

Forecasting the Cboe Vix with a Hybrid LSTM-ARIMA Model and Investor Sentiment Analysis

Yossi Shvimer¹, Victor Murinde² and Avi Herbon^{3*}

Centre for Global Finance, SOAS University of London, United Kingdom

Centre for Global Finance, SOAS University of London, United Kingdom

Department of Management, Bar-Ilan University, Israel

***Corresponding Author:** Avi Herbon, Department of Management, Bar-Ilan University, Israel, Email: avher@bezeqint.net

Received Date: October 03, 2025 **Accepted Date:** October 07, 2025 **Published Date:** October 14, 2025

Citation: Yossi Shvimer, Victor Murinde, Avi Herbon (2025) Forecasting the Cboe Vix with a Hybrid LSTM-ARIMA Model Investor Sentiment Analysis. J Artif Intel Soft Comp Tech 2: 1-18

Abstract

This paper introduces a new next-day forecasting model for the Chicago Board Options Exchange (CBOE) Volatility Index (VIX). The model is a hybrid of the long short-term memory (LSTM) neural network and the autoregressive integrated moving average (ARIMA) model. Investor sentiment scores are incorporated into the hybrid LSTM-ARIMA model and empirically evaluated using natural language processing (NLP) techniques applied to financial news sources. Based on out-of-sample data for 2019–2020, including the COVID-19 crisis period, the hybrid LSTM-ARIMA model with sentiment scores achieves higher forecasting accuracy and demonstrates improved simulated forecasting performance compared to the LSTM-ARIMA model without sentiment. Overall, the findings highlight the value of integrating hybrid soft computing methods with sentiment analysis for forecasting noisy and nonlinear time series, with financial market volatility serving as a case study.

Keywords: Soft computing; Hybrid LSTM-ARIMA model; Sentiment analysis; Time series forecasting; Volatility index (VIX); Financial market forecasting



© [2025] Yossi Shvimer, Victor Murinde, Avi Herbon. This is an open access article published by JsScholar Publishers and distributed under the terms of the Creative Commons Attribution 4.0 International License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited.

Introduction

Arguably, stock market forecasting is an important theoretical and practical problem in economics and finance. Timely prediction of market dynamics is challenging due to volatile stock market characteristics [1]. Stock market volatility acts as a barometer of financial risk or uncertainty surrounding investments in financial assets and, therefore, it is of natural interest to individual investors, mutual fund managers, financial industry regulators as well as policymakers [2]. Many efforts to forecast stock market volatility are recorded in existing literature [3]. Yet, the results of these efforts have been rather diverse [4].

The Chicago Board Options Exchange's (CBOE) implied volatility index (VIX) measures the expected volatility associated with the S&P 500 index returns over the subsequent 30 days, as implied by the prices of the basket of options contracts (on the S&P 500 index) with maturities between 23 and 37 days. Given the nature of the underlying asset (S&P500 index), VIX futures contracts are cash-settled, with final settlement taking place on Wednesday, which is 30 days before the third Friday of the subsequent expiry month. These futures contracts' primary purpose is to enable hedgers and speculators to trade volatility at a low cost (high liquidity) environment. Forecasting VIX level has received less attention than stock indices by existing literature in this regard [5], although the VIX has become the standard benchmark for measuring stock market volatility for S&P500 [6].

1.1 VIX Estimation

Various previous pricing studies about the forecasting VIX index level used the generalized autoregressive conditional heteroskedasticity (GARCH)-type models. GARCH models try to minimize the noise from the time-series data itself. The basic GARCH specification captures time-varying volatility and incorporates known characteristics of real-world return processes, including asymmetric response to up and down shocks as well as jumps that are well suited for VIX estimation. [7] Estimated VIX futures contract prices based on the Heston and Nandi GARCH (HN-GARCH) model. It is shown that the VIX index level and VIX futures prices for a joint estimation can effectively cap-

ture the variations of the market VIX index level and the VIX futures contract prices simultaneously [8]. Studied the out-of-sample VIX futures pricing based on GARCH and Goldstein-Jagannathan-Runkle GARCH (GJR-GARCH) models. It is found that concerning pricing errors of the VIX futures contracts prices and the VIX index's level, the new methods significantly outperform a continuous-time benchmark based on the Heston volatility model [9, 10] found that the IG-GARCH model can reduce the absolute pricing errors in evaluating the VIX index level obtained by the HN-GARCH model by 11–29%.

Financial time series, such as stock prices and VIX index level, are vulnerable to behavioral factors such as risk aversion and exogenous factors such as macroeconomic shocks. Both elements are practically impossible to capture with existing mathematical models and add noise to time series estimations. Linear models, which tend to dominate the financial economics literature, are only partially successful in capturing the relevant underlying trend [5]. These models have low forecasting accuracy and high volatility [11, 12]. The models seem to be of limited assistance to traders in terms of delivering consistently useful forecasts for decision-making.

We draw insights from machine learning literature by using the long-short term methodology (LSTM) from [13] and the auto-regressive integrated moving average (ARIMA) model [14]. Both models are known for processing and forecasting values based on time-series data. ARIMA models assume that the present data have a linear function of past data points and past errors. These errors are white in nature and require that the data be made stationary before fitting it a linear equation.

Because both LSTM and ARIMA models can forecast time-series, several papers empirically compare the prediction accuracy of the models [15]. Empirically compared ARIMA with LSTM prediction models for several stock indices between 1985 and 2018. They found that LSTM outperforms traditional-based algorithms such as the ARIMA model. The root-mean-square error (%) obtained by ARIMA was 55.3 while the corresponding value from LSTM was 7.814, indicating the superiority of LSTM for the given dataset. Hence, researchers have tried to hybridize ARIMA

and LSTM models to improve the forecast accuracy of either of the models used separately. For example, Zhang (2003) proposed a hybrid methodology that combines ARIMA and Artificial Neural Network (ANN) models and showed empirically on real datasets of Forex indicator that the integrated model can improve the forecast accuracy of either of the models used separately.

