[1]**Anjali Verma**

[2]**L. K. Vishwamitra**

A Stochastic Regression and Sentiment Analysis-Based Approach for Forecasting Trends in the Stock Market

JES

Journal of Electrical Systems

Abstract: - Stochastic regression problems especially applied to time series forecasting problems often encounter the challenge of volatility and unpredictable seasonality in datasets. One such application happens to forecasting the movement of stock markets globally. Stock data across the world exhibits an intrinsic baseline noise, which often can't be correlated to previous temporal patterns. Rather, such divergences are an effect of multiple non-numeric variables which govern stock movement such as political scenarios, trade wars, imminent economic waxing and waning trends to name a few. However, it is essential to incorporate these factors while training a model so as to design a strongly sentient and anticipatory regression algorithm which is both pervasive and robust, encompassing as many numeric and non-numeric factors as possible. While conventional machine learning and deep learning approaches tend to leverage tested model architectures, they often miss out on the optimization of both training data and training method to yield potentially high forecasting accuracy. The paper combines stochastic regression in terms of momentum based training and data optimization which is shown to exhibit considerable lower forecasting error compared to existing approaches. To encompass a wider feature space, sentiment analysis has been incorporated to correlate stock movements with public opinions. The proposed approach has been tested on a multitude of benchmark datasets, to validate the performance of the approach. The mean absolute percentage error (MAPE) and regression values have been selected as primary metrics for the performance evaluation. The proposed approach attains a mean MAPE value of 3.22%. The analysis of the MAPE and forecasting accuracy w.r.t. existing benchmark approaches proves the improved forecasting performance of the proposed approach over a period of 300 days.

Keywords: Stochastic Regression, Sentiment Analysis, Stock Movement, Data Optimization, Momentum Based Training, Mean Absolute Percentage Error (MAPE)

I. INTRODUCTION

Forecasting the trends of stock markets happens to be one of the most critical applications of time series forecasting problems. The reliance of investment decisions on forecasting trends makes it imperative to design models which attain low forecasting error, while mitigating the noisy nature of stock data. The movement patterns of stock markets worldwide tend to be intrinsically noisy and volatile along with varying fluctuations that can be attributed to many factors that impact prices of stocks, indirectly or directly (Xi et al., 2014, [1]). The variable and fluctuating nature of stock market prices happens to be one of most complex challenges pertaining to stochastic regression applied to stock data. Also, the challenge remains in quantification of the influential factors that affect the stock prices variability. The training of the data driven model needs to be robust incorporating lack of seasonality and randomness in the data. The stock markets and trading scenarios keep undergoing volatile situations which are often non-numeric in nature making is extremely challenging to quantify them (Chatzis et al., 2018, [2]). There are major macro economical causes and factors that can have a huge influence on the stock market price volatility, as observed from existing literature in the field. (Pane et al, 2022, [3]) highlights the impact of Covid 19 pandemic on the forecasting performance of the machine learning backed methods. The advent of the pandemic had an immense effect on overall global economy along with the major economical upheavals. The market prices saw major fluctuations around this period along with the economic slowdown being a natural repercussion (Ghasemieh et al., 2022, [4]). Political instability and turmoil with war like events further escalated the economic disruption and it was found to influence the sentiments of the common people. This, in turn created a biased opinion towards trading in share markets thereby having a major impact in the overall stock market performance. (Chatziantoniou et al, 2022, [5]).

The varying nature of the stock markets coupled with the changing sentiments of the public poses a serious challenge in achieving high accuracy in prediction of stock prices based on sentiment evaluation (Sarvanos et al., 2023 [6]. There arises the necessity of substantial amount of research in training of the natural language processing

[1] *Corresponding author: Research Scholar, Oriental University, Indore, India; anjali.phdcse@orientaluniversity.in

[2] Professor, Oriental University, Indore, India; lkvishwamitra@gmail.com

(NLP) based models for forecasting operational stock movement (Choi et al., 2023[7]). Another critical factor that greatly impacts the functioning and performance of the global share market is the state and performance of the markets opening and closing a day prior or opening early on the day due to different time zones in various parts of the world (Syed, 2021, [8]). An evaluation on the share market in China was done to assess the impact and the performance based behavior of the market prices relative to the prices of the previous day's closing prices (Wang et al., 2020, [9]). The sentiment of the common public also varies in times of major events like pandemics and war like turmoil. The opinions vary from person to person and it also tends to very volatile regarding a lot of socio-economical contexts. Without loss of generality, its can be stated that while incorporating non-quantifiable data can enhance the robustness of the system, it can be prone to strong biases among the sampled data (Mao et al., 2023, [10]).

II. RELATED WORK

This section presents an overview of the existing research in the domain citing the salient attributes of each approach. (Karmiani et al., 2019, [11]) used the Kalman filter along with regression algorithms such as back-propagation (BP), long short term memory (LSTM) and support vector machine (SVM) for forecasting stock trends. It was shown that the Kalman filter, based on linear-quadratic fitting effectively filtered baseline noise from the data set prior training. (Jagwani et al., 2018 [12]) incorporated the use of ARIMA model technique to understand the functioning of trends of stock market. This method worked upon the safety of stock market investments. The research looked for the safe windows for doing investments that could considerably minimize losses. The accuracy obtained was fair. (Almaafi et al., 2023 [13]) implemented the xboost technology alongside Arima that outperformed it on individual capacity. It further increased the accuracy substantially. (Subakkar et al., 2023, [14]).(Jena et al., 2023 [15]) put forth the use of sentiment based analysis along with the use of LSTM model. This worked very well to drastically improve the overall accuracy. The twitter based analysis was done for stock price prediction and the results outperformed the other approaches used earlier. (Juairiah, et al., 2022 [16]) made use of multi time series methodology for share market and trading forecasting. It gave improved results in terms of accuracy and the error percentage was also low. While sentiment based evaluation an provide necessary insights and metrics of stock market price analysis but it can also lead to a biased decision making based on individual preferences and perceptions. (Soun et al., 2022 [17]) This can lead to decline in accuracy. There can bias towards certain types of stocks and this can lead to an inaccurate way of assessment. So, proper methods can be used to filter out the biased data and extracting the intended information for stock prices evaluation. Other than incorporating sentiment analysis, another effective approach in correlating the feature vectors over specified intervals of time to test for causality. This approach is implemented by causality testing through effective transfer entropy (ETE), proposed by (Kim et al., 2020, [18]). As unbalanced datasets are often prevalent in regression problems, out of sample forecasts based on the generalized autoregressive conditional heteroscedasticity (GARCH) model was proposed which achieves an MAE value of 5.67% (Sen et al., 2021 [19]). It is often found that optimizing the weight update, for regression models such as multi-layer perceptron (MLP) or neural networks attain higher accuracy (Ecer et al., 2020, [20]). This approach utilized the genetic algorithm (GA) and particle swarm optimization (PSO) techniques for optimizing weight updated and achieved MAPE values of 28.16%, and 29.09% respectively. Focus has also been on design of ensemble models combining two or more models to enhance forecasting accuracy. (Lu et al., 2020 [21]) combined convolution neural networks and long short term memory to create a CNN-LSTM ensemble which achieved an MAPE of 27.564%. Optimizing the training of deep neural networks such as the LSTM was proposed by (Rokhsatyazdi et al., 2020 [22]). This paper used the ant colony optimization for adjusting the LSTM weights and achieved an MAPE of 11.67%. As imbalanced datasets or lack of correlated feature data might be encountered, alleviating such issues poses a non-trivial challenge. Employing the Generative adversarial network (GAN) with LSTM was proposed by (Kumar et al., 2022, [23]), which achieved a best case accuracy of 64.58%.

III. METHODOLOGY

The methodology developed in this paper tried to address the existing research gap in the domain. While several approaches revolve around the application of existing benchmark models such as ARIMA, GARCH. SVM, MLP, LSTM, bi-LSTM, CNN-LSTM etc., the approaches miss out on the data preparation and optimization part (Sun et al., 2020, [24]). While leveraging existing benchmark models vanquishes the need for separate data-processing and

feature engineering, it misses out on the control over training features. Moreover, it leaves a leeway for the vanishing gradient and overfitting problems, commonly associated with benchmark models (Tan et al., 2019, [25]).

