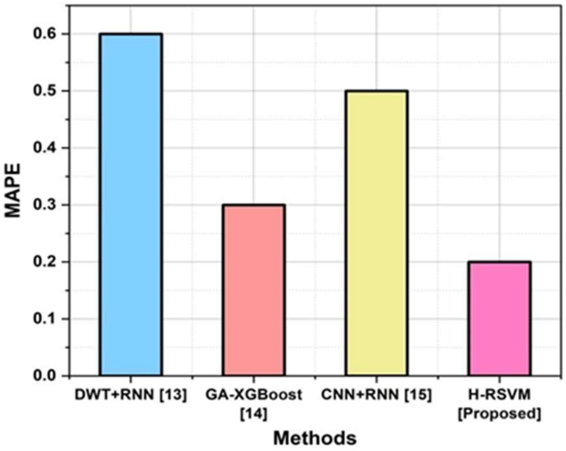

Fig.7 Contrast of MAPE with conventional and suggested approach

## Discussion

Predicting daily market trends in investments is a complex and challenging task due to the multitude of factors that can influence financial markets. Making more precise predictions may be accomplished by using an integrated machine learning strategy, such as the H-RSVM. It's crucial to comprehend the fundamentals of Support Vector Machines (SVMs) before diving into the H-RSVM. SVMs are supervised learning models used in regression and classification. Numerous variables, many of which are unanticipated and may cause random price changes, have an impact on daily market movements. These variables include geopolitical developments, market mood, and news events. It is challenging to effectively anticipate short-term trends on a regular basis due to this unpredictability. Investors may get distracted from their long-term financial goals if they try to forecast everyday developments. It may result in rash choices and a lack of discipline in following a well-planned investing strategy. Daily market patterns may also be influenced by traders' and investors' emotions, prejudices, and cognitive limitations. These characteristics are sometimes challenging to measure and forecast.

## V. CONCLUSION

A market where people and organizations may purchase and sell ownership interests in publicly listed corporations is referred to as the stock market. Prices in stock markets may change quickly because they are naturally volatile. Investors may experience uncertainty as a result, which may result in substantial profits or losses over a short period of time. Some equities could not have enough buyers and sellers, which would indicate a lack of liquidity. This may make it more difficult to execute big deals and cause greater bid-ask spreads. Cyber attacks may halt trade and jeopardize investor data security at stock exchanges and other financial organizations. To overcome this issue, we suggested an H-RSVM approach, which provided a 97.38% f1score, 94.85% recall, 96% precision, and 95% accuracy, which provides superior performance than other traditional methods. In the future, to increase forecast accuracy, a hyper-parameter optimization strategy might be worthwhile.

## References

[1] D.P. Gandhmal, and K. Kumar, "Systematic analysis and review of stock market prediction techniques," Computer Science Review, 34, p.100190, 2019.

[2] N. Rouf, M.B. Malik, T.Arif, S. Sharma, S.S ingh, S. Aich, and H.C. Kim, "Stock market prediction using machine learning techniques: a decade survey on methodologies, recent developments, and future directions" Electronics, 10(21), p.2717, 2012.

[3] D. Wei, "Prediction of stock price based on LSTM neural network," In 2019 international conference on artificial intelligence and advanced manufacturing (AIAM), pp. 544-547, IEEE, 2019, October.

[4] M. Nabipour, P. Nayyeri, H. Jabani., S. Shahab and A. Mosavi, "Predicting stock market trends using machine learning and deep learning algorithms via continuous and binary data; a comparative analysis," IEEE Access, 8, pp.150199-150212, 2020.

[5] H.N. Bhandari, B. Rimal, N.R. Pokhrel, R. Rimal, K.R. Dahal and R.K. Khatri, "Predicting stock market index using LSTM," Machine Learning with Applications, 9, p.100320, 2020.

[6] X. Zhong and D. Enke, "Predicting the daily return direction of the stock market using hybrid machine learning algorithms," Financial Innovation, 5(1), pp.1-20, 2019.

[7] H. Liu and Z. Long," An improved deep learning model for predicting stock market price time series," Digital Signal Processing, 102, p.102741, 2020.

[8] J. Long, Z. Chen, W. He, T. Wu and J. Ren, "An integrated framework of deep learning and knowledge graph for prediction of stock price trend: An application in Chinese stock exchange market" Applied Soft Computing, 91, p.106205, 2020.

[9] K. Zhan, G. Zhong, J. Dong, S. Wang and Y. Wang, "Stock market prediction based on the generative adversarial network," Procedia computer science, 147, pp.400-406, 2019.

[10] J. Patel, S. Shah, P. Thakkar and K. Kotecha, "Predicting stock and stock price index movement using trend deterministic data preparation and machine learning techniques," Expert systems with applications, 42(1), pp.259-268, 2015.

[11] M. Nabipour, P. Nayyeri, H. Jabani, A. Mosavi and E. Salwana, “Deep learning for stock market prediction," Entropy, 22(8), p.840, 2020.

[12] D. Cheng, F. Yang, S. Xiang, and J.Liu, “Financial time series forecasting with multi-modality graph neural network,” Pattern Recognition, 121, p.108218, 2022.

[13] M. Jarrah and N. Salim, “A recurrent neural network and a discrete wavelet transform to predict the Saudi stock price trends,” International Journal of Advanced Computer Science and Applications, 10(4), 2019.

[14] K.K. Yun, S.W. Yoon, and D. Won, “Prediction of stock price direction using a hybrid GA-XGBoost algorithm with a three-stage feature engineering process,” Expert Systems with Applications, 186, p.115716, 2021.

[15] M. Zulqarnain, R. Ghazali, M.G. Ghouse, Y.M.M. Hassim, and I. Javid, “Predicting financial prices of stock market using recurrent convolutional neural networks,” International Journal of Intelligent Systems and Applications (IJISA), 12(6), pp.21-32, 2020.