iJRASET
International Journal For Research in Applied Science and Engineering Technology



INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 10 **Issue:** VII **Month of publication:** July 2022

DOI: https://doi.org/10.22214/ijraset.2022.45961

www.ijraset.com

Call: 08813907089 | E-mail ID: ijraset@gmail.com

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 10 Issue VII July 2022- Available at www.ijraset.com

Improving Stock Trend Prediction using LSTM Neural Network Trained on a Complex Trading Strategy

S. Z. Mahfooz[1], Iftikhar Ali[2], Muhammad N. Khan[3]
[1]*Department of Computer Science & Engineering,* [2]*Department of Mathematics,* [3]*Department of Chemical Engineering, University of Hafr Al Batin, Kingdom of Saudi Arabia.*

Abstract: ***Technical analysis in stock trading addresses the crucial matter of making optimal trading decisions promptly. Predicting directional movement in the target market using technical indicators is quite common. Besides its many other applications, machine learning helps to solve the algorithmic trading problem of determining optimal trading positions, and some types of deep neural networks have been proven as up-and-coming methods for forecasting the returns of the stock market. The current work presents the idea of training a neural network on a new trading strategy, named, Unified Trading Strategy (UTS) that integrates technical indicators from three well-known categories referred to as leading, lagging, and volatility. The trained network serves as an excellent alternative to the classical technical analysis model by simplifying the process of finding potential events of effective trade with better performance and reusability.***
Keywords: Automated stock trading; Trend prediction; Stock market; Unified trading strategy; LSTM neural network

I. INTRODUCTION

Historical data in financial trading contains latent temporal features that may preserve economic patterns and market trends. Traditional methods of technical analysis utilize charting and mathematical tools to detect latently profitable patterns from historical data [1]. The financial technical indicators use specific mathematical and statistical methods to reflect the more profound connotation of the available stock data in forecasting future trends [2]. There exist many automated trading systems that help to monitor and manage trades. Real-time trading systems [3], simulators [4], and recommendation systems [5] help investors to reduce the risk and maximize profit in stock trading. Algorithm-based technical indicators use calculations of past price data to trace all oversold or overbought conditions of markets. Short-term and long-term traders have several options for analyzing securities to decide their entry and exit points. Some traders may prefer to formulate custom strategies and indicators for better trading ideas. It helps them work with a greater degree of flexibility and increases the chances of successful trading. A suitable process also helps to fine-tune entry and exit levels by overcoming the lagging that results from the usage of past price data.

Many machine learning models capture nonlinear characters in data without prior knowledge [6]. Since the 1990s, automated trading systems based on machine learning and technical indicators have been employed to predict stocks, foreign exchange, and commodity futures markets [7–10]. Various neural networks have been applied for forecasting the stock market returns. Multi-Layered Feed Forward Network with backpropagation algorithm [11], Adaptive Neuro-Fuzzy Inference System [12], Radial Basis Function Network [13], and Long Short Term Memory (LSTM) [14] have outperformed other machine learning techniques such as SVM [15] and Random Forest [16]. LSTM and basic stock market data, including OHLC (Open, High, Low, and Close) prices used for market prediction, have shown improvement over the traditional methods with few exceptions [17,18]. LSTM have been applied extensively for investigating stock markets [19], econometrics [20], and financial market predictions [21].

LSTM model being well-suited to time series data has been used to predict stock future price values after observing historic prices [22-23]. LSTM model with its feature of sequence prediction can be added with additional methods to predict stock market trends for automated trading [24-25]. We developed a method that uses LSTM to solve a classification problem whether a set of price values will form an uptrend or a downtrend. The model is trained to learn the patterns of up/down trends and it generates buy/sell signal as its output.

For training the LSTM model over different patterns of market trends, our trading strategy uses multiple technical indicators to validate existing trends in stock data. The trained LSTM model finally mimic the behavior of these technical indicators and become able to automatically generate buy/sell signals.

©IJRASET: All Rights are Reserved

In this research work, a new trading strategy called Unified Trading Strategy (UTS) has been developed that utilizes the performance and efficiency of well-known technical indicators. UTS validated trend data is used to train an LSTM model to automate the detection of patterns that take over the trends. Our results show that the developed trading strategy and the trained neural network model are finally able to generate buy/sell signals for automated trading that outperforms the manual stock trading. We try to evaluate how a LSTM based recognizer behaves compared to the traditional algorithm-based system.

The rest of this paper is arranged as follows: The second part describes the methodology of the proposed model, and the third part is composed of experimental work and results. Finally, the fourth part provides the conclusion and future directions.

II. METHODOLOGY

Technical indicators are used for chart analysis to detect patterns in price charts. They also offer price analysis to predict future price action. Depending on the functionality, a technical indicator may be lagging, leading, or volatile. Moving Average (MA) and Moving Average Convergence Divergence (MACD) are lagging indicators, also known as trend indicators. Leading indicators like RSI (Relative Strength Index) try to predict future price trends by calculating the price rate changes. They are also referred to as momentum-based indicators. Average Directional Movement Index (ADX) and Average True Range (ATR) are examples of volatility-based indicators [26] that measure the volatility level of the price. Many financial prediction models use these naïve trading signals for feature engineering. Our strategy applies one or more indicators from three different categories of technical indicators. It follows an algorithm to search and select common trend signals. A total of five technical indicators, including Average Directional Index (ADX), Exponential Moving Averages (EMA), Moving Average Convergence Divergence (MACD), Triple Exponential Average (TRIX), and Relative Strength Index (RSI), are considered by UTS. A short description of these indicators is given below.

1) *Exponential Moving Average (EMA:* is widely used in trading practices. It smooths the effects of price changes by giving more weight to the latest data. Our model uses a smoothing factor N=9 to calculate EMA as given below:

$$EMA = Value_{Today}\frac{2}{N+1} + EMA_{yesterday}\left(1 - \frac{2}{N+1}\right) \quad (1)$$

2) *Moving Average Convergence-Divergence (MACD):* is calculated by subtracting the 26-day EMA from the 12-day EMA. A signal line is then calculated by taking a 9-day EMA of MACD. The MACD line crossing above the signal line is a buy signal, and if it is crossing below the signal line, it indicates a sell signal. MACD is a trend-following indicator that reveals changes in momentum, duration, and direction of a stock trend.

$$MACD = 12\,period\ EMA - 26\,period\ EMA \quad (2)$$

3) *The Relative Strength Index (RSI):* is a momentum indicator. The RSI value is measured in a range from 0 to 100. A "high" value above 70 indicates overbought underlying security. A "low" worth less than 30 means that it is oversold. The RSI at each time step is calculated as:

$$RSI = 100 - \frac{100}{1+\frac{AG}{AL}} \quad (3)$$

Where,

AG and AL are simple average-gains and average-losses over 9-day period.