A.   Data Filtration

Stock data, typically exhibits a baseline noise nature making accurate forecasting challenging (Idrees et al., 2019, [2019]). In addition to that, incorporating sentiment analysis as additional feature vectors may contain large biases, arising out of the sample size of the opinion mined data (Asyrofi et al., 2022, [27]). Thus, to ascertain accurate pattern recognition, it is critically important to pre-process the raw data prior to applying it the machine learning model. Several approaches have been explored for noise filtration such as averaging filters as well as filtration in the transform domain. We investigate each of them subsequently. The averaging based filtration tools are often based on the assumption that the data would be centred around the first/second moment of the random variable (Devore et al., 2021, [28]).
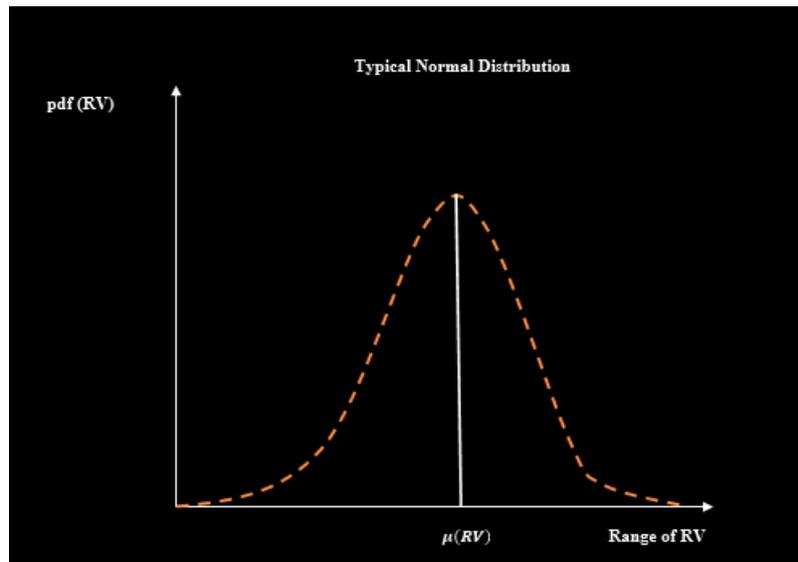

Fig.1 A typical normal distribution

Figure 1 depicts a typical normal distribution, in which the first moment or expectation mean has been chosen as $\boldsymbol{\mu = 60,}$ where,

$$\mu = \frac{1}{N}\sum_{i}^{N} x_i f_i \tag{1}$$

Here,

x = random variable (RV).

f = frequency distribution.

μ = first moment or expectation of the RV.

N = number of samples in the distribution.

While the mean can be considered to be the baseline value, it is significantly governed by the sample values, which tend to move the mean. A rather, more effective approach is using the median as the effective central value, and then filtering out the samples, based on a windowed median filter given by (Batson et al., 2019, [29]):

$$y_{MF}(y) = median(y[i:i+l]) \tag{2}$$

Here,

$y_{MF}(i)$ = median filtering operation.

i = index of the variable.

l = median window.

While the median filtering based approach is computationally efficient, it can be affected by strong bias in the sample space, leading to positive or negative skewness (Bono et al., 2019 [30]).
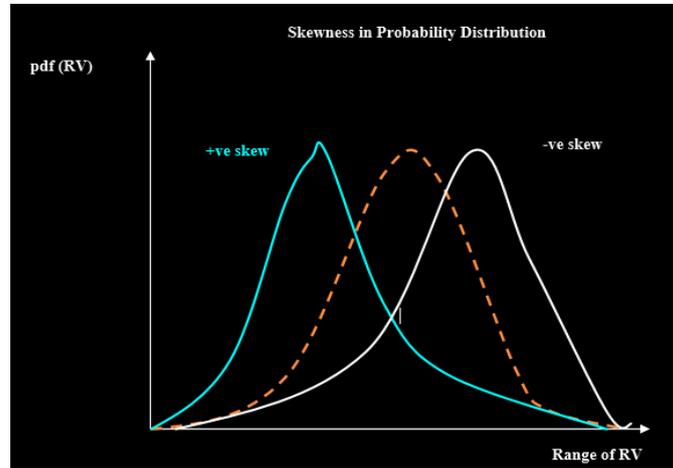

Fig.2 Skewness in probability distribution.

Figure 2 depicts the skewness (both positive and negative) for the distribution. Presence of strong biases in the data may often lead to high skewness in the distribution which if not corrected may lead to inaccuracies in pattern recognition. The skewness of a random variable is defined as:

$$s = \sqrt[3]{\frac{1}{N}\sum_{i,j}^{N}(f_i - \mu)^3} \tag{3}$$

Here,

s = skewness.

f = frequency distribution.

μ = first moment or expectation of the RV.

N = number of samples in the distribution.

As stochastic filtering of the data is heavily affected by biases, a more effective approach turns out to be the data filtration in the transform domain. The main ides of the approach is to transform the index of the variable to a separate variable, which can allow separation of the actual information and the noisy part (Xu et al., 2020 [31]). Any transform T is expressed as:

$$X_k(z) \overset{T(x)}{\Longrightarrow} x_i(t) \tag{4}$$

Here,

x = RV in 't-domain'.

X = data in the 'z-domain'.

T = transform.

i = number of variables in the 't-domain'.

k = number of variables in the 'z-domain'.

Typically, the filtration in the transform domain is done by transforming the data to the frequency domain for multi-resolution frequency analysis. The Fourier Transform is the fundamental transform which can present the frequency domain analysis of data given by (Hou et al., 2022 [32]):

$$X(f) = \int_{-\infty}^{+\infty} x(t)e^{-j2\pi ft}dt \qquad (5)$$

Here,

$X(f)$ = RV in 'f domain'.

$x(t)$ = data in 't-domain'.

$e^{-j2\pi ft}$ = exponential kernel.

j = imaginary number$\sqrt{-1}$.

Several variants of the Fourier Transform are typically used for the data filtration process, where the transform essentially separate the low frequency component (f_L) and high frequency components (f_H). It is observed that the (f_L) components contain the most information in the data while the (f_H) components contain the noisy part, which is to be removed. An important aspect of the filtration process is the choice of the kernel of the transform (Osipenko, 2021 [33]). The Fourier Transform and its derivatives such as the Laplace Transform (LT), the Discrete Fourier Transform (DFT), Fast Fourier Transform (FFT), Z-Transform, Discrete Cosine Transform (DCT) and Discrete Sine Transform (DST), rely on the complex exponential Kernel function, given by (Falsone at l., 2021, [34]):

$$e^{-(\sigma+jf)} = e^{\sigma}(cosf - jsinf) \qquad (6)$$

Here,

$e^{\sigma+jf}$ = complex exponential.

e^{σ} = constant amplitude.

f = frequency.

The limitation of the Fourier derivatives stem from the fact that all of the transforms have smooth kernel functions in the form of a sine or cosine function. Thus, they are well suited for the multi-resolution analysis of smooth RVs. In case of datasets, with abrupt fluctuations, these methods fail to render accurate multi-resolution analysis (Heurtier, 2019 [35]). A summary of the Fourier derivatives along with their kernels is presented in table 1.

