2022 2nd International Conference on Intelligent Technologies (CONIT)
Karnataka, India. June 24-26, 2022

Method and Apparatus for Stock Performance Prediction Using Momentum Strategy along with Social Feedback

Vishu Agarwal
Westborough High School
Westborough, Massachusetts, USA
agarwalvis23@student.westboroughk12.org

Madhusudan L
Principal Scientist
SymphonyAI
Bangalore, Karnataka, India

HarshaVardhan Babu Namburi
Lead Scientist
SymphonyAI
Bangalore, Karnataka, India

2022 International Conference on Intelligent Technologies (CONIT) | 978-1-6654-8407-7/22/$31.00 ©2022 IEEE | DOI: 10.1109/CONIT55038.2022.9848364

***Abstract*—Stock prediction and historical stock data analysis have been of great interest over the decades. The research is wide from classical deterministic algorithms to machine learning models and techniques along with the supply huge amounts of historical data. Volatility and Market Sentiment are key parameters to account for during the construction of any stock prediction model. Commonly used techniques like the n-moving days average is not responsive to swings in the stocks and the information sent and posted online has made a huge effect on investors' opinions on the market, making these the two optimal parameters of prediction. Hence, we present an automatic pipeline that has 2 modules — N-Observation period momentum strategy to identify potential stocks and then a stock holding module that identifies market sentiment using NLP techniques.**

***Index Terms*—stock prediction, NLP, sentiment analysis, momentum strategy**

I. Introduction

The stock market allows the investors to buy/sell the stocks, thereby facing the people with the problem of which set of stocks to invest in thanks to the hidden parameters. The complexity lies in the fact that modeling the firm's forecast for a long period of time includes unknown factors, which are difficult to accurately model even by AI algorithms. It requires astute knowledge on services offered by the organization, market competitiveness, consumer taste and preferences, etc. to include in the model for realistic predictions.

Stock market forecasts and historical analysis have received interest over the decades and the earliest analysis dates back to the 1800s where candle sticks were used to analyse the stocks. Statistical analysis was performed on stocks to identify patterns and then its summary provided humans readable signals and help to intelligently invest in stocks. A wider variety of computational techniques have now been adopted over time, which range from recent deep learning techniques to predict the stock values for the next few seconds or minutes [1] [2] to classical variants of moving average techniques [4]. The common claim is that the averaging techniques dissolves the noise in the stocks, although these techniques are not prompt to more recent trends in the stock prices.

The Efficient Market Hypothesis[EMH] [3] states that all the information available at a given point of time that enters the market induces fluctuations in the stock price, making it impossible to model all the information by technical analysis. This hypothesis signifies that price trends almost follow a random walk, making it impossible for accurate forecasts. Stock price forecasting is a difficult problem to solve due to the lack of indicators that drive the volatility.

We take a more liberal approach to induce strategy to invest in stocks. Our work captures 2 aspects of market parameters - Volatility and Market Sentiment. Although, the mentioned parameters are correlated to each other, we model them in 2 different steps. In the first step, we keep our approach simple by modeling the volatility to identify the right set of stocks to invest in, thus not making any predictions on stock prices, but by using historical data to classify the right set of stocks using S&P indicators. In the second step, we use the market sentiment to act on the selected stock holdings by ML model on the news headlines data. We track market sentiment by a state-of-the-art Transformers model [5] to make intelligent decisions on invested stocks. We hand curated 1713 NEWS headlines to 3 labels (Buy, Sell, Neutral) and we make it publicly available to the research community. Our approach show promising results to identify optimal stocks outperforming baseline.

The paper is organized as follows. Section 2 describes the related work. Section 3.1 explains the N-Observation period stock picker, Section 3.2 illustrates the approach used to identify the market sentiment. Section 4 and 5 describe the datasets used and the training approaches. The experimental results are presented in Section 6.