4) *Average Directional Index (ADX):* works together with a Minus Directional Indicator (-DI) and a Plus Directional Indicator (+DI). The directional movement indicators are used to detect stock trends, typically used to determine entry and exit points. The ADX is derived from the smoothed averages between +DI and –DI. It measures the strength of the trend over time. Using all three indicators together, we can determine both the direction and strength of the movement. The value of ADX ranges between 0 and 100. A reading below 20 indicate trend weakness, and all readings above 40 are considered vital.

5) *Triple Exponential Average (TRIX):* is a momentum oscillator representing the percent rate of change of a triple exponentially smoothed moving average. Given below are the steps involved in the calculations of the nine days TRIX.

$$EMA_1 = 9 - period\ EMA\ of\ close\ price \quad (4)$$

$$EMA_2 = 9 - period\ EMA\ of\ EMA_1 \quad (5)$$

$$EMA_3 = 9 - period\ EMA\ of\ EMA_2 \quad (6)$$

$$TRIX = \frac{(EMA_{3,today} - EMA_{3,yesterday})}{EMA_{3,yesterday}} \quad (7)$$

When the stock market index increases its value compared to its previous performance, it is referred to as an uptrend, and a relatively decreasing market index exhibits a downtrend movement [27]. A buy signal is expected when security is believed to be overbought, i.e., trading above its fair or intrinsic value. On the other hand, a sell signal results from an oversold market that has fallen sharply. A trading strategy is a set of predetermined rules that decides how to approach stock trading to achieve profitable returns. UTS is a trend-following strategy that works on a breakout model, buying strength, and selling weakness. The following points help to understand the methodology of UTS, which is also illustrated in Fig. 1.

Technical Indicators (TI's) are applied on close prices of available historical data. Each of these TI's try to determine and mark a trend.

a) A trend period may be determined by some TI's early, and others may observe it late or may not observe it at all. The algorithm scans adjacent days around a trend period to search and count several TI's, marking this period as a potential trend.

b) To validate an uptrend or a downtrend, we set the minimum number of TI's to confirm this trend.

c) The average of days marked by different TI's is selected as the start day of the trend.

d) Sideways are ignored. Only uptrends and downtrends are marked alternatively.

The significant and consistent price change serves as a good feature for the trend period. This change is calculated by taking the difference of prices of consecutive days. A positive change in prices represents an increase (buy-signals), and a negative change indicates a decrease in prices (sell-signals). This increase and decrease serve as class labels for the regression method determining the directional movement. We use these features to train our LSTM model.

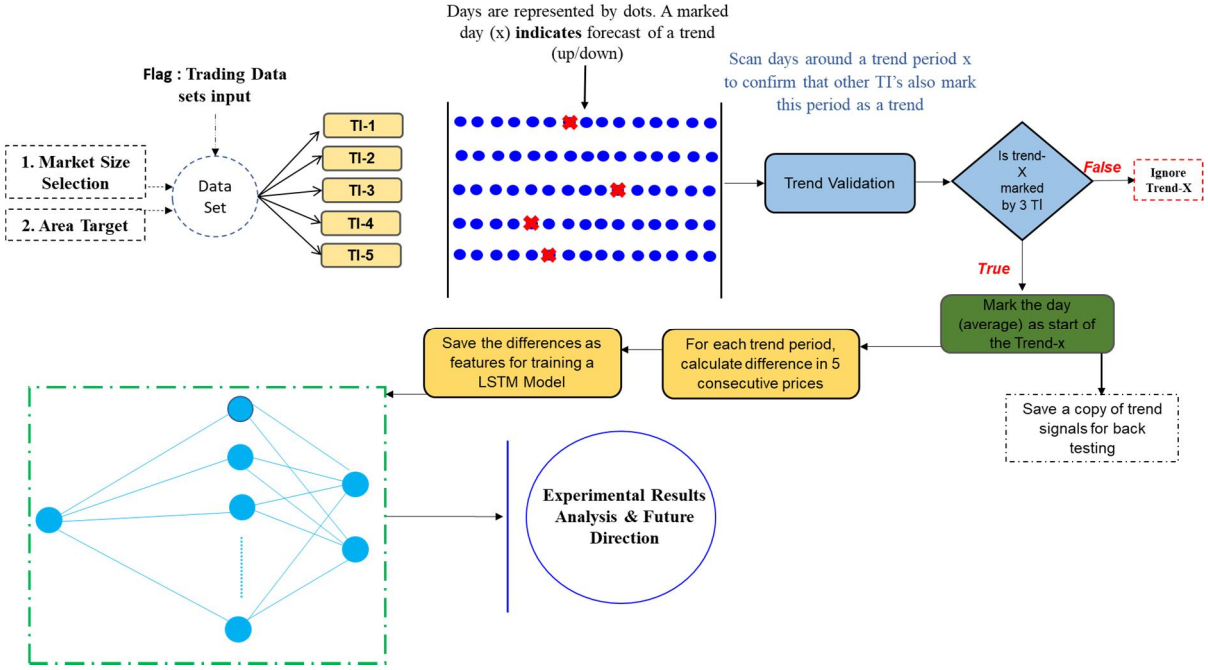


Fig 1. Methodology of Unified Trading Strategy (UTS)

The LSTM model as a machine learning-based model can estimate the predictive power of traditional indicators in different market situations. Also, compared to the other models, the feature detection is excellent by using LSTM models. They are Recurrent Neural Networks (RNNs) designed to tackle the problem of vanishing or blowing up gradients. It is carried out by introducing long short-term memory in RNN, where only necessary information is allowed in the evaluations. Equations 8 to 14 illustrate the structure of LSTM and its architecture is given in Fig. 2.