Table 1. Fourier Derivative Transforms

S.No.	Transform	Kernel	Function Analysis
1.	Fourier Transform	$e^{-j2\pi ft}$	Bounded
2.	Laplace Transform	e^{-st}	Unbounded
3.	Discrete Fourier Transform (DFT)	$e^{-j2\pi kn}$	Bounded
4.	Fast Fourier Transform (FFT)	$e^{-j\frac{2\pi kn}{N}}$	Bounded
5.	Z-Transform (ZT)	z^{-n}	Unbounded
6.	Discrete Cosine Transform (DCT)	$cos(\dfrac{(2x+1)jx}{2N}$	Bounded
7.	Discrete Sine Transform (DST)	$sin(\dfrac{\pi nk}{N+1})$	Bounded

The bounded/unbounded nature of the transform refers to the capability of the transform to handle asymptotically increasing (both positive and negative) functions. This is however, a generalization in practical cases, as stock data would invariably be bounded in nature. The limitation of the Fourier derivatives can be overcome employing

the Wavelet Transform whose Kernel is an abruptly changing function (Liu et al., 2019 [36]). The wavelet transform separates the low frequency and high frequency components as the approximate co-efficient(C_A) and detailed co-efficient(C_D) of the transform (Hau et al., 2022, [37]). Generally, detailed co-efficient(C_D) and a low frequency resolution component termed as the detailed co-efficient(C_A) (Hajiabotorabi et al., 2023 [38]).The DWT of any function $z(x)$ is computed as:

$$Z(x, \Theta_0^n, i\Theta_0^n) = \delta_0^{n\frac{-1}{2}} \sum_i z(x) K^* \left[\frac{n - i\Theta_0^n}{\Theta_0^n} \right] \tag{7}$$

Here,

$z(x)$ = time series vector

K^* = wavelet kernel

Z = data in the transform domain or DWT domain

Θ_0^n = scaling operation

$i\Theta_0^n$ = shifting operations

δ_0^n = dilation constant of the transform

Retaining the low frequency component(C_A) while discarding the higher frequency component(C_D) for a number of iterations helps in removal of the baseline noise of the system (Diker et al., 2018 [39]).

B.   Data Optimization

Typically, stochastic regression with large datasets, can be prone to overfitting and sporadic vanishing gradients. This occurs due to a multitude of often repeated and redundant data samples in the training data. One of the most effective data optimization and dimensional reduction techniques happens to be the principal component analysis (PCA). The basic idea of the algorithms is the maximization of the variance (σ^2) among data samples, while minimizing the correlation. The PCA employs a search for the principal components in an n-dimensional feature space given by (Sorzano et al., 2014 [40[):

$$x \epsilon \mathbb{R}^n \; \forall n \tag{8}$$

Here,

$\mathbb{R}$ = feature space.

n = order of the feature space.

The idea of the approach is to map the values of $\mathbb{R}$ along different, mutually orthogonal feature directions, satisfying the condition (Reddy et al. 2020, [41]).

$$\sigma^2 = \sum_{i=1}^{n} var(\chi_i) \tag{9}$$

The variance can be maximized along orthogonal directions, χ after which the component with maximum variance is chosen to be the PCA-transformed representation of the data.
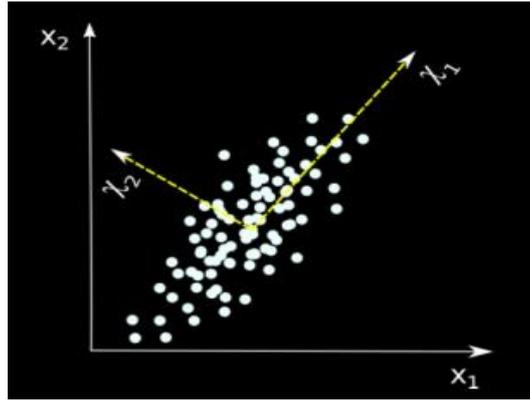
Fig.3 Illustration of PCA-decomposition

Figure 3 illustrates the process of PCA-decomposition of the data samples along $\chi_1\ and\ \chi_2$, considering only 2-feature dimensions for the sake of simplicity. The idea can be extended to a n dimensional feature space. It can be observed that the data can be mapped along $\chi_1\ and\ \chi_2$. However, the data samples depict maximum variance along χ_1, which is adjudged the direction of the principal component.

The PCA matrix M, for the $l\ Xl$ PCA base matrix and the data matrix D is computed as (Yang et al., 2017 [42]):

$$M = B_{lxl}D \qquad (10)$$

Here,

M= representation matrix

B = PCA basis matrix

D= original data matrix

l= dimensions of the PCA basis matrix

The PCA base matrix exhibits orthogonality in $l\ dinensinal$ vector space (Hu et al., 2016 [43]). The cost function to be maximized can be expressed as:

$$Maximize\ (V(l^T f)) \qquad (11)$$

Here,

V= variance

In order to minimize the co-variance among the lxl orthonormal values, the approach used is to diagonalize the co-variance matrix of M, given by:

$$M_D = \frac{1}{l-1}(B^{-1}B)D(B^{-1}B) \qquad (12)$$

Here,

M_D= diagonalized re-represented matrix

D = diagonal matrix

In case, B exhibits orthogonality, the following condition holds true,

$$B^T = B^{-1} \qquad (13)$$

In case the above condition holds true, the diagonalization boils down to:

$$M_D = \frac{1}{l-1}(D) \tag{14}$$

One of the most effective ways to obtain the diagonalized version of M is to compute the PCA using the singular value decomposition (SVD), constrained to the condition (Winursito et al., 2018 [44]):

$$f \gg n \, \nabla n, f \tag{15}$$

The cost function for the PCA decomposition is given by (Seghouane et al., 2019 [45]):

$$J = \overline{\llbracket x - WW^T x \rrbracket} \tag{16}$$

Here,

$W = n$ orthogonal directions of the PCA decomposition.

The dimensionally reduced data, is subsequently used to compute a moving window of recent samples which is to be later added to the feature training vector. The idea of the approach is to capture the recent trends in the data along with the entirety of pattern recognition. This is computed as:

$$X_{split} = X_{LV}: f(X_{t-1, \ldots \ldots} X_{t-l}), X_{HV}: f(X_{t-l}, X_{t-h, \ldots \ldots} X_{t-kh}) \tag{17}$$

Here,

$X_{LV}=$ low volatility (LV) component

$X_{HV}=$ high volatility (HV) component

$X_{split}=$ split training vector in terms of a two components LV and HV

$f=$ function

The filtered samples contain the **LV** components as well as the **HV** components corresponding to the actual information and high frequency noise components respectively. A moving window of $(\boldsymbol{n}:\boldsymbol{n}+\boldsymbol{k})$ samples is computed and expressed as:

$$X_{LV} = mean(g(\textstyle\sum X_{t-kh}), \epsilon_t) \tag{18}$$

Here,

$g =$ non-linear function

$\epsilon_t=$ high volatility error function

It is assumed that the mean would capture the recent stochastic features of the wide sense stationary random process (Guo et al., 2016[46]). It is assumed that the addition of baseline random noise to the data would still keep the nature of the original data as wide sense stationary (Girault, 2015, [47]).

3.3 Training

While several benchmark models such as RNN, LSTM, bi-LSTM and CNN-LSTM ensemble employ the back propagation based training in the fully connected layer, deep nets are often susceptible to a non-optimal acceleration along the normal direction of convergence (Oymak et al., 2020 [48]). A more optimal approach would be increasing the acceleration along convergence, while simultaneously minimizing the normal acceleration component, thereby still keeping the overall acceleration magnitude constant (Chnaga et al., 2022 [49]). The momentum based training considers the acceleration normal to convergence to manifest itself in an overshoot/undershoot or swing in the cost function, which needs to be minimized for optimal training. The learning rate is adjusted as (Cutkosky et al., 2020 [50]):

$$\mu_{k+1} = \mu_k - \gamma \frac{\partial f(w_k)}{\partial \mu_k} \tag{19}$$

The aim of the approach is thus to minimize both the cost function and overshoot or swing (M)(Zheng et al., 2018, [51]). Let us define the swing for the cost function as:

$$S_{cost} = \frac{\varepsilon_{inst} - \varepsilon_{opt}}{\varepsilon_{opt}} \qquad (20)$$

Here,

S_{cost} = swing of the cost function

ε_{inst} = instantaneous value of the cost function

ε_{opt} = optimal values of the cost function at convergence

The value of ε_{opt} should be computed at convergence subsequent to cross validation. The swing parameter is to be used to truncate the training process based on the gradient descent with momentum (Liu et al., 2020 [52]). The weights of the neural network model are to be updated constrained to the condition:

$$W \triangleq argmin(\frac{\varepsilon_{inst} - \varepsilon_{opt}}{\varepsilon_{opt}}) \qquad (21)$$