II. Related Work

The moving average techniques [4], [7], [8], [9] are widely adopted to identify trends in stock prices by using historical data for a period and then checking if the stock is trending bullish/bearish. The signals are later used to work out strategies to invest. [4] states the 5 different variants of moving average techniques and concludes that a simple moving average technique outperforms all the other variants. They state that the averaging technique removes most of the noise rather than sophisticated modeling techniques.

The Average Crossover Technique [9] helps to understand the more recent trend change for a stock. The technique has 2 Moving Average(MA) periods: short term and long term. If the short term moving average crosses above the long term moving average, the market sentiment is inferred to be bullish for the stock. If it is vice-versa, then the market is inferred to be bearish on the stock. The complexity is to identify the right period for short term and long term averages.

In modern day techniques, the usage of stock prediction models using machine learning algorithms has increased. The breadth of models include linear models [11] [13], perceptron models [2], and deep learning models [9], [10], [12]. [11] models the stock prediction problem using linear regression and decision tree regression and the work concludes that linear regression shows more promising results when compared to decision trees. [13] models the stock prediction using SVM and the authors state that since SVMs optimize on generalization error rather than the training error, it provides a better estimates for the forecast. The SVMs are then compared with the multi-layer perceptron model and the former outperforms again. Additionally, with the advent of deep learning, more sophisticated models like BERT GAN are used to model stock trends [13].

The above techniques model the stocks using historical stock prices, but [14] tries to understand the correlation between news sentiment and associated price trends to stock prices. They identify that the correlation is weak between a sector of companies, but shows better sentiment for the individual company W.R.T media. [15] The Fine-Tuned Contextualized-Embedding Recurrent Neural Network (FT-CE-RNN) model uses contextual-based embedding on news articles using BERT to compute predictions on stock prices. With the advent of language modeling using efficient low-dimensional embedding, [16] proved to be very effective in storing information. Embeddings store different variants of contextual information subject to the design of the architecture. The advantage of these architectures is that we can use semi-supervised learning and simple gradient methods to accelerate the learning process. Embeddings capture are successful in the language tasks because they are customized to capture a variety of information subject to the problem being solved, even though contextualized based embedding models generalize well across most of the tasks. This problem is well studied by different variants of CNNs and RNNs.

III. Architecture

We propose the N-Observation period optimizer (NBOS-OPT) to determine the hot stocks at any point of time and then to apply the decision module powered by machine learning to provide useful signals on whether to hold the stock or not based on market sentiment polarity. The pipeline comprises of the following modules.

A. Stock Picker

Moving average techniques are widely studied in the literature as they are able to remove the noise in the stock volatility. However, it takes time to adhere to recent changes. The moving average crossover method tries to fix this problem by using 2 moving averages - one for short term and one for long term, thereby identifying bullish and bearish trends. We present a more generalized version (NOBS-OPT) on moving averages to optimize the stocks in the observation period.

Stock predictions for a distant future is merely impossible since we do not have all the information to create a model that accurately derives the stock trends. Our predictions are created for a mid-range period investment, say around 5-30 days.

The N-Observation period optimizer(NBOS-OPT) monitors the stocks over N number of observation periods and identifies the optimal stocks. The motivation for the proposed model is "if the stock has done well recently, the trend should continue for a short period in the future". We have to encapsulate the stocks recent performance in the algorithm design. Determining the effectiveness of the stock in the recent times and forecast of the same requires efficient and robust indicators. We simplify this approach by breaking the recent stock performance into N-Observation periods. We monitor the stocks in the observation periods and evaluate the performance of the same. We optimize the following gain function to identify the optimal set of stocks.

We determine the optimal number of number of stocks as follows. We evaluate each stock and compute its win ratio by the $\psi_{\mathcal{P},s}^{OPT\_SCR}$ function like

$$\psi_{\mathcal{P},s}^{OPT\_SCR} = \sum_{p_i \in \mathcal{P}} IS\_OPT\_STOCK(s, p_i) \qquad (1)$$

where $\mathcal{P}$ is the set of observation periods of interest and s is the stock for which the win ratio is computed.