$$i = \sigma(x_t U^i + s_{t-1} W^i) \quad (8)$$

$$f = \sigma(x_t U^f + s_{t-1} W^f) \quad (9)$$

$$o = \sigma(x_t U^o + s_{t-1} W^o) \quad (10)$$

$$g = tanh(x_t U^g + s_{t-1} W^g) \quad (11)$$

$$c_t = c_{t-1} * f + g * i \quad (12)$$

$$s_t = tanh(c_t) * o \quad (13)$$

$$y = softmax(Vs_t) \quad (14)$$

Where,

$\boldsymbol{i}$: input gate, how much of the new information will be let through the memory cell
$\boldsymbol{f}$: forget gate, responsible for information should be thrown away from memory cell.
$\boldsymbol{o}$: output gate, how much of the information will be passed to expose to the next time step.
$\boldsymbol{g}$: self-recurrent which is equal to standard RNN
c_t : internal memory of the memory cell
s_t : hidden state
y: final output
Moreover, x_t is input at time t and U, V, W are weight matrices.

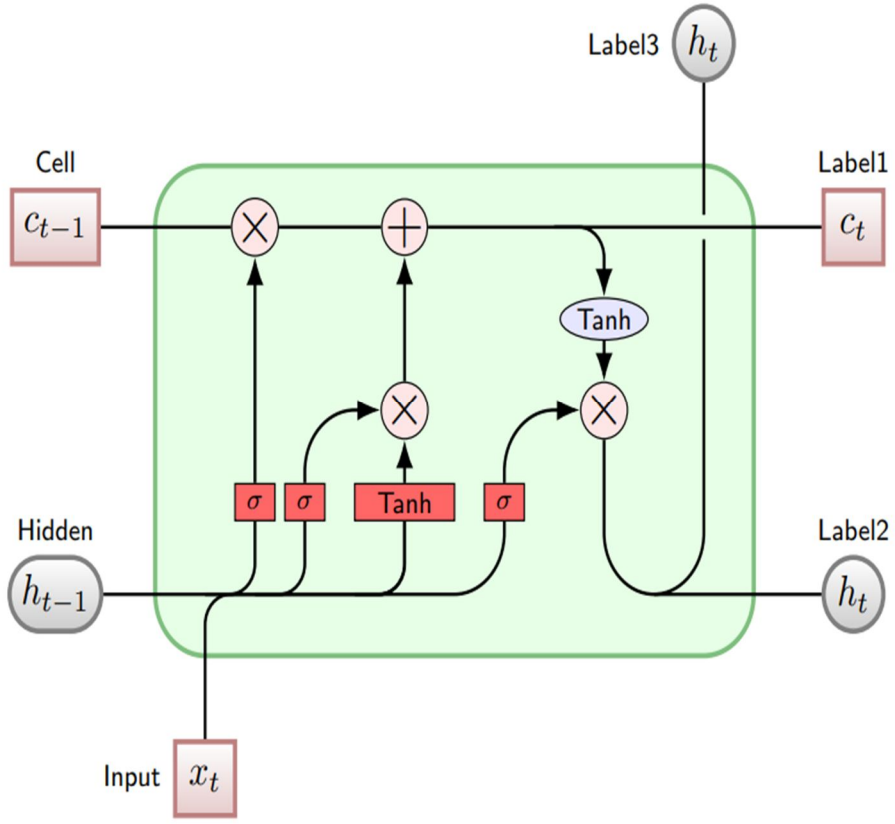


Fig 1. Architecture of Long Short-term Memory (LSTM)

III. EXPERIMENT AND RESULTS

The input features for training our DNN model are generated from the dataset for 120 different US stocks collected from Yahoo Finance [28]. UTS marks the days for up and downtrends, and the change in price for the marked days is calculated. The output is of binary type as for uptrends and downtrends, the difference in prices is positive or negative, respectively. The objective of adapting changes instead of values is to offer a fair comparison across different stocks and periods. It confirms that our model learns a generalized pattern for predicting any stock using historical data, which is the essence of technical analysis.

The proposed model is implemented in Python by using the deep learning library Keras [29]. Model's hyper-parameters candidate and optimal sets are shown in Table I. Different candidate sets were tested including number of hidden units, parameter initializer, activation type, dropout rate, and optimization type. The example of the optimal hyper-parameter values is shown in the last column.

Table I: LSTM model hyper-parameters candidate and optimal sets

Hyper-parameter name	Hyper-parameter values	Example of optimal hyper-parameter values
# of hidden units	{16, 32, 64, 128}	32
initializer	{normal, he_normal, glorot_normal}	normal
Type of activation	{tanh, relu,sigmoid}	tanh
Dropout rate	{0.0, 0.2, 0.3, 0.4}	0.0
Type of optimization	{SGD, Adagrad, Adam}	SGD

Total 4,656 features extracted from trend periods (predicted by UTS) are fed sequentially to the model consisting of an LSTM layer connected with an output layer using 32 connections. LSTM layer is optimized and it finally used tanh as its activation function. As our features are labelled by 1 for uptrend signals and 0 for downtrend signals, we used sigmoid function (Equation. 15) as an activation function at the output layer, and the loss function is set to binary cross-entropy. This function receives the output from the previous layer and generates a probability value between [0,1].

$$S(t) = \frac{1}{1+e^{-t}} \quad (15)$$

A fraction of all total signals that are not a part of the training set is used to test the model. Table II is a binary confusion matrix that is used as evaluation metrics for measuring the performance of the LSTM model. True positive (TP) in a confusion matrix indicates cases wherein both actual label and predicted label are correctly positive (i.e., buy signal). False positive (FP) indicates cases wherein the actual label is negative (i.e., sell signal) but the model prediction is incorrectly positive. False negative (FN) indicates the cases when the actual label is positive but the predicted label is incorrectly negative (i.e., sell signal). Finally, true negative (TN) indicates cases wherein both actual label and predicted label are correctly negative.