The least squares (LS) optimization for the cost function is given by: (Gan et al., 2018 [53]):

$$C = \frac{1}{n}\sum_{i=1}^{n}(p_i - t_i)^2 \qquad (22)$$

Here,

C = function associated with least squares minimization (LSE)

n = number of samples

p = predicted value

t = target

The conventional gradient descent for a cost function C can be implemented using the following relation (Vu et al., 2022 [54]):

$$w_{k+1} = w_k - \mu\nabla C(w_k) \qquad (23)$$

Here,

k = iteration number

$k + 1$ = subsequent iteration

k = present iteration

w = weight

μ = learning step size

The conventional gradient descent tries to update the weight matrix W so as to maximize the negative error gradient$(g = -\frac{\partial e}{\partial w})$. This allows to minimize the cost function $C(w_k)$ over a range of iterations (Baden et al., 2018 [55]). The weight update for the gradient descent with momentum based approach is given by (Ilboudo et al., 2022 [56]):

$$w_{K+1} = w_k - \mu\beta(k) \qquad (24)$$

Where,

$$\beta(k) = \alpha\beta(k-1) + \nabla C(w_k) \qquad (25)$$

Here,

$\beta(k)$ = momentum factor

α = momentum attenuation factor

$C(w_k)$ = cost function

μ = learning step size

For the purpose of convergence, the learning rate is constrained to (Franceschi et al., 2017, [57]):

$$-1 < \alpha < 1 \; and \; \mu > 0 \qquad (26)$$

For non-negative α values and small values of μ, the swing S_{cost} needs to be minimized employing the friction parameter $\frac{\varepsilon_{inst} - \varepsilon_{opt}}{\varepsilon_{opt}}$, which meeds to be minimized as:

$$Argmin \left(\frac{\varepsilon_{inst} - \varepsilon_{opt}}{\varepsilon_{opt}} \right) \forall \alpha, \mu \qquad (27)$$

The acceleration along convergence, is thus analogous to minimization of the cost function J, which is in turn accelerated through the momentum factor $\beta(k) = \alpha\beta(k-1) + \nabla C(w_k)$. Adding a customary bias term at node i for the network defined as (Wang et al., 2019 [58]):

$$b_k^i = i - \alpha S^i \qquad (28)$$

Here,

α = learning rate

S = learning sensitivity

The learning sensitivity is critically important as it governs the weight update for the proposed approach while estimating the changes in the learning rate for the model. This typically applicable for an approach with a search for optimal learning step size in each approach as compared to the conventional stochastic gradient descent (SGD) based approach (Haddadpour et al., 2019[59]). The adaptive nature of learning step size, already constrained to ($\alpha \; and \; \pi$) may result in higher computational complexity, bound by both convex a non-convex settings (Li et al., 2020 [60]).

3.4 Combining Opinion Mining

The inclusion of public sentiments is typically challenging to the fact that they are non-numeric in nature. However, sentiment data as an additional parameter with tokenization inputs tries to correlate the non-numeric factors with the feature vector. The sentiment inclusion for the proposed study includes the sentiment vector expressed as:

$$T_{sentiment} = [T_{positive}, T_{neutal}, T_{negative}] \qquad (29)$$

The sentiment training vector is extracted for posts (public sentiments) for the training period of the data (in the training time interval). To add $T_{sentiment}$ to the roaster of the training vector X, polarities $(+1, 0, -1)$ are included as an extra feature. Non-existent values of the sentiment data are substituted with **NaN** (Not a Number) format, to keep training sample size constant across all features. The data is pre-processed by removing special characters such as (*, # etc.) and filtering out repeated words and symbols. A probabilistic approach is employed to classify sentiments into the three categories. The tokenization allows for a three way inclusion of public sentiments to the training vector.

The training rule for the approach is based on the Bayes theorem of conditional probability which is effective for classifying overlapping feature vectors, based on a penalty $\rho = \frac{\mu}{\nu}$. The weights are updated based on the modified regularized cost function (Wongkar et al., 2019, [61]):

$$F(w) = \mu w^T w + v[\frac{1}{n}\sum_{i=1}^{n}(p_i - a_i)^2] \tag{30}$$

If $(\pi \ll v)$: Network error are generally low.

else if $(\pi \geq v)$: Network errors tend to increase, in which case the weight magnitude should be reduced so as to limit errors (Penalty).

This is done be maximizing the weight Posteriori Probability using the Bayes theorem of Conditional Probability as (Tand et al., 2022, [62]):

$$P(\langle w|X \rangle, \mu, v) \tag{31}$$

Aggregation of sentiment polarities in a monotonic price movement interval would aid the training process, subject to the condition that the sentiment polarity correlates with the movement patterns of the stock.

Proposed Algorithm	
Step.1:	Initialize weights w and learning rate μ randomly, set maximum iterations as $Maxitr = 1000, e_{tolerance} = 10^{-6}$
Step.2:	$for \; i = 1: Maxitr$, do
Step.3:	$for \; (k = 1:n),$
Step.4:	Retain (C_A) while discarding (C_D)
Step.5:	$Minimize \; (J = \overline{\llbracket x - WW^T x \rrbracket})$ and compute $M = B_{lxl}D$
Step.6:	Compute $X_{LV} = mean(g(\sum X_{t-kh}), \epsilon_t)$
Step.7:	$if \; i \leq Maxitr \; \&\& \; if \; J \leq e_{tolerance}$
Step.8:	Minimize: $Argmin \left(\frac{\varepsilon_{inst}-\varepsilon_{opt}}{\varepsilon_{opt}}\right) \forall \alpha, \mu$
Step.9:	Compute $w_{K+1} = w_k - \mu\beta(k)$
Step.10:	Compute $\beta(k) = \alpha\beta(k-1) + \nabla C(w_k)$
Step.11:	$else$
Step.12:	Truncate training
Step.13:	$end \; if$
Step.14:	Compute MSE, MAPE, R^2
Step.15:	$end \; for$
Step.16:	$end \; for$

The performance of the proposed model is evaluated in terms of the following parameters:

1) Regression(R^2): The values of R^2 are considered for the training, testing, validation and overall conditions.

2) The mean squared error (MSE):

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(p_i - t_i)^2 \tag{32}$$

3) The mean absolute error (MAE):

$$MAE = \frac{1}{n}\sum_{i=1}^{n}|p_i - t_i| \tag{33}$$

4) Mean Absolute Percentage Error (MAPE):

$$MAPE = \frac{1}{n}\sum_{i=1}^{n}\frac{|p_i-t_i|}{t_i} * 100 \tag{34}$$

Here,

n = number of samples

p = predicted value

t= target

IV. EXPERIMENTAL RESULTS

This work presents an experiment on multiple benchmark (S&P-500) datasets. The day wise opening price, maximum price, minimum price, previous day closing price have been chosen as the independent variables, while the present day closing price has been chosen as the dependent variable. Each of the steps performed during the implementation of the proposed algorithm has been presented sequentially.