$IS\_OPT\_STOCK(s, p)$ determines if the stock s can be declared winner for the given observation period p. We compare the stock s performance with the $s\&p$ performance. If stock s did comparatively better than the $s\&p$ indicator, we declare the stock as winner. The function returns $TRUE$ or $FALSE$, thus $\psi_{\mathcal{P},s}^{OPT\_SCR}$ returns the frequency of wins for stock s for the observation period $\mathcal{P}$ of interest.

We compute the win ratio ψ for all the stocks and select the stocks doing better than the threshold. We found that that threshold of 0.8 to be an optimal value to identify a healthy bucket size and the optimal performance for the forecast period. The hyperparameters for the algorithm are length of observation period O, number of observation periods p, threshold on win_ratio T. We present the sensitivity of the results subject to the hyperparameters.

1) Baseline: We propose a baseline model (OBS-OPT) which use one Observation period of length p * O for fair comparison to NOBS-OPT. We use $s\&p$ indicator to identify the stocks. For the status observation period of interest, we identify the stocks performing better than $s\&p$. It is observed that the baseline model shortlists a huge stock bucket and this is expected.

We present the baseline and NOBS-OPT algorithms below. The OBS-OPT algorithm is a simple version identifying the stock performing better than the $s\&p$ indicator during the observation period. The proposed algorithm, NOBS-OPT generalizes OBS-OPT to N observation periods and computes win-ratio for the identified stocks. The configurable parameters for NOBS-OPT is the observation period $\mathcal{O}$, the number of observation periods p, and the threshold score $\mathcal{T}$ that filters the stocks using the computed win_ratio. Configuring win_ratio to a value close to 1 insists that the stock should have a consistence performance in the recent period. We observed that configuring win_ratio between (0.75-9) gave consistent stock buckets and high performance in forecasts.

Algorithm 1 OBS-OPT

Input : Observation Period $\mathcal{O}$, stocks s_i, $i, \dots, n$
Output : s_j $j, \dots, k$ where k is the number of stocks
Set $Result$ = []
for $i = 1, \dots, n$ **do**
    r = $IS\_OPT\_STOCK(s_i, O)$
    if r **then**
        $Result$.add(s_i)
end for
return $Result$

Algorithm 2 NOBS-OPT

Input : Observation Period $\mathcal{O}$, Number of Periods p, stocks s_i, $i, \dots, n$, threshold score $\mathcal{T}$
Output : s_j $1, \dots, k$ where k is the number of stocks
Set $Result$ = []
for $i = 1, \dots, n$ **do**
    win_ratio = 0
    for $j = 1, \dots, p$ **do**
        r = $IS\_OPT\_STOCK(s_i, O_j)$
        if r **then**
            win_ratio += $1/p$
    end for
    if $win\_ratio > \mathcal{T}$ **then**
        $Result$.add(s_i)
end for
return $Result$

B. AI Model

Sentimental analysis is an area of NLP where we identify the emotional tone of the given text. Sentiment analysis can be applied to varying scopes such as documents, paragraphs, sentences, and even phrases. Sentiment analysis is normally domain dependent and some examples where it can be used are

- Understand the customer behavior from the comments/reviews they write about a specific product.
- Predict the connotation of the given text, whether positive/negative/neutral.

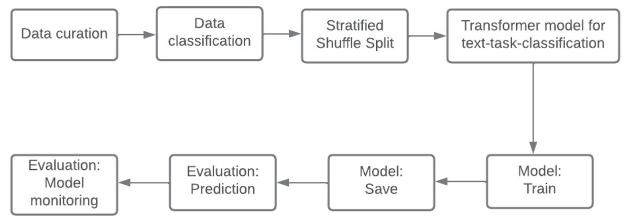

Fig. 1. AI Model Architecture

1) Transformers: Recent advances in neural architectures, such as the Transformer, coupled with the emergence of large-scale pre-trained models such as BERT [5] have revolutionized the field of Natural Language Processing (NLP), pushing the state-of-the-art method for a number of NLP tasks.