[16] Used ARIMA models to identify and magnify the existing linear structure in data, and then a multilayer perception to select a model to capture the underlying data generating process and predict the future price, using pre-processed data. It is shown that the [16] model had better performance for one-step-ahead performance than Zhang (2003). Babu and Reddy (2014) used the Hybrid ARIMA-LSTM methodology by filtering the data using the moving average for the trend. Then they estimated the trend with the ARIMA model and the noise from the trend with the LSTM model, separately. Their one-step-ahead forecast obtained higher accuracy than both Zhang (2003) model and the [16] model.

Our investigation extends the above work of Babu and Reddy (2014) and follows the growing literature of using machine learning to forecast the VIX index level. We propose a new forecasting model of the VIX level using the LSTM algorithm and ARIMA model and validate it for real-time data. The new hybrid LSTM-ARIMA model, with deep learning, is used for forecasting VIX; we integrate sentiment extracted from financial news into a hybrid ARIMA-LSTM forecasting framework, providing a soft computing approach that combines statistical, deep learning, and natural language processing components.

1.2 Investor Sentiment Forecasts Future Stock Returns

There has been considerable debate in the recent literature as to whether investor sentiment predicts stock returns. [17] Examine extensive dataset of messages on 91 firms posted on the Yahoo! Finance message board over the period January 2005 to December 2010, and examined whether investor sentiment as expressed in posted messages has predictive power for stock returns, volatility, and trading volume. They found no evidence that investor sentiment forecasts future stock returns either at the aggregate

or at the individual firm level. Furthermore, they didn't found no significant evidence that investor sentiment from Internet postings has predictive power for volatility and trading volume. Despite [17] empirical results, recent studies reported that news articles could improve the accuracy of predicting stock price movements. For example, [18] examined the effectiveness of deep generative approaches for stock movement prediction from social media data using a neural network architecture for this task. They tested their model on a new comprehensive dataset and showed that their model increased price accuracy rather than without using social media [19]. Introduced a knowledge-based method to extract relevant financial news adaptively. They used an output attention mechanism to allocate different weights to different days to stock price movement. Through empirical studies based upon three individual stocks' historical prices, they showed an accuracy of 68%, higher than 58% accuracy obtained by sentence embeddings input and standard neural network prediction model. These studies showed that news articles could improve accuracy in predicting stock price movement. However, few studies have applied sentiment directly to VIX forecasting, with most focusing on the price level of the underlying asset. This highlights a research gap where hybrid soft computing approaches can be applied.

1.3 Contribution

This paper contributes to the artificial intelligence and soft computing literature in three main ways. First, we suggest engaging investors' sentiment using natural language processing (NLP), which relies on machine learning techniques to parsing text sentiment obtained from financial news articles for stock prediction. Investors' sentiment can improve prediction accuracy obtained by models basing their predictions only on historical prices. Second, to better forecast future values, we add to the LSTM model the investors' sentiment scores. The LSTM approach has the advantages of analyzing relationships among time-series data through its memory function. This property can quantify the long-term relationship between sentiment analysis data and VIX values. Third, we adopt the ARIMA model and combine it with the LSTM method to capture the VIX index's trend fluctuation.

The remainder of this article is structured as follows. In Section 2, we explain the data, software, and hardware of our proceeding analysis. The methodology, which consists of three stages, is discussed in Section 3. Section 4 presents the key results of applying the suggested model and some robustness tests. The conclusion is offered in Section 5.

2. Data, Software, and Hardware

2.1. Vix Index Data

We use the S&P 500 VIX index data from CBOE for the empirical study. The entire dataset is from January 1th, 2017, until December 31th 2020, covering 1026 trading days, consisting of the COVID-19 period characterized by a sharp increase in the VIX index level.

2.2. Financial Websites Data

Using Webhose.io platform published from January 1th, 2017, until December 31th 2020, we have collected from 13 major financial websites mentioning any of the words “S&P”, “Volatility”, “Stocks”, “VIX”, “and Futures”. The cause for using these keywords rather than the only “VIX” keyword draws from the assumption under which the investor sentiment of the underlying asset (S&P500) affects the VIX index level. The websites include CNN, Reuters, Bloomberg, etc. The total number of financial news articles in the whole dataset is 79,455.

2.3. Software and Hardware

We used Python 3.7 (Python Software Foundation, 2016), with NumPy (Van Der Walt et al., 2011) and pandas [20] packages for data preparation. We developed the architecture of deep learning LSTM networks with Keras (Chollet, 2015) on top of Google TensorFlow, a powerful library for large-scale machine learning on heteroge-

neous systems [21].

Investors’ sentiment analysis uses NLP methods and algorithms that are either rule-based, hybrid, or rely on machine learning techniques to learn data from datasets. The investors’ sentiment analysis in our study is carried out separately with TextBlob Library [22] as well as with Valence Aware Dictionary for Sentiment Reasoning (“Vader”) that constructs and empirically validates a list of lexical features [23] (Hutto and Gilbert, 2014). Vader is a human-validated sentiment analysis method developed for micro-blogging and social media, requiring no training data. It consists of a list of lexical features and associated sentiment measures. Based on the language’s grammatical and syntactic usage, several rules are formed, which are used to determine the text’s sentiment, whereas each word is assigned to a semantic orientation as a positive or negative value [24].

3. Hybrid Lstm-Arima Model

3.1. The General Framework

Our methodology consists of three steps shown in Figure 1. First, we build the input vector based on historical VIX index levels and investments’ sentiment analysis scores necessary for training and forecasting VIX index level. Second, we provide an in-sample analysis separately for LSTM networks and the ARIMA model based on 75% of the data (“train data”). We present the Hybrid LSTM-ARIMA approach to obtain forecasts, based on the advantages of ARIMA and LSTM models. Using the Hybrid LSTM-ARIMA model, our goal is to estimate the VIX index level in day $t+1$.

The rest of this section details the three stages outlined above and are illustrated in Figure 1. Third, we make out-of-sample data forecasting, based on the remaining 25% of the data (“test data”), separately for the LSTM model and ARIMA model.