Fig. 4 Raw Data: Google Stocks

Figure 4 presents the variation in the stock for Google stocks, for over 3000 samples. The variation and noise content in the data can be observed.
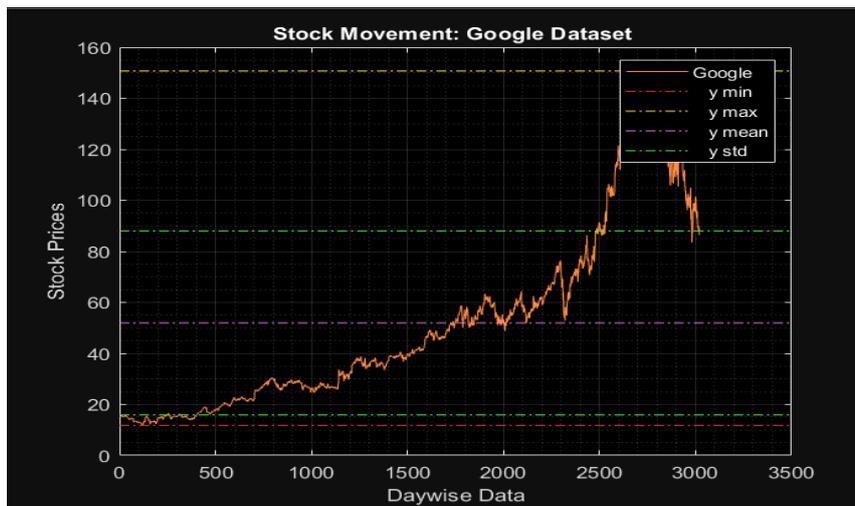
Fig. 5 Stochastic Parameters of raw data: Google

Figure 5 presents the stochastic parameters for the data, such as minimum, maximum, mean and standard deviation. The summary of the stochastic parameters are presented in table 2.

Table 2: Stochastic Parameters of Google Stocks

S.No.	Parameter	Value
1	Stock	Google
2	Minimum	11.8227
3	Maximum	150.7090
4	Mean	51.9638
5	Standard Deviation	36.0943

Table 2 presents the stochastic parameters of the stocks over the entire day wise period. The minim, maximum, mean and standard deviation values are found to be 11.8227, 150.7090, 51.9638 and 36.0943 respectively. The next process is the data filtration employing the DWT.
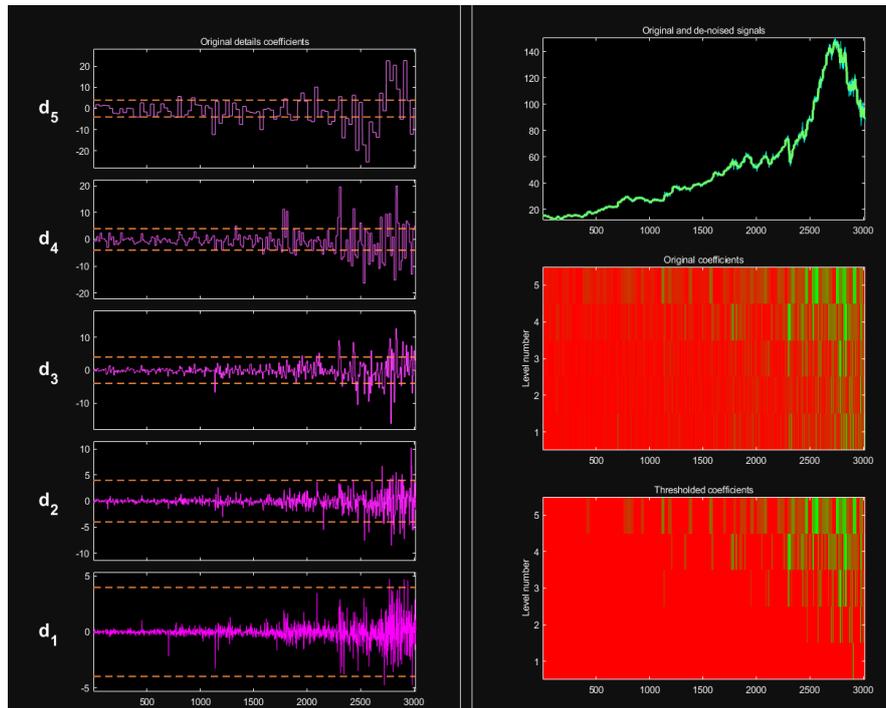

Fig.6 DWT decomposed target variable

Figure 6 presents the DWT decomposed target variable at level 5. The decomposition yields the approximate co-efficient at level 5 along with the detailed co-efficients at each of the 5 levels. It can be observed that the maximum noisy nature of the data is present at level 1 while the least noisy nature of the data is present in level 5. Moreover, decimation can be clearly observed on the data samples as the number of levels increase. It is assumed that the raw data is affected by unscaled white noise characterized by:

$$N(f) = k, \forall f \qquad (35)$$

Here,

$N(f)$ = unscaled white noise.

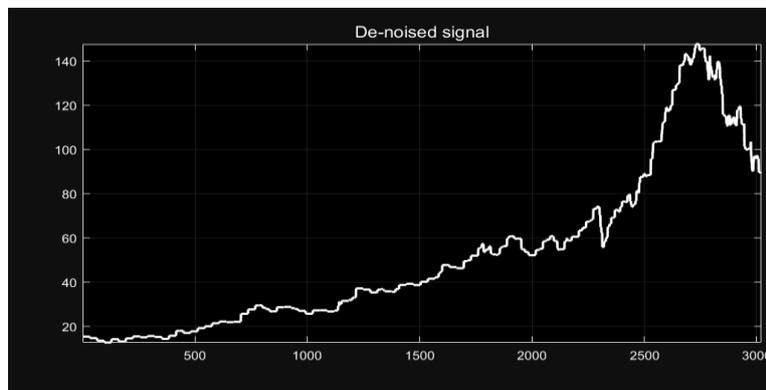k = a constant.

f = frequency distribution.

Fig.7 Filtered data at level 5 of DWT

Figure 7 depicts the filtered data at level 7 of DWT. Smoothening of the data can be observed. The actual filtration removal of baseline noise can be clearly observed through the zoomed in data samples.
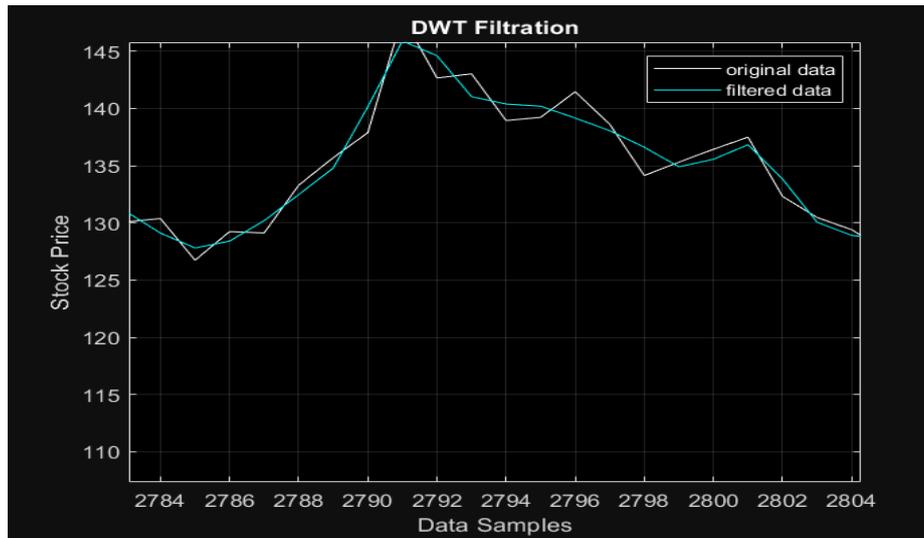

Fig.8 Zoomed in Analysis of Data Filtration

Figure 8 presents the actual and filtered data samples of 20 samples starting from 2784 to 2804. Clearly the filtered data can be seen to have lesser sharp fluctuations.
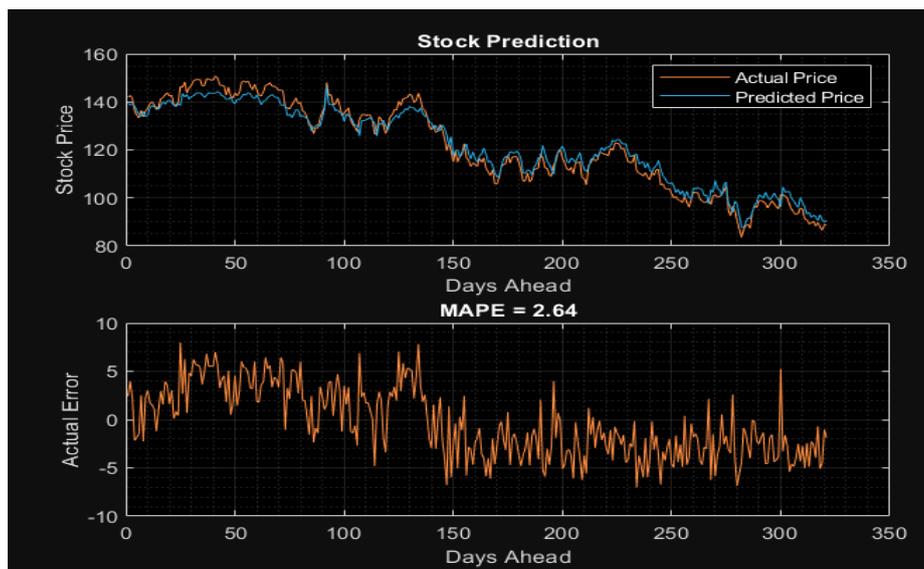