2) Transfer Learning: To train the transformers from scratch is a computationally expensive task. It is not recommended to train a transformer from scratch, but the usage of domain-specific data to fine-tune the pre-trained transformer models is considered fine. This is also referred to as transfer learning. Transfer learning reduces the time to train the model on domain specific data and can leverage smaller datasets for training. The drawbacks of transfer learning is that it carries the inherent bias present in the pre-trained models.

We use transformer model to identify the sentiment of the NEWS headline (Buy, Sell, Neutral). The predicted sentiments can be used to make intelligent decisions for the invested stocks. We use pre-trained model and fine tune the same to the curated dataset of 1.7K sample data-points.

IV. Dataset Preparation

A. Stock History

Our optimization module heavily relies on the $s\&p$ indicator to identify the right set of stocks to invest. The optimal stocks identified by the module are also derived from the $s\&p$ indicator. We prepared a dataset comprising the stock values for every day for the time period (2013-2020). All our experiments are conducted on this dataset.

B. News Sentiment

News headlines about different companies are collected from various sources and hand labelled as Buy, Sell, Neutral. We size of the dataset is 1713 comprising (Buy - 1046, Sell - 353, Neutral - 314). The distribution of the train-test split is stated in TABLE I. [1]

V. Training

The Transformers library in Python is used to train a transformer model for predicting the sentiment of the given text. In this context, it utilized the labels of buy/sell/neutral

[1] https://github.com/VishuAgarwal237/StockSentimentDataset

TABLE I
DISTRIBUTION OF THE DATASET FOR DIFFERENT LABELS FOR TRAIN AND TEST SETS

	Buy	Sell	Neutral
Total	1046	353	314
Training	837	282	251
Testing	209	71	63

for the stock. distilbert-base-uncased-finetuned-sst-2-english is used for transfer learning on the hand labelled data. The Transformers library has integrations with MLflow and it is leveraged to track the training experiment runs.

The model is trained for 10 epochs and the graph below shows the loss during training.

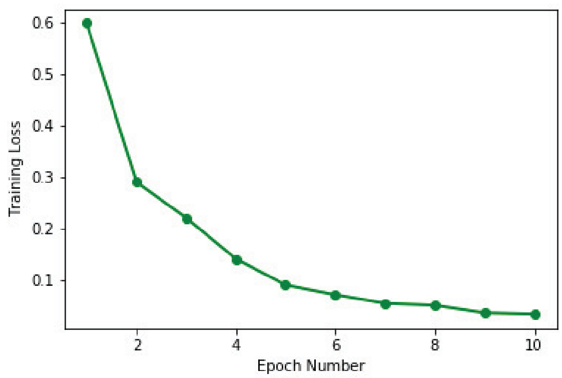

Fig. 2. Training loss

We present the F1-Score during the training period and it is seen that the performance saturates within 10 epochs. The data is imbalanced to Buy/Sell sentiment, For fair comparison on results we use F1 score as metric to represent any findings.

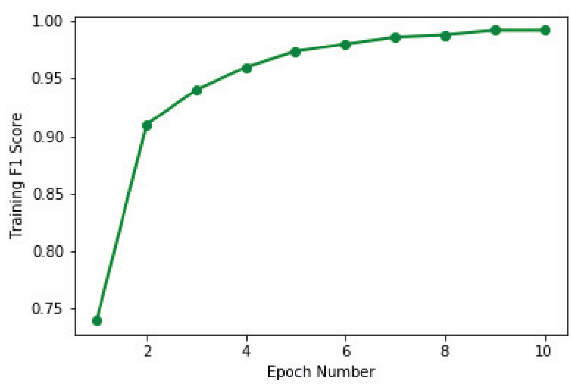

Fig. 3. F1 Score

The Table IV illustrates the performance of the model on the test dataset. The fine-tuned model on train test is evaluated on (Positive - 209, Negative - 71, Neutral - 63) data points.