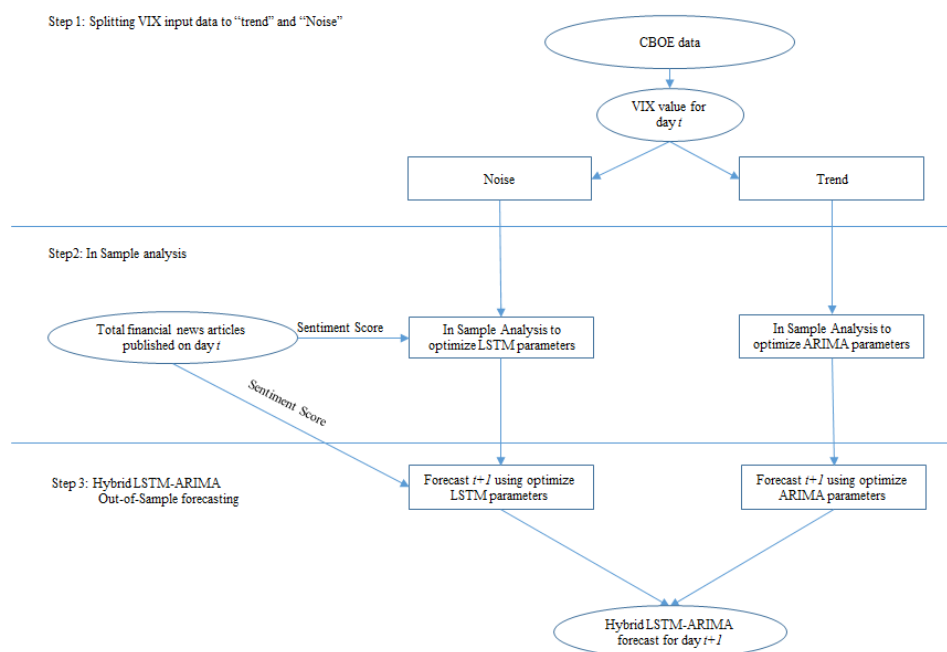


Figure 1: The suggested Hybrid LSTM-ARIMA methodology for forecasting VIX index level in $t+1$

3.2. Splitting VIX Input Data to “Trend” And “Noise” In-Sample and Out-Of-Sample Dataset

LSTM models and the ARIMA models require time series of input data for training, i.e., the values at successive points in time. Our modeling method splits the VIX time series raw data into trend and noise. For setting the trend, we applied the augmented Dickey-Fuller stationary test to detect the stationary level of the time series data and found that all series are integrated at the third difference.

Hence, we used the trend level as the moving average of three days. The noise is the difference between the raw data and the trend data.

Following [25, 26], we define an “in-sample” training period as a set, consisting of a training period of 748 days (approximately 75% of the dataset), which is equivalent to nearly three years, and an “out-of-sample” test period of the succeeding 248 days (25% of the dataset), as shown in Figures 2a-2c.