Table II: Confusion matrix for the LSTM model evaluation

	Predictive positive	Predictive negative
Actual positive	1093 True Positive	7 False Negative
Actual negative	11 False Positive	1089 True Negative

The values of Precision (Equation. 16) and Recall (Equation. 17) can express the performance of a binary classifier model or more accurately, we can use f1-score (Equation. 18) that represent the harmonic mean of precision and recall. For above confusion matrix, the value of f1-score is 0.99 (1 is the best).

$$Precision = \frac{TP}{TP+FN} \quad (16)$$

$$Recall = \frac{TP}{TP+FP} \quad (17)$$

$$f1\text{-}Score = \frac{2 \times Precision \times Recall}{Precision + Recall} \quad (18)$$

The changes in close prices for the new stock are fed to the trend prediction model that marks the days as potential candidates for buy and sell opportunities. The model generates an output between [0,1] for each set of values. The threshold values are selected to sort out buy/sell signals depending upon the volatility of a stock. Typically, all the values greater than 0.70 represent buy signs, and all values less than 0.30 indicate sell signals. Rest of the values represent a range i.e., one of the three ways a market can be biased.

The performance of a trading system is assessed on existing historical data from a stock market for a specific time. The analysis resulting from a single instrument (stock or others) should not be trusted entirely, as the selection of trading data may be biased to show a trading strategy looks profitable. The choice of multiple stocks having diverse patterns features an ideal assessment. This research forms a test bench (Table III) composed of 8 stocks from different sectors, profiles, and volatilities.

©IJRASET: All Rights are Reserved | SJ Impact Factor 7.538 | ISRA Journal Impact Factor 7.894 |

For the sake of generality, all the algorithm parameters remain unchanged over the entire test bench. Regarding the trading horizon, the ten years (2010 to 2019) data is collected to represent the set of economic phenomena for the selected markets. It less likely contains significant market regime shifts that may harm the stability of a trading strategy.

Table III. Performance assessment testbench

Stock Symbol	Company Name	Sector
SPY	S&P 500 ETF Trust	Trading Index
DIA	Dow Jones Industrial Average ETF Trust	
AAPL	Apple Inc.	Technology
MSFT	Microsoft Corporation	
MCD	McDonald's Corporation	Consumer Cyclical
NKE	NIKE, Inc.	
PG	The Procter & Gamble Company	Consumer Defensive
JNJ	Johnson & Johnson	Healthcare

Some Key Performance Indicators (KPI's) are selected to assess the trading model accurately. Fig. 3. presents a list of these performance indicators employed to quantify the performance of the proposed trading system.

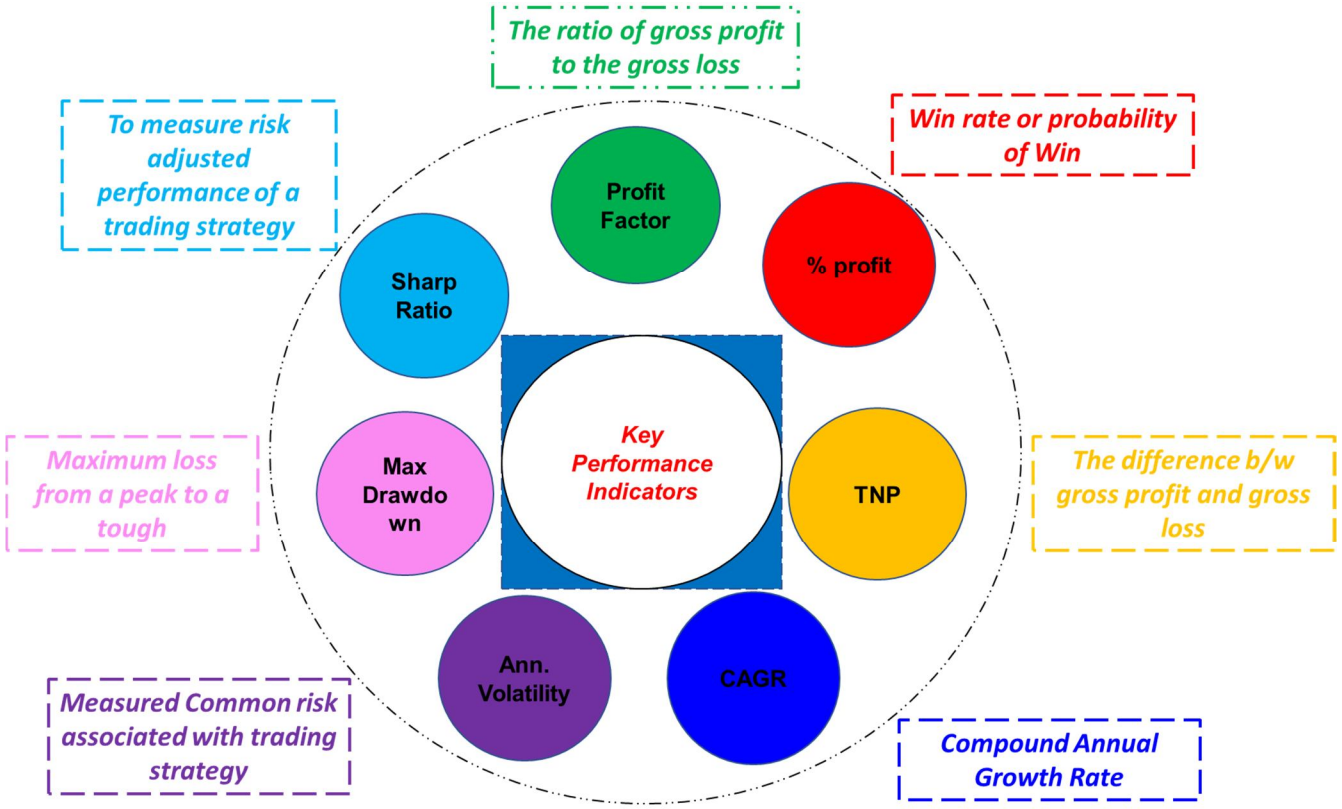


Fig 3. Quantitative performance assessment indicators

These performance indicators provide information about the performance and analysis of the trading system's strengths and weaknesses. We compare the performance of both the unified strategy and the trained model with the Buy and Hold (B& H) strategy.