VI. EXPERIMENTAL RESULTS

A. Stock Picker

The results are presented for the 8 year period between 2003-2020. We compare the results between vanilla $s\&p$ indicator, the baseline model (OBS-OPT) identifying the stocks performing better than the $s\&p$ indicator, and the NOBS-OPT model. We present 2 aspects of results - the bucket size and the performance of the selected stocks in the forecast period.

The bucket size is not specified for the average $s\&p$ since we pick all the valid $s\&p$ stocks for the year of interest. However, we do present the bucket size on the identified stocks for the OBS-OPT and NOBS-OPT models. While the baseline model uses only the $s\&p$ indicator as input, NOBS-OPT is optimized on the observation period length $\mathcal{O}$, number of observation periods p, and the threshold $\mathcal{T}$. We perform grid search on the above parameters and we identified that the total observation period length to be around 100 days and the best results are achieved for observation period length 10.

The interpretation of the numbers is as follows: For a given year y, we split the year into multiple periods W.R.T observation period length $\mathcal{O}$. For any given period in the year $d_1 - d_2$, we identify p observation periods in the past to the date d_1. We compute the optimal stocks using $Algorithm2$ for the defined hyper-parameters. We compute the performance of the identified stocks by $Algorithm2$ for the forecast period $d1 - d2$. We repeat the exercise for l forecast periods in the given year. The mean on the identified stock bucket and the gain is reported in TABLE II.

The gain for the k identified stocks during the forecast period $s_{1,\ldots,k}$ is computed as follows. The length of $d_1 - d_2$ is same as that of the Observation period length O.

$$\mathcal{G}(s, d1, d2) = s^c_{d2} - s^o_{d1} \tag{2}$$

Where s^o_{d1} is the opening price on d1 and s^c_{d2} is the closing price on d2. It is seen that the s^c_{d2} is usually noise since it is conditioned on date d_2 hence we pick the max result out of $d_2 - 2, d_2 - 1, d_2$. This is adopted for all the algorithms presented in TABLE II.

It is seen that when the individual observation period length is large, the model is limited by the number of periods to be evaluated and further away periods do not help identify the optimal stocks. The number of periods experimented is between 1 to 15. $p = 10$ gave the optimal results consistently. However, a very small number of observation periods does not carry lot of information since it would require to set a high threshold $\mathcal{T}$. A large number of observation periods, $p > 15$, is found to be noisy and configuring its threshold is difficult. It can be seen from Fig. 2 that the optimal result is achieved in between 9 to 12 observation periods.

It is seen that increasing the threshold increased the performance as well. The optimal threshold $\mathcal{T}$ is found to be 0.8 and increasing the threshold further did take a hit on the number

TABLE II
8 YEARS PERIOD AVERAGE RESULTS COMPARED WITH BASELINE MODEL.

Year	Baseline 1 Bucket Size	NOBG-OPT Bucket Size	Avg. S&P	Avg. Baseline 1	Avg. NOBG-OPT
2013	234.18	19.86	0.33	0.32	0.44
2014	233.40	19.62	0.54	0.57	0.87
2015	227.32	17.72	0.18	0.20	0.13
2016	250.10	22.86	0.20	0.17	0.23
2017	239.45	19.72	0.29	0.30	0.48
2018	242.54	22.24	0.57	0.56	0.66
2019	239.51	18.81	0.48	0.47	0.53
2020	232.97	19.27	0.77	0.81	0.91

TABLE III
RESULTS FOR 8 YEAR PERIOD, IN COMPARISON TO DIFFERENT THRESHOLD VALUES.