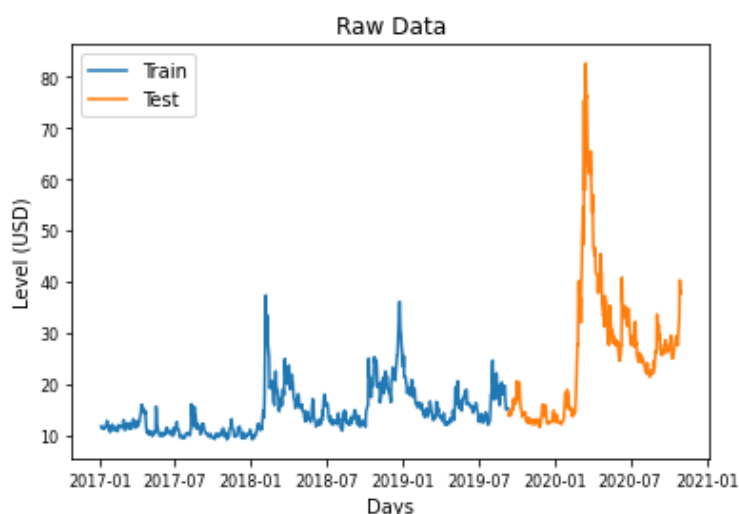


Figure 2a: In-Sample and out-of-sample of VIX raw data

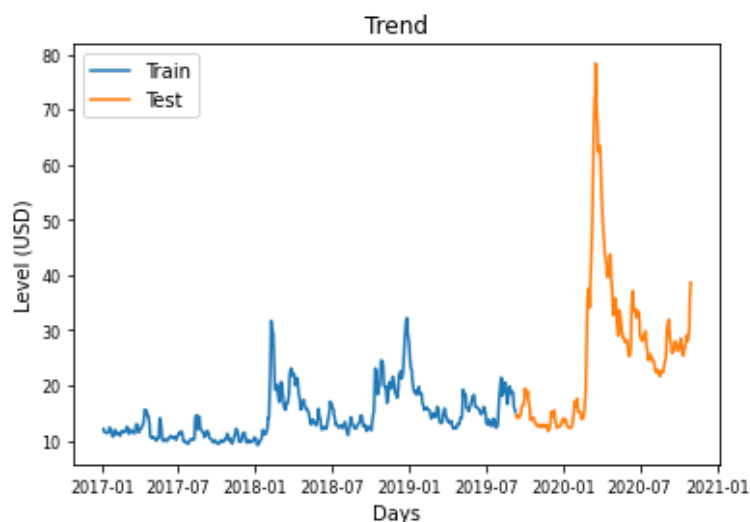


Figure 2b: In-Sample and out-of-sample of VIX trend data

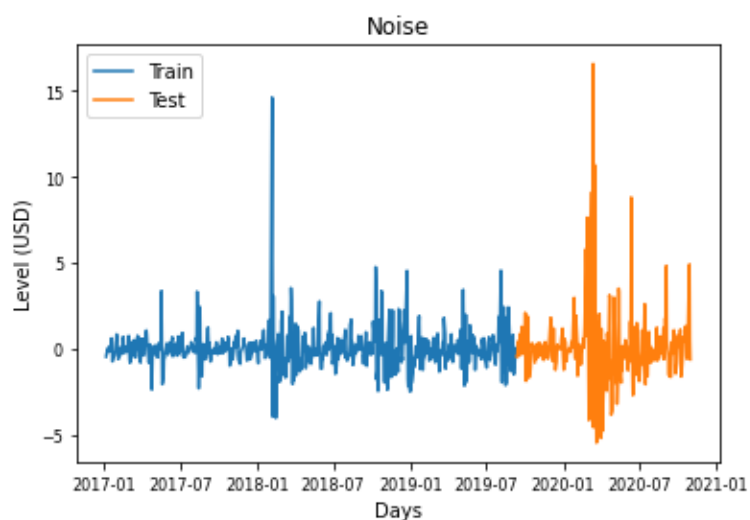


Figure 2c: In-Sample and out-of-sample of VIX noise data

Figures 2a-2c show the VIX index level over time for the raw data, trend data and noise data. The blue line describes the training set (75% of the dataset), while the orange line describes the trading set (25% of the dataset). Figures 2a-2c show that both the training set and the test set consist of significant VIX index. The COVID-19 outbreak during March 2020 associated with the spike in VIX index level is included within the test set. For the in-sample (train) analysis, the input data is used to find the best parameters that calibrate the model and best fit them in terms of the lowest error between theoretical VIX index level and actual VIX index level. In this step, we separate the in-sample

analysis for the LSTM model and the in-sample analysis for the ARIMA model. First, we detail the in-sample process for both models (i.e., LSTM model and ARIMA model), features, and architecture separately. Then, we introduce the hybrid LSTM-ARIMA model.

3.3. The LSTM Model for Estimating VIX Noise

We use the LSTM model to capture nonlinear connection to forecast the VIX noise for day $t+1$. We define VIX “noise” x_t for day t as the difference between the VIX level on day t and the VIX trend and calculated as

The model input is a matrix of sentiment scores from news articles and a sequence of past VIX noise. First, we explain the LSTM model briefly. Second, we explain how to obtain sentiment scores for any given day t and the VIX noise for day t . Finally, we construct model architecture.

LSTM model has been introduced by [13] and further refined in the following years by [27, 28], to name a few. LSTM models are specifically designed to learn long-term dependencies and overcome the previously inherent problems of RNNs, i.e., vanishing and exploding gradients [29].

The hidden layers in LSTM models consist of memory cells. Each of the memory cells has three gates maintaining and adjusting the value its cell state: an input gate, an output gate, and a forget gate. At every time step t each of the three gates acts as filter of the information obtained from the previous layer. For more details, the reader is referred to [26]. Every neural network, such as the LSTM model, has a loss function and an optimizer function. The loss function is the error between the actual output and the predicted output. For accurate predictions, one needs to minimize the calculated error. In a neural network, minimizing loss function is carried using backpropagation. The current error is typically propagated backward to a previous layer, where it is used to modify the weights and biases so that the error is minimized. The weights are adjusted using a function called Optimization Function.

We used the TextBlob library in Python to compute the sentiment (polarity) score for each news article out of the $N = 73,566$ news articles along the 1027 trading days. Let n_t denote the total number of articles published in a day t . In TextBlob, the polarity score $tp_{i,t}$ for the given article text of index $i = 1, 2, \dots, n_t$ in day t is within the range of $[-1, 1]$. If the polarity score is positive, it is regarded as positive sentiment, meaning that the news article is positive in the sense of semantic total positive words. If the polarity score is negative, it is regarded as a negative sentiment, meaning that the news article is negative in the sense of semantic total positive words. If the polarity is equal or close to zero, it is considered neutral. We use the TextBlob library also to compute the subjectivity score $t_{si,t}$ for given article text i on day t . The subjectivity score $t_{si,t}$ is within the range $[0, 1]$, where 0.0 is a very objective article text, and 1.0 is a very subjective article text. In addition, we use the Vader Sentiment Analyzer library in Python to calculate for given article text i on day t . the sentiment score $v_{si,t}$ of each news article. The Vader sentiment score $v_{si,t}$ is calculated by summing each word's scores in the lexicon and then normalized it to be between -1 (most extreme negative) and +1 (most extreme positive). We also use Vader positive score $v_{pi,t}$ from the Vader Sentiment Analyzer library for a given article text index i , which scales the intensity on a scale between -4 (extremely negative) and 4 (extremely positive).

$$TP_t = \sum_{i=1}^{n_t} tp_{i,t}/n_t,$$
 polarity score

$$VS_t = \sum_{i=1}^{n_t} vs_{i,t}/n_t,$$
 subjectivity score

$$TS_t = \sum_{i=1}^{n_t} ts_{i,t}/n_t,$$
 sentiment score

$$VP_t = \sum_{i=1}^{n_t} vp_{i,t}/n_t.$$
 and

Vader positive score

Table 1 presents the statistics for each sentiment score for the total dataset. We calculated the scores based on the daily average score.

Table 1: presents the statistics for each sentiment score for the total dataset. We calculated the scores based on the daily average score.

Sentiment Score Source	Average Score	Median Score	Max Score	Min Score	Std. Dev.
TextBlob Polarity (TP)	0.077	0.078	0.733	-0.265	0.063
TextBlob Subjectivity (TS)	0.395	0.405	0.900	0.000	0.092
Vader Sentiment (VS)	0.530	0.963	1.000	-1.000	0.732
Vader Positive (VP)	0.088	0.085	0.271	0.000	0.034

Table 1 shows that the sentiment score source (TextBlob polarity, TextBob subjectivity, Vader sentiment, and Vader positive) and the corresponding average scores, median score maximum score, minimum score and the standard deviation of the scores. It is shown that the average and median scores are higher than their mid-range absolute theoretical values, meaning that in general, the data relatively positive.

While TextBlob polarity shifts in relatively narrow boundaries, the Vader sentiment score is more diverse, indi-

cating a higher sensitivity for each article's text lexicon. The average subjectivity score suggests that the news contains more objective data than subjective data.

Figures 3a-3d present the correlation between different sentiment scores and VIX level. Each dot in Figures 3a-3d characterizes the TextBlob polarity (TP), TextBob subjectivity (TS), Vader sentiment (VS), and Vader positive (VP) sentiment scores on day t (y-axis) and VIX level on day t (x-axis) for the in-sample data (718 trading days), respectively.