Fig.9 Forecasting Results: Google Stocks

Figure 9 depicts the actual and forecasted values for Google Stocks. It can be observed that the MAPE is 2.64% for the Google Stocks. The accuracy of forecasting is therefore, $100 - MAPE = 97.56\%$. The actual error variation for 320 samples has been shown here (long term case, approximately 1 year period).
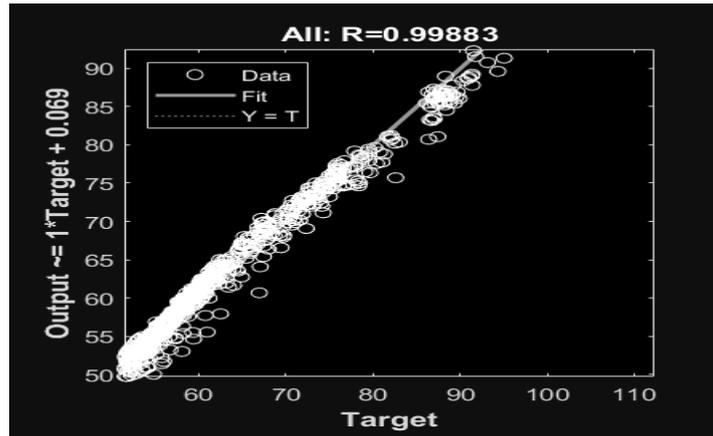
Fig.10 Regression Value: for Google Stock Forecasting

Figure 10 presents the regression (R^2) value for the Google stock forecasting. It can be observed that the proposed forecasting model attains an R^2 value of 0.99883. The MSE of the proposed model in case cross-validation (MSE) is found to be 2.2144 at 1000 iterations with a clear monotonic decrease in the cost function. This illustrates the stability of the proposed approach, in terms of a steady convergence without sudden positive or negative swing in the cost function. The best case gradient at 1000 iterations is found to be 6.7383.

The analysis and results presented for the Google stocks can be applied to other stocks as well, to check the robustness of the proposed approach. In this work, six diverse stocks have been analysed, and the results have been tabulated, for ready reference.

Table 3. Summary of Forecasting Results

S. No.	Stock	MAPE	MSE	R^2
1.	GOOGLE	2.64%	2.2144	0.99883
2.	AMAZON	2.15%	1.926	0.999
3.	TESLA	5.83%	4.77	0.9788
4.	Apple	2.37%	2.062	0.0892
5.	SBI	3.62%	2.844	0.9889
6.	TCS	2.97%	3.085	0.9882

Table 3 presents the forecasting results have been presented for 6 stocks with the MAPE, MSE and R^2 values. Multiple benchmark S & P stocks have been analysed to cross check the performance of the proposed approach.
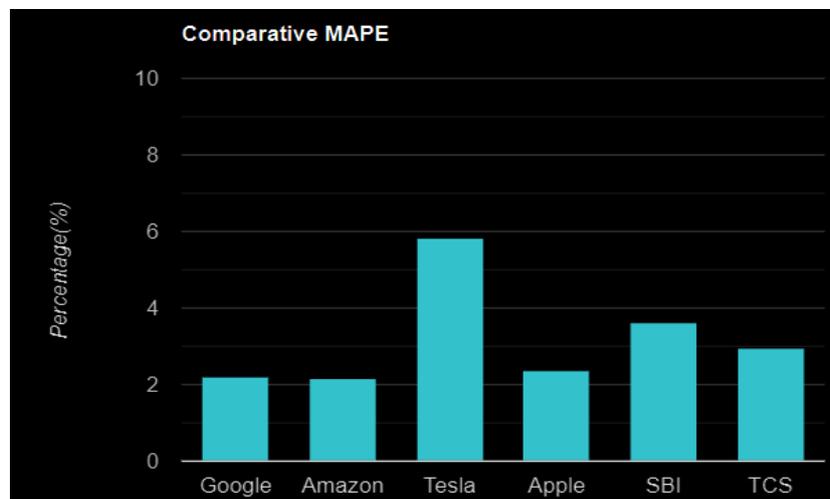

Fig.11 Comparative MAPE for all stocks

Figure 11 presents the comparative MAPE analysis of the six stocks analysed. It can be observed that the best case MAPE of 2.15% is obtained for Amazon stocks and the highest case MAPE of 5.83% is obtained for Tesla stocks. The average MAPE for all the six stocks analysed in this case renders a value of 3.22%. Thus the average forecasting accuracy can be found to be:

$$100 - MAPE = 96.78\%$$

The average MAPE and accuracy% renders an insight into the forecasting capability of the proposed approach across different datasets. To weigh the performance of the proposed approach against contemporary work, a comparison in made in table 4.

Table 4. Comparison with Existing Work

S.No.	Authors	Approach	Performance Metric
1.	Subakkar et al., 2023	ARIMA	MAPE=14% (TESLA)
2.	Jena et al., 2023	Sentiment Analysis and LSTM	Best Case R^2 value of 0.57.
3.	Soun et al., 2022	LSTM with Sentiment Analysis	MAPE of 55.86%
4.	Sen et al., 2021.	GARCH model	MAE of 5.67%
5.	Ecer et al., 2020.	MLP+PSO GA+PSO	MAPE=28.16% MAPE-=29.09%
6.	Lu et al., 2020.	CNN-LSTM Ensemble	MAPE=27.562%
7.	Kim et al., 2020	ETE	MAPE=47%
8.	Rokhsatyazdi et al., 2020	ACO-LSTM	MAPE=11.67%
9.	Kumar et al., 2022.	GAN-LSTM	MAPE=35.42% (best case).
10.	**Proposed Approach**		**MAPE of 3.22%. Accuracy of 96.78%**

The primary metric for comparison used in table 4 are the MAPE and accuracy%. It can be clearly observed that the proposed approach renders improved forecasting performance over a wide range of stock data compared to existing benchmark approaches. The improvement in the results compared to existing work can be attributed to:

1. While deep learning algorithms such as LSTM, bi-LSTM etc. do not need to invest in the separate feature extraction and preparation, it takes away the feature selection control.

2. Exogenous input in the form of a moving average helps extract most recent patterns in the data. This is typically useful in case the moving window size is similar to the number of samples to be forecasted.

3. Combining DWT and PCA provides both data filtration as wells as dimensional optimization making the training process more effective and accurate, while avoiding the possibilities of overfitting and vanishing gradient.

4. Combining a three way opinion mining based approach to incorporate sentiment analysis along with the proposed approach.

V. CONCLUSION

Previous analysis of existing literature emphasizes the importance of accurate stock market forecasting models as the forecasting results are considered prior making large investments. Inaccuracies in forecasting estimates may lead of staggering financial loses, making the stakes high for erroneous forecasting models. The paper tries to combine stochastic regression with sentiment analysis with an aim to bolster pattern recognition based on both previous historical data as well as sentiment polarities of tweets about a particular stock under discussion. Data optimization based on principal component analysis and denoising the raw data using DWT act as the data pre-processing tools removing noise effects and data redundancies respectively. Multiple S & P-500 datasets have

been analyzed using the proposed approach. A comparative analysis based on the MAPE% and R^2 values clearly indicate superior results of the proposed approach compared state of the art algorithms. While successful noise filtration has been done using the DWT, increasing the number of levels results in increasing computational complexity as well as decimation. Hence, maximum overlap based DWT (MODWT) can be explored further. Additional, skewness correction can be employed to alleviate the strong bias in the opinion of the subjects, whose data is inferred as positive, negative and neural values.