Year	Threshold = 0.7		Threshold = 0.8		Threshold = 0.9	
	Bucket Size	Return	Bucket Size	Return	Bucket Size	Return
2013	53.86	0.32	19.86	0.44	4.08	0.30
2014	56.59	0.73	19.62	0.87	3.86	0.40
2015	53.10	0.18	17.72	0.13	3.75	0.53
2016	65.08	0.25	22.86	0.23	4.89	0.44
2017	56.86	0.29	19.72	0.48	3.94	0.37
2018	60.59	0.51	22.24	0.66	4.48	0.69
2019	57.45	0.44	18.81	0.53	3.40	0.41
2020	55.21	0.86	19.27	0.91	4.56	1.14

TABLE IV
TRANSFORMER MODEL RESULTS TO PREDICT THE SENTIMENT LABELS ON NEWS HEADLINES.

	Precision	Recall	F1 Score	Support
Buy	0.91	0.98	0.94	209
Sell	0.92	0.79	0.85	71
Neutral	0.95	0.86	0.90	63
Accuracy			0.92	343
Macro Average	0.92	0.87	0.90	343
Weighted Average	0.92	0.92	0.91	343

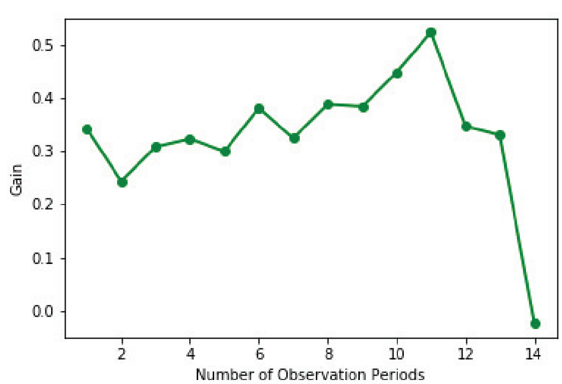

Fig. 4. Observation Periods Comparison

of stocks that can be identified by the algorithm. TABLE II presents the results for different threshold values (0.7, 0.8, 0.9). It can also be seen that if $\mathcal{T}$ = 0.7, the bucket size is huge, therefore, any gain is lost on averaging out the stocks. $\mathcal{T}$ = 0.9 does not generalize well since the number of support stocks are very small in number. The algorithm predicted promising results when the bucket size identified is between 17-22 stocks.

B. News Sentiment

The sentiment detection of the news headlines is modeled as a sentiment classification task. We solve a 3 class problem - Buy, Sell, Neutral. Buy is aligned with a Positive label and Sell is aligned with a Negative label. Since our curated dataset is very small, we use a pre-trained model using transformers. The transfer learning from the pre-trained distilbert-base-uncased-finetuned-sst-2-english model fine-tunes the same to our dataset, showing promising results. It can be seen that the model adopts well to the market sentiment domain with a small set of samples during training.

The results for the classification problem is presented in TABLE IV with respect to support for each label type. The model generalizes well on label imbalance and scores high precision and recall for different classes. The F1-scores are reported for each class. We also present the score and label generated by the fine-tuned model for few sample sentences in the below TABLE V. It is easy to see that the model distinguishes the context for the word Apple between different scenarios. The robust predictions is used for strategic investments on the stocks bucketed from the NOBS-OPT.

TABLE V
SAMPLE SCORES GENERATED BY OUR MODEL, IT CAN BE SEEN THAT MODEL CAPTURES WORDS BY CONTEXT

Sentence	Label	Score
Apple incurred loses	Sell	0.96
Apple had huge profit	Buy	0.99
This is a random text	Neutral	0.99
Apple is a fruit	Neutral	0.85

The frequency distribution of the words from different labels is represented as the following word clouds below.