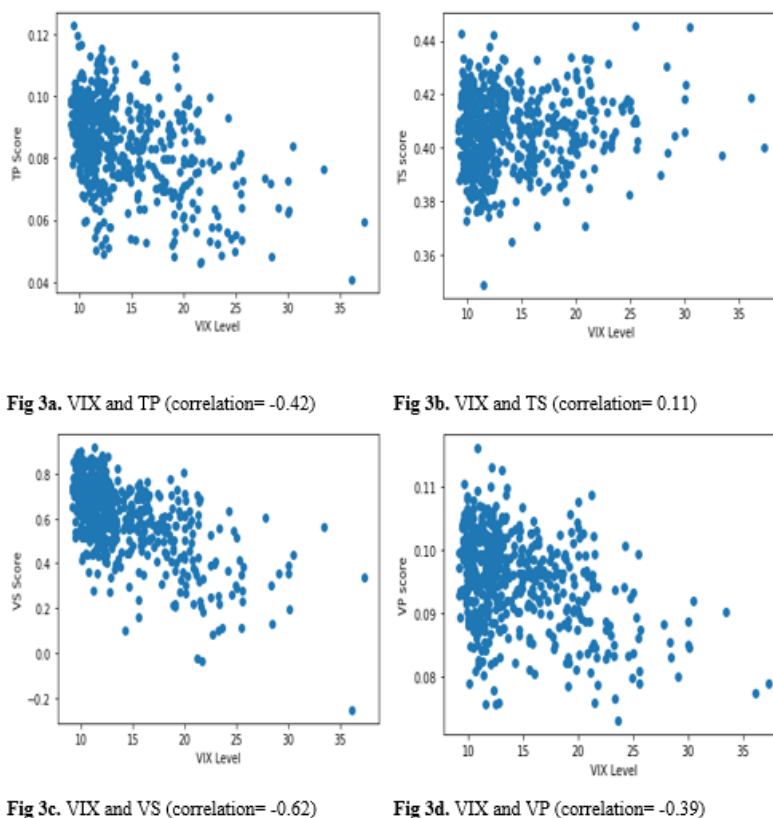


Figure 3. Correlation between different sentiment scores and VIX level.

Using the Pearson correlation coefficient, shown in Figure 3, the correlation between the scaled VIX and TP, VS, and VP are negative. These results align with the literature [30, 31]. The three negative Pearson correlation coefficients express the relation between higher VIX levels and investor sentiment scores. As for the subjective score (TS) shown in Figure 3b, we would expect it to be not necessary correlated to the VIX level since it does not reflect positive

or negative sentiment scores.

3.3.3. The Model Architecture

For any given day t , we define a squared matrix L_t of dimension 5, including input sets of the last five previous trading days (one trading week). Each column includes “noise” of VIX index level and sentiment scores TP, TS, VS, VP.

$$L_t = \begin{bmatrix} x_{t-4} & x_{t-3} & x_{t-2} & x_{t-1} & x_t \\ TP_{t-4} & TP_{t-3} & TP_{t-2} & TP_{t-1} & TP_t \\ TS_{t-4} & TS_{t-3} & TS_{t-2} & TS_{t-1} & TS_t \\ VS_{t-4} & VS_{t-3} & VS_{t-2} & VS_{t-1} & VS_t \\ VP_{t-4} & VP_{t-3} & VP_{t-2} & VP_{t-1} & VP_t \end{bmatrix} \quad (1)$$

This input matrix is used to forecast the VIX index level “noise” $\overline{x_{t+1}}$ for the day for the LSTM algorithm.

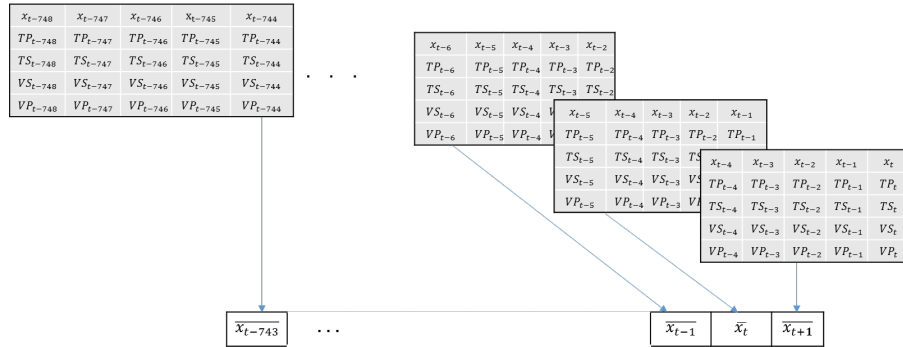


Figure 4: Input matrix for LSTM model

Figure 4 shows the input matrix L_t for estimating VIX index level $\overline{x_{t+1}}$ for the day $t+1$.

Our model topology, as shown in Figure 5, is a “many-to-one” model, meaning that we have a sequence of input parameters L_t and one output result (VIX index level $\overline{x_{t+1}}$). The specified topology of our trained LSTM network is specified above (see Figure 4):

- Input layer L_t .
- Two LSTM hidden layers, each with $h = 30$ hidden neurons and a dropout value of 0.2.
- Output layer (dense layer) with one neuron

representing the forecast for day $t+1$ using the linear activation function.

Following [32], we apply dropout regularization within each of the two hidden layers. Because of this, 20% of the input units are randomly dropped at each update iteration during training time, both at the input gates and the recurrent connections, resulting in reduced risk of overfitting and better generalization.

The training samples were split into two sets: one training set and one validation set. We kept about 25% of the in-sample dataset as a validation set (these samples are assigned randomly to either training or validation set). The first set is used to train the network and iteratively adjust its

parameters to minimize the loss function. The second set of the network predicts the unseen samples from the validation samples and tries to forecast the VIX index level “noise” and validate the selected parameters.

We apply two advanced methods for the LSTM

model training; each of them uses Keras (an open-source neural-network library written in Python). First, we make use of Nesterov accelerated adaptive moment (Nadam) as an optimizer. Second, we use absolute mean error as the loss function in all the experiments, as the absolute mean error produces minimum loss during the training.