REFERENCES

[1] L Xi, H Muzhou, MH Lee, J Li, D Wei, H Hai, Y Wu, "A new constructive neural network method for noise processing and its application on stock market prediction", Applied Soft Computing, Elsevier 2014, vol.15, pp. 57-66.

[2] SP Chatzis, V Siakoulis, A Petropoulos, "Forecasting stock market crisis events using deep and statistical machine learning techniques", Expert Systems with Application, Elsevier, 2018, vol.112, pp. 353-371.

[3] S. F. Pane, Heriyanto, A. G. Putrada, N. Alamsyah and M. N. Fauzan, "The Influence of The COVID-19 Pandemics in Indonesia On Predicting Economic Sectors," 2022 Seventh International Conference on Informatics and Computing (ICIC), Denpasar, Bali, Indonesia, 2022, pp. 1-6.

[4] Ghasemieh and R. Kashef, "A Robust Deep Learning Model for Predicting the Trend of Stock Market Prices During Market Crash Periods," 2022 IEEE International Systems Conference (SysCon), Montreal, QC, Canada, 2022, pp. 1-8.

[5] I Chatziantoniou, C Floros, D Gabauer, "Volatility Contagion between Crude Oil and G7 Stock Markets in the Light of Trade Wars and COVID-19: A TVP-VAR Extended Joint Connectedness Approach", Applications in Energy Finance, Springer, 2022, pp.145–168.

[6] C.Saravanos, A.Kanavos, "Forecasting Stock Market Alternations Using Social Media Sentiment Analysis and Regression Techniques" Artificial Intelligence Applications and Innovations. AIAI 2023 IFIP WG 12.5 International Workshops. AIAI 2023. IFIP Advances in Information and Communication Technology, Springer 2023, vol 677.

[7] J. Choi, S. Yoo, X. Zhou and Y. Kim, "Hybrid Information Mixing Module for Stock Movement Prediction," in IEEE Access, 2023, vol. 11, pp. 28781-28790.

[8] A.A.Syed, "Symmetric and asymmetric influence of macroeconomic variables on stock prices movement: study of Indian stock market." In New Challenges for Future Sustainability and Wellbeing, Emerald Publishing Limited, 2021, pp. 319-339.

[9] S. Wang, Z. Guo. "A study on the co-movement and influencing factors of stock markets between China and the other G20 members." International Journal of Finance & Economics, Wiley, 2020, vol.25, no. 1, pp.43-62.

[10] R. Mao, Q. Liu, K. He, W. Li and E. Cambria, "The Biases of Pre-Trained Language Models: An Empirical Study on Prompt-Based Sentiment Analysis and Emotion Detection," in IEEE Transactions on Affective Computing, 2023, vol. 14, no. 3, pp. 1743-1753.

[11] D. Karmiani, R. Kazi, A. Nambisan, A. Shah and V. Kamble, "Comparison of Predictive Algorithms: Backpropagation, SVM, LSTM and Kalman Filter for Stock Market," 2019 Amity International Conference on Artificial Intelligence (AICAI), Dubai, United Arab Emirates, 2019, pp. 228-234.

[12] J. Jagwani, M. Gupta, H. Sachdeva and A. Singhal, "Stock Price Forecasting Using Data from Yahoo Finance and Analysing Seasonal and Nonseasonal Trend," 2018 Second International Conference on Intelligent Computing and Control Systems (ICICCS), Madurai, India, 2018, pp. 462-467

[13] A Almaafi, S Bajaba, F Alnori, "Stock price prediction using ARIMA versus XGBoost models: the case of the largest telecommunication company in the Middle East", International Journal of Information Technology, Springer 2023, vol.15, pp.1813–1818.

[14] A Subakkar, S Graceline Jasmine, L Jani Anbarasi, J Ganesh, CM Yuktha, "An Analysis on Tesla's Stock Price Forecasting Using ARIMA Model", Proceedings of the International Conference on Cognitive and Intelligent Computing, Springer, 2023, pp 83–89.

[15] PR Jena, R Majhi,"Are Twitter sentiments during COVID-19 pandemic a critical determinant to predict stock market movements? A machine learning approach", Scientific African, Elsevier 2023, vol.18, Art. e01480.

[16] F. Juairiah, M. Mahatabe, H. B. Jamal, A. Shiddika, T. Rouf Shawon and N. Chandra Mandal, "Stock Price Prediction: A Time Series Analysis," 2022 25th International Conference on Computer and Information Technology (ICCIT), IEEE, 2022, pp. 153-158

[17] Y. Soun, J. Yoo, M. Cho, J. Jeon and U. Kang, "Accurate Stock Movement Prediction with Self-supervised Learning from Sparse Noisy Tweets," 2022 IEEE International Conference on Big Data (Big Data), Osaka, Japan, 2022, pp. 1691-1700.

[18] S Kim, S Ku, W Chang, JW Song, "Predicting the Direction of US Stock Prices Using Effective Transfer Entropy and Machine Learning Techniques", IEEE Access 2020, Vol-8, pp: 111660 – 111682.

[19] J. Sen, S. Mehtab and A. Dutta, "Volatility Modeling of Stocks from Selected Sectors of the Indian Economy Using GARCH," 2021 Asian Conference on Innovation in Technology (ASIANCON), IEEE, 2021, pp. 1-9.

[20] F Ecer, S Ardabili, SS Band, A Mosavi, "Training multilayer perceptron with genetic algorithms and particle swarm optimization for modeling stock price index prediction", Entropy, MDPI, 2020, vol. 22, Art. No.1239.

[21] W Lu, J Li, Y Li, A Sun, J Wang, "A CNN-LSTM-based model to forecast stock prices", Complexity, Hindawi-Wiley, 2020, vol.2020, pp.1-10.

[22] E. Rokhsatyazdi, S. Rahnamayan, H. Amirinia and S. Ahmed, "Optimizing LSTM Based Network For Forecasting Stock Market," 2020 IEEE Congress on Evolutionary Computation (CEC), Glasgow, UK, 2020, pp. 1-7.

[23] A Kumar, A Alsadoon, PWC Prasad, S Abdullah, "Generative adversarial network (GAN) and enhanced root mean square error (ERMSE): deep learning for stock price movement prediction", Multimedia Tools and Applications, Springer, 2021, vol.81, pp.3995–4013.

[24] S. Sun, Z. Cao, H. Zhu and J. Zhao, "A Survey of Optimization Methods From a Machine Learning Perspective," in IEEE Transactions on Cybernetics, 2020, vol. 50, no. 8, pp. 3668-3681.

[25] H. H. Tan and K. H. Lim, "Vanishing Gradient Mitigation with Deep Learning Neural Network Optimization," 2019 7th International Conference on Smart Computing & Communications (ICSCC), Sarawak, Malaysia, 2019, pp. 1-4.

[26] M. H. Asyrofi, Z. Yang, I. N. B. Yusuf, H. J. Kang, F. Thung and D. Lo, "BiasFinder: Metamorphic Test Generation to Uncover Bias for Sentiment Analysis Systems," in IEEE Transactions on Software Engineering, vol. 48, no. 12, pp. 5087-5101.

[27] M. H. Asyrofi, Z. Yang, I. N. B. Yusuf, H. J. Kang, F. Thung and D. Lo, "BiasFinder: Metamorphic Test Generation to Uncover Bias for Sentiment Analysis Systems," in IEEE Transactions on Software Engineering, vol. 48, no. 12, pp. 5087-5101.

[28] JL Devore, KN Berk, MA Carlton, "Continuous random variables and probability distributions", Modern Mathematical Statistics with Applications, Springer 2021, pp 189–275.