The calculations of performance indicators in Table IV seem appropriate to analyze the trading policies and the trained model accurately. LSTM based model trained on UTS delivered higher risk-adjusted measurements for all stocks as indicated by the Sharpe ratio [30] and profit factor. Sharpe ratio was introduced by Sharpe in 1996. It is called reward-to-variability ratio that determines the performance of an investment. It is defined as follows:

$$SR = \frac{\mu - R_f}{\sigma}$$

Where μ represents the expected return, σ is the standard deviation of return, and R_f risk-free rate of return. Profit factor is the ratio of total number of winning trades to the total number of losing trades for a specific trading period. It is a simple performance metric to assess the efficiency of a trading algorithm. It is defined as follows:

$$Profit\ factore = \frac{Number\ of\ winning\ trades}{Number\ of\ losing\ trades}$$

Table IV. Performance assessment on different stocks

Stock	Strategy	Sharpe Ratio	Profit Factor
AAPL	B& H	1.00	∞
	UTS	1.37	14.39
	LSTM	1.76	15.51
DIA	B& H	0.93	∞
	UTS	1.25	15.20
	LSTM	1.98	8.33
SPY	B& H	0.93	∞
	UTS	1.25	15.20
	LSTM	1.98	8.33
JNJ	B& H	0.71	∞
	UTS	0.88	3.10
	LSTM	1.53	16.57
MCD	B& H	0.77	∞
	UTS	1.18	9.05
	LSTM	1.56	12.00
NKE	B& H	1.14	∞
	UTS	1.35	8.31
	LSTM	1.45	11.27
MSFT	B& H	0.92	∞
	UTS	1.12	4.32
	LSTM	1.89	7.98
PG	B& H	0.58	∞
	UTS	0.96	5.55
	LSTM	3.30	12.39

The results of percent profit and CAGR (Fig. 4(a-b)) strengthen the observation that the proposed forecast-based strategy and LSTM model consistently deliver significant excess returns over the market index.

The graphs of annual volatility, max drawdown, total net profit in Fig. 4(c) & (e) show that the trained model can self-adapt during both bullish and bearish stock markets. The UTS algorithm can be ranked second as its overall performance is better than the buy-and-hold trading strategy.

©IJRASET: All Rights are Reserved | SJ Impact Factor 7.538 | ISRA Journal Impact Factor 7.894 |

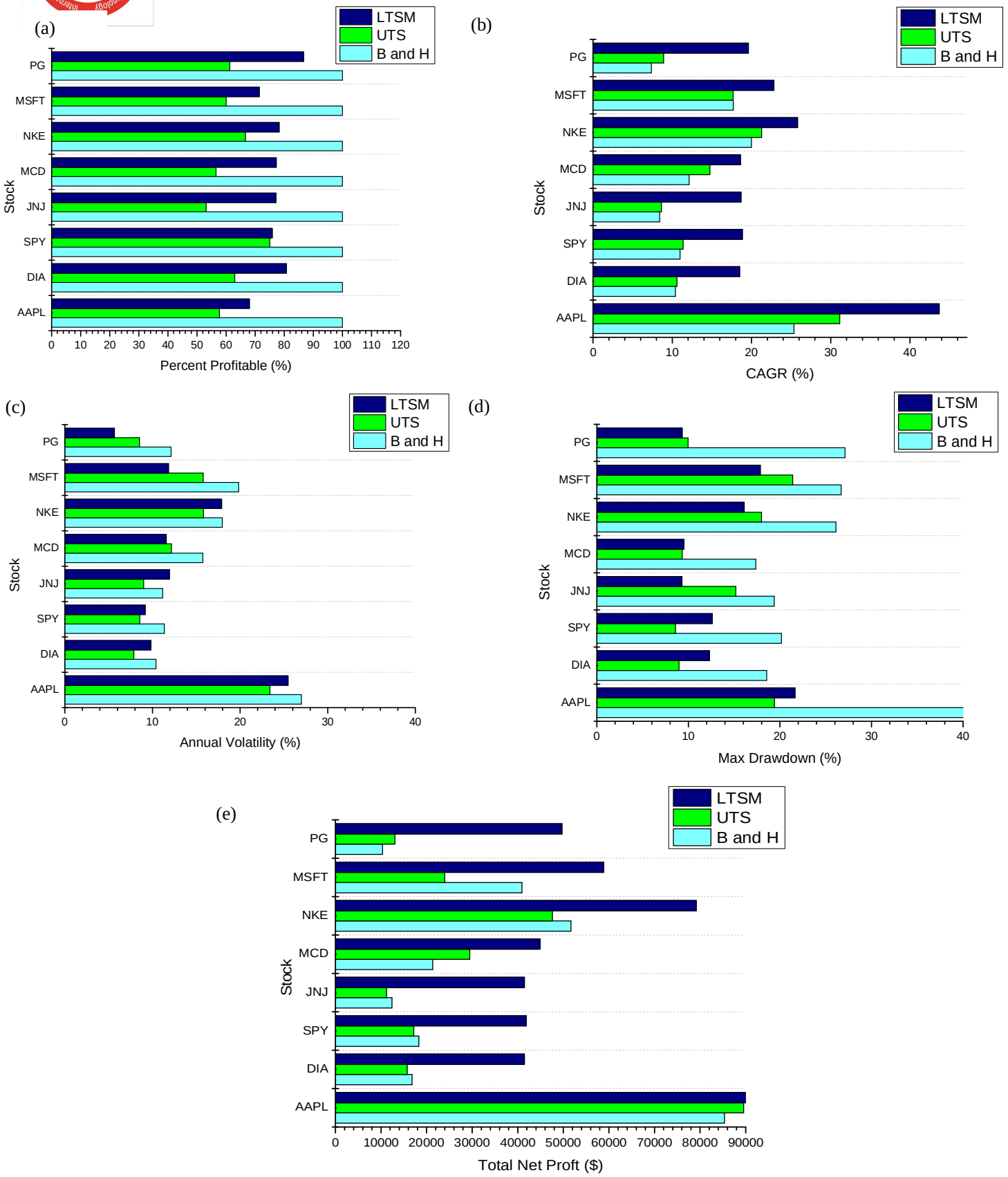


Fig 4. Forecasted performance assessment (a) Percent profitable (b) CAGR (%) (c) Annual volatility(d) Maximum Drawdown (e) Total net Profit

Besides the computation of given performance indicators, Fig 5 & 6 respectively provides extra insights with a graphical representation of stocks' annual returns resulting from each trading strategy. The proposed LSTM trading model trained with UTS leads with all positive returns that is an evidence of a profitable trading system. It attempts to properly handle both bull and bear markets with different levels of volatility.