Fig. 5. Word Cloud - Buy

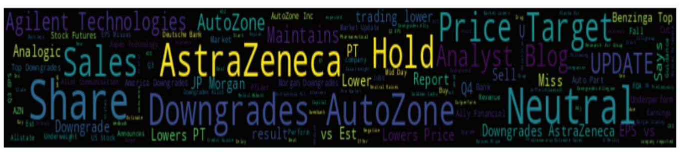

Fig. 6. Word Cloud - Sell

From the above word clouds, we can see that the specific key words help us in classifying the text into sell/buy/neutral classes.

VII. CONCLUSION

Our approach to intelligent investment strategy, using 2 steps to identify the optimal stocks by NOBS-OPT and the encapsulation of market sentiment via AI methodology, shows promising results.

VIII. FUTURE WORK

NOBS-OPT use the observation periods as a key to identify the right candidate stocks. We like to modify NOBS-OPT to a more generalized version to infuse the time factor into the observation periods. This will induce one more hyper-parameter which can be approximated and optimized empirically.

REFERENCES

[1] S. Selvin, R. Vinayakumar, E. A. Gopalakrishnan, V. K. Menon and K. P. Soman. (2017) "Stock price prediction using LSTM, RNN and CNN-sliding window model." International Conference on Advances in Computing, Communications and Informatics: 1643-1647.

[2] A. V. Devadoss and T. A. A. Ligori, "Forecasting of stock prices using multi layer perceptron," Int J Comput Algorithm, vol. 2, pp. 440–449,2013.

[3] Choudhry, Rohit Garg, Kumkum. (2008). A Hybrid Machine Learning System for Stock Market Forecasting. World Academy of Science, Engineering and Technology. 39

[4] Naved, Mohd Srivastava, Prabhat. (2015). The Profitability of Five Popular Variations of Moving Averages on Indian Market Index S P CNX Nifty 50 During January 2004–December 2014. 2.

[5] BERT: Jacob Devlin, Ming-Wei Chang, Kenton Lee, Kristina Toutanova Pre-training of Deep Bidirectional Transformers for Language Understanding. In: Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1(Long and Short Papers),

[6] Gurrib, Ikhlaas. (2014). The Moving Average Crossover Strategy: Does it Work for the SP500 Market Index?. SSRN Electronic Journal. 10.2139/ssrn.2578302

[7] Pasternak, M., July 2003. Inside the Black Box: Technical analysis made clear and simple-Using multiple moving averages. StreetAuthority Swing Trader Newsletter (www.streetauthority.com), 1(19)

[8] Ihab A. El-Khodary (2009). A Decision Support System for Technical Analysis of FinancialMarkets Based on the Moving Average Crossover. World Applied Sciences Journal 6 (11).

[9] G.Zhang, B.E. Patuwo, M.Y.Hu, Forecasting with artificial neural networks: the state of the art, Int.J.Forecasting14 (1998) 35-62.

[10] R. Verma, P. Choure and U. Singh, "Neural Networks through Stock Market Data Prediction," in International Conference on Electronics, Communication and Aerospace Technology, 2017

[11] Rezaul Karim, Md Khorshed Alam. Stock Market Analysis Using Linear Regression and Decision Tree Regression

[12] Priyank Sonkiya, Vikas Bajpai, Anukriti Bansal. Stock price prediction using BERT and GAN

[13] K. Kim, Financial time series forecasting using support vector machines. Neurocomputing, 55, pp. 307–319, 2003.

[14] Wan, X.; Yang, J.; Marinov, S.; Calliess, J.-P.; Zohren, S.; Dong, X. Sentiment correlation in financial news networks and associated market movements. Sci. Rep. 2021, 11, 3062. [CrossRef] [PubMed]

[15] QINKAI CHEN. Stock Movement Prediction with Financial News using Contextualized Embedding from BERT

[16] Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013a. Efficient Estimation of Word Representations in Vector Space. Workshop at International Conference on Learning Representations (ICLR).