[29] J Batson, L Royer, "Noise2self: Blind denoising by self-supervision", Proceedings of the 36th International Conference on Machine Learning, Proceedings of MLR Press, 2019, vol.97, pp. 524-533

[30] R Bono, J Arnau, R Alarcón, MJ Blanca, "Bias, precision, and accuracy of skewness and kurtosis estimators for frequently used continuous distributions", Symmerty, MDPI, 2019, vol.12., no.1., pp.1-17.

[31] K Xu, M Qin, F Sun, Y Wang, "Learning in the frequency domain", Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2020, pp. 1740-1749.

[32] C. Hou, G. Liu, Q. Tian, Z. Zhou, L. Hua and Y. Lin, "Multisignal Modulation Classification Using Sliding Window Detection and Complex Convolutional Network in Frequency Domain," in IEEE Internet of Things Journal, 2022, vol. 9, no. 19, pp. 19438-19449.

[33] KY Osipenko, "Inequalities for derivatives with the Fourier transform", Applied and Computational Harmonic Analysis, Elsevier 2021, vol.53, pp. 132-150.

[34] G Falsone, R Laudani, "Probability transformation method for the evaluation of derivative, integral and Fourier transform of some stochastic processes", Journal of Engineering Mathematics, Springer 2021., vol.131., no.12, pp.77-92.

[35] A. Humeau-Heurtier, "Texture Feature Extraction Methods: A Survey," in IEEE Access, 2019, vol. 7, pp. 8975-9000.

[36] W. Liu and W. Chen, "Recent Advancements in Empirical Wavelet Transform and Its Applications," in IEEE Access, vol. 7, pp. 103770-103780.

[37] Z. Hua, Z. Zheng, E. Pahon, M. -C. Péra and F. Gao, "Lifespan Prediction for Proton Exchange Membrane Fuel Cells Based on Wavelet Transform and Echo State Network," in IEEE Transactions on Transportation Electrification 2022, vol. 8, no. 1, pp. 420-431

[38] Z Hajiabotorabi, A Kazemi, FF Samavati, "Improving DWT-RNN model via B-spline wavelet multiresolution to forecast a high-frequency time series", Expert Systems with Applications, Elsevier 2023, vol.138, Art.No.112842.

[39] Diker, Z. Cömert, E. Avci and S. Velappan, "Intelligent system based on Genetic Algorithm and support vector machine for detection of myocardial infarction from ECG signals," 2018 26th Signal Processing and Communications Applications Conference (SIU), Izmir, Turkey, 2018, pp. 1-4.

[40] COS Sorzano, J Vargas, AP Montano, "A survey of dimensionality reduction techniques", arXiv preprint, 2014. arXiv:1403.2877, 2014 - arxiv.org.

[41] G. T. Reddy et al., "Analysis of Dimensionality Reduction Techniques on Big Data," in IEEE Access, 2020, vol. 8, pp. 54776-54788.

[42] D Yang, Z Dong, LHI Lim, L Liu, "Analyzing big time series data in solar engineering using features and PCA", Solar Energy, Elsevier 2017, vol.153, pp.317-328.

[43] Z. Hu, G. Pan, Y. Wang and Z. Wu, "Sparse Principal Component Analysis via Rotation and Truncation," in IEEE Transactions on Neural Networks and Learning Systems, 2016, vol. 27, no. 4, pp. 875-890.

[44] Winursito, R. Hidayat, A. Bejo and M. N. Y. Utomo, "Feature Data Reduction of MFCC Using PCA and SVD in Speech Recognition System," 2018 International Conference on Smart Computing and Electronic Enterprise (ICSCEE), Shah Alam, Malaysia, 2018, pp. 1-6.

[45] A. -K. Seghouane, N. Shokouhi and I. Koch, "Sparse Principal Component Analysis With Preserved Sparsity Pattern," in IEEE Transactions on Image Processing, 2019, vol. 28, no. 7, pp. 3274-3285.

[46] T. Guo, Z. Xu, X. Yao, H. Chen, K. Aberer and K. Funaya, "Robust Online Time Series Prediction with Recurrent Neural Networks," 2016 IEEE International Conference on Data Science and Advanced Analytics (DSAA), Montreal, QC, Canada, 2016, pp. 816-825

[47] B. Girault, "Stationary graph signals using an isometric graph translation," 2015 23rd European Signal Processing Conference (EUSIPCO), Nice, France, 2015, pp. 1516-1520.

[48] S. Oymak and M. Soltanolkotabi, "Toward Moderate Overparameterization: Global Convergence Guarantees for Training Shallow Neural Networks," in IEEE Journal on Selected Areas in Information Theory, 2020, vol. 1, no. 1, pp. 84-105.

[49] K Chandra, A Xie, J Ragan-Kelley, "Gradient descent: The ultimate optimizer", Proceedings in the 36th Conference on Neural Information Processing Systems (NeurIPS 2022), pp.1-12.

[50] A Cutkosky, H Mehta, "Momentum improves normalized sgd", Proceedings of the 37th International Conference on Machine Learning, PMLR 2020, vol.119., 2260-2268.

[51] Q. Zheng, M. Yang, J. Yang, Q. Zhang and X. Zhang, "Improvement of Generalization Ability of Deep CNN via Implicit Regularization in Two-Stage Training Process," in IEEE Access 2018, vol. 6, pp. 15844-15869.

[52] W. Liu, L. Chen, Y. Chen and W. Zhang, "Accelerating Federated Learning via Momentum Gradient Descent," in IEEE Transactions on Parallel and Distributed Systems 2020, vol. 31, no. 8, pp. 1754-1766

[53] M. Gan, C. L. P. Chen, G. -Y. Chen and L. Chen, "On Some Separated Algorithms for Separable Nonlinear Least Squares Problems," in IEEE Transactions on Cybernetics 2018, vol. 48, no. 10, pp. 2866-2874

[54] T. Vu and R. Raich, "On Asymptotic Linear Convergence of Projected Gradient Descent for Constrained Least Squares," in IEEE Transactions on Signal Processing, 2022, vol. 70, pp. 4061-4076.

[55] J. M. Baden, B. O'Donnell and L. Schmieder, "Multiobjective Sequence Design via Gradient Descent Methods," in IEEE Transactions on Aerospace and Electronic Systems 2018, vol. 54, no. 3, pp. 1237-1252.

[56] W. E. L. Ilboudo, T. Kobayashi and K. Sugimoto, "Robust Stochastic Gradient Descent With Student-t Distribution Based First-Order Momentum," in IEEE Transactions on Neural Networks and Learning Systems 2022, vol. 33, no. 3, pp. 1324-1337.

[57] L Franceschi, M Donini, P Frasconi, "Forward and reverse gradient-based hyperparameter optimization", Proceedings of the 34th International Conference on Machine Learning, PMLR 2017,vol.70, pp.1165-1173.

[58] Y. Wang, J. Liu, J. Mišić, V. B. Mišić, S. Lv and X. Chang, "Assessing Optimizer Impact on DNN Model Sensitivity to Adversarial Examples," in IEEE Access, 2019, vol. 7, pp. 152766-152776.

[59] F. Haddadpour, M. Kamani, M. Mahdavi, V. Cadambe, "Local SGD with periodic averaging: Tighter analysis and adaptive synchronization" Part of Advances in Neural Information Processing Systems 32 (NeurIPS 2019), pp.1-13.

[60] X Li, F Orabona, "A high probability analysis of adaptive sgd with momentum", arXiv preprint arXiv:2007.14294, 2020 - arxiv.org., pp.1-13.

[61] M. Wongkar and A. Angdresey, "Sentiment Analysis Using Naive Bayes Algorithm Of The Data Crawler: Twitter," 2019 Fourth International Conference on Informatics and Computing (ICIC), Semarang, Indonesia, 2019, pp. 1-5.

[62] J Tang, Y Xue, Z Wang, S Hu, T Gong, "Bayesian estimation-based sentiment word embedding model for sentiment analysis", CAAI Transactions on Intelligence Technology, Wiley Online Library, 2022, v.ol.7, no.2., pp. 144-155.