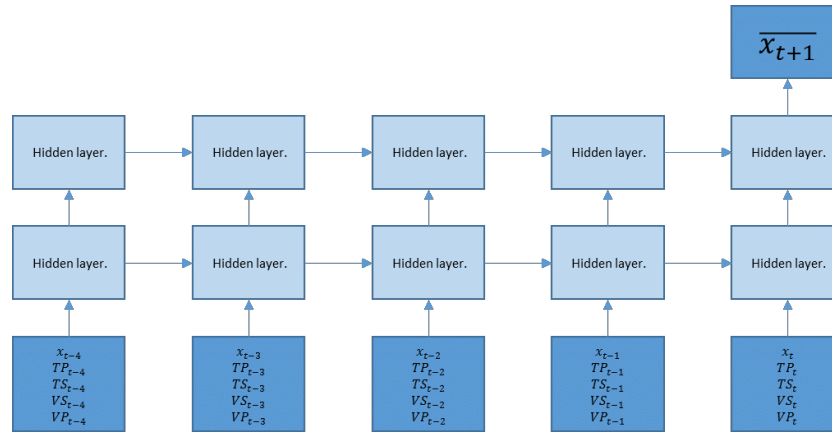


Figure 5: LSTM model architecture

Figure 5 presents the topology of our model, inputs, and outputs for obtaining the optimal parameter weights for the model. There is a total of 15,231 parameter weights estimated for calculating the optimal VIX index level “noise” for $t+1$.

3.4. ARIMA Model for Estimating VIX Trend

The ARIMA model [33, 34] is a generalization of an Auto-Regressive Moving Average (ARMA) model, with an integrated component as a measure of how many non-seasonal differences are needed to achieve stationarity. Both models use time-series data to better understand the data or forecast future points in the series. They are based on a com-

bination of two polynomials, one for the autoregressive part and the other for the moving average part.

An ARIMA model is characterized by three parameters terms: p , d , q where p is the autoregressive term, q is the order of the moving average and d is the order of differencing required to make the time series stationary. For finding the best parameters (p, d, q), we followed [35, 36] and use a stepwise parameter selection to identify the best combination. Next, the model selects the best combination of parameters that can provide minimum Akaike information criterion (AIC) error and is assigned to the best model. Table 2 presents the experimental results for the in-sample data analysis.

Table 2. ARIMA stepwise model and the AIC score

Model	AIC
ARIMA(0,1,0)	1768.901
ARIMA(1,1,0)	1439.735
ARIMA(0,1,1)	1519.731
ARIMA(2,1,0)	1414.122
ARIMA(3,1,0)	1367.891
ARIMA(4,1,0)	1322.92

ARIMA(5,1,0)	1307.757
ARIMA(6,1,0)	1264.347
ARIMA(7,1,0)	1248.154
ARIMA(8,1,0)	1244.324
ARIMA(9,1,0)	1234.646
ARIMA(10,1,0)	1220.775
ARIMA(11,1,0)	1218.251
ARIMA(12,1,0)	1207.355
ARIMA(13,1,0)	1203.675
ARIMA(14,1,0)	1202.932
ARIMA(15,1,0)	1204.582
ARIMA(14,1,1)	1204.853
ARIMA(13,1,1)	1204.448
ARIMA(15,1,1)	1204.394

As shown in Table 2, the optimal model according to AIC, is ARIMA (14, 1, 0), which has the lowest score (1202.932). The optimal ARIMA model and has autoregression of the last 14 trading days. The order of differencing required to make the time series stationary in the optimal ARIMA model is one. The MA element in the optimal ARIMA model is equal to zero. This result is not surprising as the input model data is the trend level which has the moving average level as the input level of the previous three days. For any given day, we set as the input value based in ARIMA (14, 1, 0) model in trading day t to forecast the VIX trend $\overline{\text{MA}_{t+1}}$ for day $t+1$.

3.5. Hybrid LSTM-ARIMA Model

As the ARIMA model has a single forecast for the VIX index level in $t+1$ and LSTM model approach has a single forecast for “noise” from the trend, hybridizing the ARIMA and LSTM forecast for $t+1$ by adding them will provide a forecast to VIX index levels: $\text{VIX}_{t+1} = \overline{x_{t+1}} + \overline{MA_{t+1}}$.

By doing so, we capture both the trend and the specific error derived from the four sentiment scores and the interconnections in the VIX index itself for recent days.

4. Out-of-Sample Empirical Results

For all models, we have used RMSE, MAPE, and

MAE to measure the efficiency of the suggested method in forecasting the actual VIX index level in $t+1$. Low RMSE, MAPE, and MAE scores imply better forecasting.

4.1. Out-of-Sample Performance: Gap Analysis

Our results include three steps. First, we analyze RMSE, MAPE and MAE for the Hybrid LSTM-ARIMA approach. We then analyze each model’s profitability, separately, for both models, and last, we perform a robustness test on the Hybrid LSTM-ARIMA results.

We Compare our Hybrid LSTM-ARIMA with the Following Peer Models:

Hybrid LSTM-ARIMA model without sentiment – This model is similar to the Hybrid LSTM-ARIMA model, but the input parameters do not include the sentiment scores for the LSTM model but only the VIX noise level.

ARIMA model – We used the stepwise methodology for the VIX level data to obtain the optimal ARIMA model. The ARIMA (1, 1, 1) model was the optimal model with the lowest AIC score. These results are similar to other empirical papers like [38] for financial time series.

LSTM model – This is a single LSTM architecture. The model estimates the VIX level based on the input ma-

trix L_t

LSTM model without sentiment– This model is similar to the LSTM model to estimate the VIX level, while the input parameters do not include the sentiment scores for the LSTM model but only the VIX noise level.

Hybrid GARCH-ARIMA – the model estimates the trend level with ARIMA (14, 1, 0) model, and the VIX noise with GARCH (1, 1) model.

Table 3 shows the results for out-of-sample forecasting performance accuracy. We used RMSE and MAPE and MAE, following [37] Zhu and Lian (2012) and others.