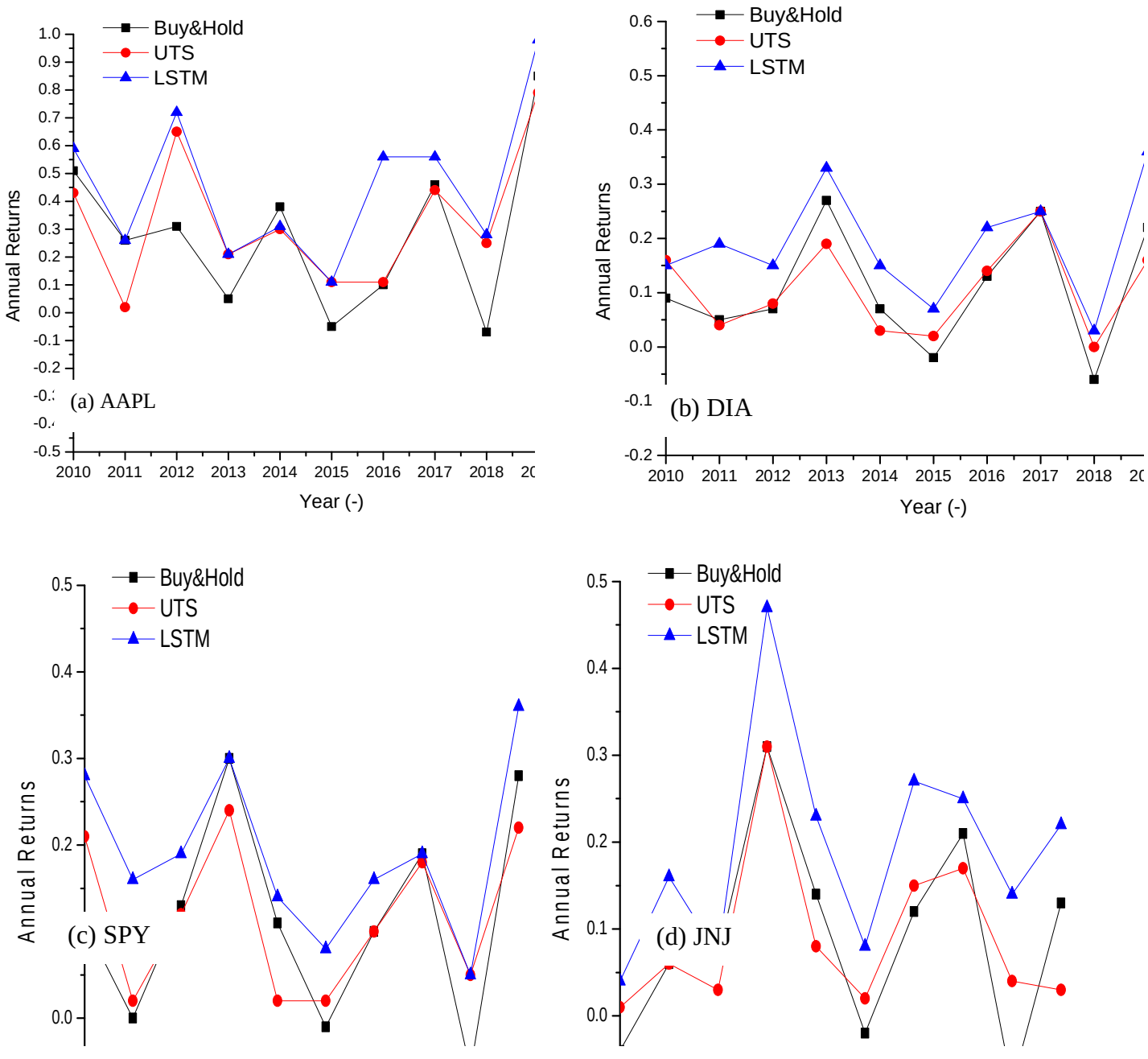


Fig 5. Stocks annual returns for (a) AAPL (b) DIA (c) SPY and (d) JNJ over a span of 2010-2019.