Table3: Out of sample results in 6 different models

MAE	MAPE	RMSE	Model
1.97112	0.431938	3.36909	Hybrid LSTM-ARIMA with sentiment
1.94572	0.435897	3.40039	Hybrid LSTM-ARIMA without sentiment
2.00947	0.440263	3.5442	ARIMA (1,1,1)
5.81984	0.306023	10.546	LSTM with sentiment
5.42987	0.308665	10.0932	LSTM without sentiment
2.00147	0.439839	3.42759	Hybrid ARIMA-GARCH

VIX prediction

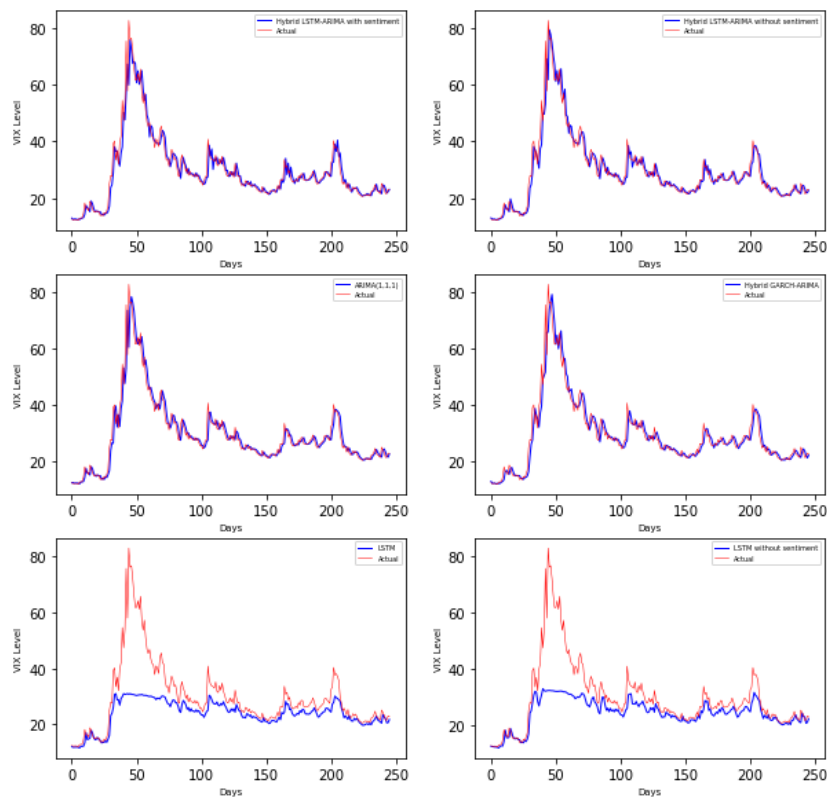


Figure 5: VIX Index level forecast for hybrid models with (and without) sentiment compared to VIX actual index levels

Figure 5 shows that except for the LSTM models

solely, the VIX index level forecast is very close to actual val-

ues, even during the COVID-19 trading days, which had extremely high spikes.

Interestingly, prior to COVID-19 pandemic, all models show high accuracy to the actual VIX levels. However, during COVID-19 trading days, the LSTM component didn't capture the higher volatility. We conclude that investor sentiment improves the performance of the LSTM component, while ARIMA stabilizes short-term fluctuations. This combination illustrates the strength of hybrid soft computing approaches, though further validation across calm and turbulent market regimes is needed.

4.1. Out-of-Sample Performance Analysis - Trading Strategy

VIX is traded with futures contracts in CBOE.

VIX futures contracts provide market participants with the ability to trade a liquid volatility product based on the VIX Index. In our model, we assume that we can trade at the VIX market price, zero Bid-Ask spread and no transaction fees. For trading strategy, at $t+1$:

- According to the model, if $VIX_{model, t+1} > VIX_t$, buy the VIX index and the profit (loss) on day $t+1$ will be
- According to the model, if sell the VIX index and the profit (loss) on day $t+1$ will be
- According to the model, if do not trade.

We present in Table 4 the empirical trading strategy results before any transaction costs.

Table 4: Trading strategy results for each model

Model	% profitable transactions	% Buy	Profit (\$)
Hybrid LSTM-ARIMA with sentiment	0.5673	0.4734	139.91
Hybrid LSTM-ARIMA without sentiment	0.5387	0.4693	87.51
ARIMA (1,1,1)	0.5306	0.5428	80.11
LSTM with sentiment	0.551	0	-10.21
LSTM without sentiment	0.563	0.012	20.71
Hybrid ARIMA-GARCH	0.5346	0.4979	62.37

Table 4 shows that Hybrid LSTM-ARIMA with sentiment achieves the strongest simulation-based forecasting performance, reflected in the highest outcome values

among tested models. These results are presented under idealized conditions (ignoring transaction costs and liquidity effects) and should be seen as indicative rather than directly implementable. The cumulative profit is shown in Figure 6.

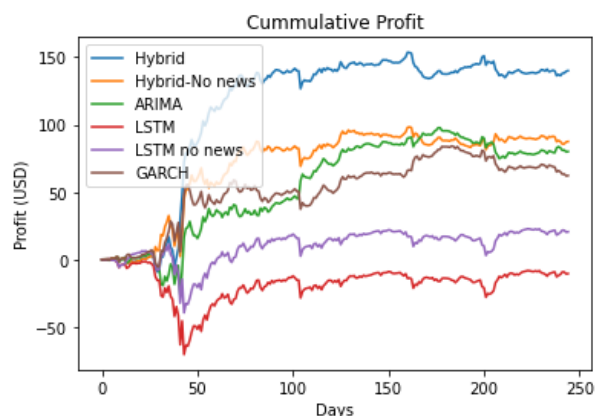


Figure 6: VIX cumulative profit (in \$) for all tested models

Figure 6 shows that the hybrid LSTM with sentiment obtains 29% more profit than the hybrid LSTM without sentiment, mainly during COVID-19 volatile time. The inclusion of the COVID-19 period is interesting because it provides insight into the power of the model even in the face of unexpected shocks.