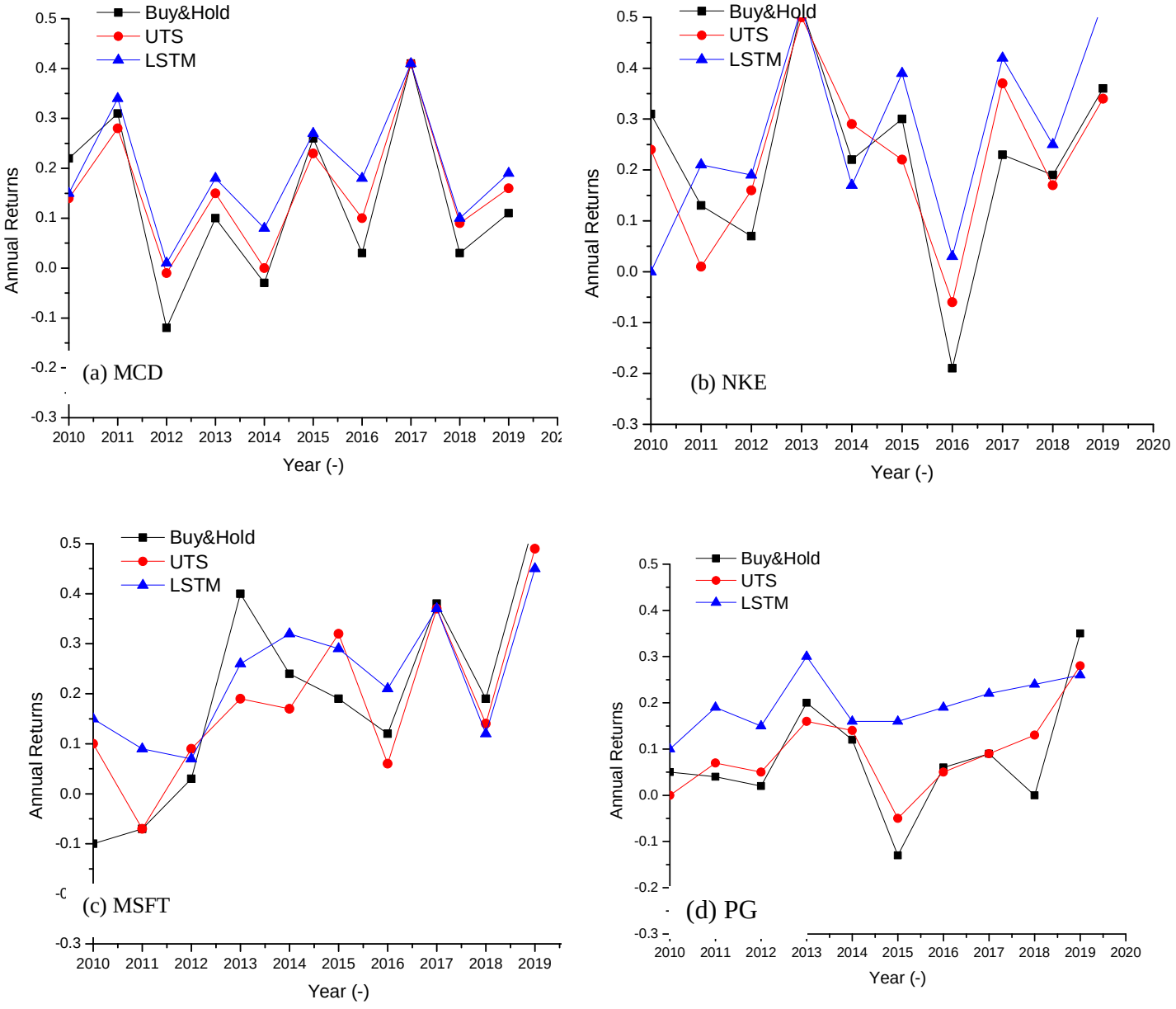


Fig 6. Stocks annual returns for (a) MCD (b) NKE (c) MSFT and (d) PG over a span of 2010-2019.

IV. CONCLUSION & FUTURE DIRECTIONS

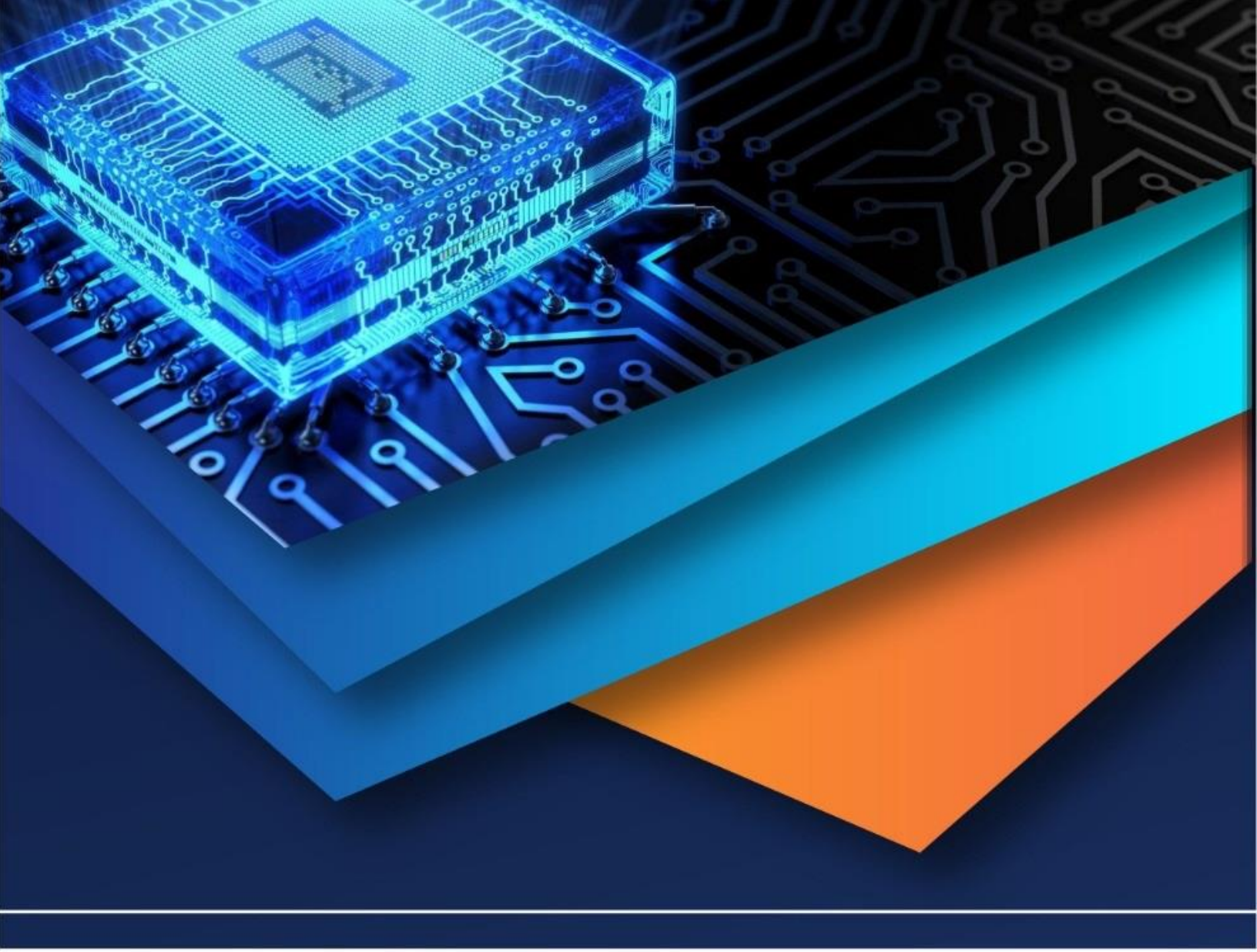
The paper introduced a unified trading strategy that can blend the distinctive features of individual indicators from different categories. The research work also employed an LSTM neural network model trained over the trend signals generated by the designed trading strategy. The model learns the characteristics of the unified trading strategy and becomes able to provide insight into the future movements of any input security prices. The trained model efficiently exploits the correlation between stock patterns and future trends. Our results show that the designed LSTM neural network learns behavior of a complex trading strategy and outperforms the benchmark or traditional trading algorithms in forecasting of the market trends.

Any system that can exploit the potential of diverse methods in technical analysis may serve as a reliable mode for significant profit. A hybrid model for prediction is possible by incorporating the proposed unified trading strategy to the sentiment analysis of news articles, social media, and corporate reports. Furthermore, new research directions may formulate advance metrics to measure the performance and robustness of the prediction models.

REFERENCES

[1] Lo, A.W.; Mamaysky, H.; Wang, J.; Acampora, R.; Allen, F.; Berger, S.; Epstein, M.; Jegadeesh, N.; Kao, E.; Sanzone, D.; et al. NBER WORKING PAPER SERIES Foundations of Technical Analysis: Computational Algorithms, Statistical Interference, and Empricial Implementation. J. Finance 2000, 4, 1706–1715.
[2] Lai, T. L., & Xing, H. (2008). Statistical models and methods for financial markets. New York: Springer.
[3] Chen, Y.; Wang, X. A hybrid stock trading system using genetic network programming and mean conditional value-at-risk. Eur. J. Oper. Res. 2015, 240, 861–871, doi:10.1016/J.EJOR.2014.07.034.
[4] Jacobs, B. I., Levy, K. N., & Markowitz, H. M. (2004). Financial market simulation. The Journal of Portfolio Management, 30(5), 142-152.
[5] Wang, W.; Mishra, K.K. A novel stock trading prediction and recommendation system. Comput. Sci. Multimed. Tools Appl. 2017, 77, 4203–4215, doi:10.1007/S11042-017-4587-Z.
[6] Atsalakis, G.S.; Valavanis, K.P. Surveying stock market forecasting techniques – Part II: Soft computing methods. Expert Syst. Appl. 2009, 36, 5932–5941, doi:10.1016/J.ESWA.2008.07.006.
[7] Huang, B., Huan, Y., Xu, L. D., Zheng, L., & Zou, Z. (2019). Automated trading systems statistical and machine learning methods and hardware implementation: a survey. Enterprise Information Systems, 13(1), 132-144.
[8] Borch, C. (2022). Machine learning, knowledge risk, and principal-agent problems in automated trading. Technology in Society, 101852
[9] Kim, K.J. Financial time series forecasting using support vector machines. Neurocomputing 2003, 55, 307–319, doi:10.1016/S0925-2312(03)00372-2.
[10] Han, H., Teng, J., Xia, J., Wang, Y., Guo, Z., & Li, D. (2021). Predict high-frequency trading marker via manifold learning. Knowledge-Based Systems, 213, 106662.
[11] Ramani, P.; Murarka, P.D. Stock Price Prediction using Multi-Layer Feed Forward Neural Network. Int. J. Information, Commun. Comput. Technol. 2013, 1.
[12] Esfahanipour, A., & Aghamiri, W. (2010). Adapted neuro-fuzzy inference system on indirect approach TSK fuzzy rule base for stock market analysis. Expert Systems with Applications, 37(7), 4742-4748
[13] Guo, Z., Wang, H., Yang, J., & Miller, D. J. (2015). A stock market forecasting model combining two-directional two-dimensional principal component analysis and radial basis function neural network. PloS one, 10(4), e0122385.
[14] Kim, T., & Kim, H. Y. (2019). Forecasting stock prices with a feature fusion LSTM-CNN model using different representations of the same data. PloS one, 14(2), e0212320.
[15] Chen, Y.; Hao, Y. A feature weighted support vector machine and K-nearest neighbor algorithm for stock market indices prediction. Expert Syst. Appl. 2017, 80, 340–355, doi:10.1016/J.ESWA.2017.02.044.
[16] Park, H. J., Kim, Y., & Kim, H. Y. (2022). Stock market forecasting using a multi-task approach integrating long short-term memory and the random forest framework. Applied Soft Computing, 114, 108106.
[17] Jiang, W. (2021). Applications of deep learning in stock market prediction: recent progress. Expert Systems with Applications, 184, 115537
[18] Kumbure, M. M., Lohrmann, C., Luukka, P., & Porras, J. (2022). Machine learning techniques and data for stock market forecasting: a literature review. Expert Systems with Applications, 116659
[19] Baillie, R.T. Long memory processes and fractional integration in econometrics. J. Econom. 1996, 73, 5–59, doi:10.1016/0304-4076(95)01732-1.
[20] Sadique, S.; Silvapulle, P. Long-term memory in stock market returns: international evidence. Int. J. Financ. Econ. 2001, 6, 59–67, doi:10.1002/IJFE.143.
[21] Fischer, T.; Krauss, C. Deep learning with long short-term memory networks for financial market predictions. Eur. J. Oper. Res. 2018, 270, 654–669, doi:10.1016/J.EJOR.2017.11.054.
[22] Yadav, A., Jha, C. K., & Sharan, A. (2020). Optimizing LSTM for time series prediction in Indian stock market. Procedia Computer Science, 167, 2091-2100.
[23] Cao, J., Li, Z., & Li, J. (2019). Financial time series forecasting model based on CEEMDAN and LSTM. Physica A: Statistical Mechanics and its Applications, 519, 127-139.
[24] Zhao, Z., Rao, R., Tu, S., & Shi, J. (2017, November). Time-weighted LSTM model with redefined labeling for stock trend prediction. In 2017 IEEE 29th international conference on tools with artificial intelligence (ICTAI) (p. 1210-1217). IEEE.
[25] Rundo, F. (2019). Deep LSTM with reinforcement learning layer for financial trend prediction in FX high frequency trading systems. Applied Sciences, 9(20), 4460.
[26] Adam, K., Marcet, A., & Nicolini, J. P. (2016). Stock market volatility and learning. The Journal of finance, 71(1), 33-82.
[27] Suresh, A. A Study on Fundamental and Technical Analysis. Int. J. Mark. Financ. Serv. Manag. Res. 2013, 2, 44–59.
[28] YAHOO Yahoo Finance - Stock Market Live, Quotes, Business & Finance News 2022.
[29] KERAS Keras: the Python deep learning API Available online: https://keras.io/ (accessed on Jan 20, 2022).
[30] Sharpe, W. F. (1998). The sharpe ratio. Streetwise–the Best of the Journal of Portfolio Management, 169-185.

©IJRASET: All Rights are Reserved | SJ Impact Factor 7.538 | ISRA Journal Impact Factor 7.894 |



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Call : 08813907089 (24*7 Support on Whatsapp)