4.2. Robustness Test Results

Several methods to compute robustness quantification of several neural networks for Out-of-Sample data were presented in the literature (See, for example, Deng et al., 2016; KO Et Al., 2019). Yet, the robustness quantification of LSTM models for out-of-sample data remains an open problem because of the complexity of its architecture. To quantify robustness, we made the following robustness ran-

dom shuffling test Hybrid LSTM-ARIMA with the sentiment on the Out-of-Sample data. By doing so, we address the main vulnerability of LSTM models: The adversarial attack-based approach, which means researchers design strong adversarial attack algorithms to attack deep neural networks. Robustness is measured by the distortion between successful adversarial examples and the original ones.

The following three Hybrid LSTM-ARIMA models are baseline Hybrid LSTM-ARIMA model with the sentiment, baseline Hybrid LSTM-ARIMA model with VIX data randomly shuffle, baseline Hybrid LSTM-ARIMA model with both VIX and the sentiment score values data are randomly shuffled. By shuffling the data, we can assure that the results are not arbitrary. Table 5 shows the robustness test results for the three models.

Table5: Robustness test for out-of-sample hybrid LSTM-ARIMA baseline model

% profitable transaction	% Buy	Profit (\$)	MAE	MAPE	RMSE	Model
0.5673	0.4734	139.91	1.97112	0.431938	3.36909	Hybrid LSTM-ARIMA
0.5469	0.5061	70.27	2.23505	0.441527	3.63338	Robustness test 1 – Only VIX
0.5387	0.412	51.81	2.25772	0.4408	3.69493	Robustness test 2 – VIX and sentiment

Table 5 shows that the baseline Hybrid LSTM-ARIMA model (no data shuffle) performs better than the shuffled-data variants across RMSE, MAE, and performance outcomes. This suggests the model extracts meaningful information from sentiment and VIX data, although robustness across additional datasets and domains remains to be established.

5. Conclusion

This paper introduces a new forecasting model of VIX index returns for the next day based on both LSTM and ARIMA models. We developed the hybrid LSTM-ARIMA model, which considers investors' sentiment scores. To the best of our knowledge, this study is the first to propose and implement a hybrid LSTM-ARIMA model and to incorporate investors' sentiment analysis in the model.

The sentiment scores are empirically evaluated

based on commonly used daily article text major economic sites. The forecasts of next day VIX index level based on out-of-sample for 2019-2020 end of day data present greater robust results compared to models without sentiment parameters. We found that hybrid LSTM-ARIMA with sentiment obtains the lowest RMSE, while the LSTM model (with or without sentiment) obtains inferior results, mainly in the COVID-19 period. We relate these outcomes to the fact that the LSTM model has a positive bias towards actual prices.

The model introduced in this paper has significant advantages and implications for trading the VIX index. The model contributes to the soft computing literature by combining ARIMA and LSTM with sentiment analysis in a hybrid forecasting framework. The model also achieves greater proximity to the actual VIX index levels and higher simulated forecasting performance than the comparative methods tested. Beyond financial markets, this hybrid soft computing approach could be extended to other domains

with nonlinear, noisy, and sentiment-influenced time series such as energy demand, healthcare signals, or supply chain volatility. Importantly, our study's main conclusion is that sentiment analysis improves forecasting compared to hybrid LSTM-ARIMA without sentiment.

Our work has several limitations, which may be addressed in future research. The dataset included only investors' sentiment and VIX index levels as the input vector. In particular, we did not consider parameters such as trading volumes in the VIX index or in the S&P500 index to increase proximity or profit. Our analysis was made on end-of-day data, however it would be scientifically interesting to extend the work and perform the analysis of intraday data.

Future research should incorporate intraday data, trading volumes, and realistic frictions (transaction costs, bid-ask spreads, liquidity effects). Reporting risk-adjusted performance measures (such as Sharpe ratio and maximum drawdown) would further clarify the practical applicability of the model. This work demonstrates how combining statistical, deep learning, and NLP components within a hybrid

soft computing framework can address noisy and nonlinear time series across diverse domains

Declarations

Funding

This research did not receive any specific grant from funding agencies in the public, commercial, or not-for-profit sectors.

Declaration of Competing Interest

The authors have no competing interests to declare that could have appeared to influence the work reported in this paper.

Data Availability

The data supporting the findings of this study are available from the corresponding author on reasonable request.

References

1. Jin Z, Yang Y, Liu Y (2019) Stock closing price prediction based on sentiment analysis and LSTM. *Neural Computing and Applications*. 1-17.
2. Baek S, Mohanty SK, Glambosky M (2020) COVID-19 and stock market volatility: An industry level analysis. *Finance Research Letters*. 37: 101748.
3. Lalancette S, Simonato JG (2017) The role of the conditional skewness and kurtosis in VIX index valuation. *European Financial Management*. 23: 325-54.
4. Taylor N (2019) Forecasting returns in the VIX futures market. *International Journal of Forecasting*, 35: 1193-210.
5. Psaradellis I, Sermpinis G (2016) Modeling and trading the US implied volatility indices. Evidence from the VIX, VXN and VXD indices. *International Journal of Forecasting*. 32: 1268-83.
6. Qiao G, Yang J, Li W (2020) VIX forecasting based on GARCH-type model with observable dynamic jumps: A new perspective. *The North American Journal of Economics and Finance*. 53: 101186.
7. Wang T, Shen Y, Jiang Y, Huang Z (2017) Pricing the CBOE VIX futures with the Heston–Nandi GARCH model. *Journal of Futures Markets*. 37: 641-59.
8. Guo S, Liu Q (2020) Efficient out-of-sample pricing of VIX futures. *The Journal of Derivatives*, 27: 126-39.
9. Heston SL (1993) A closed-form solution for options with stochastic volatility with applications to bond and currency options. *The Review of Financial Studies*. 6: 327-43.
10. Yang X, Wang P (2018) VIX futures pricing with conditional skewness. *Journal of Futures Markets*. 38: 1126-1151.
11. LeBaron B (2000) Agent-based computational finance: Suggested readings and early research. *Journal of Economic Dynamics and Control*. 24: 679-702.
12. Qi M, Wu Y (2006) Technical trading-rule profitability, data snooping, and reality check: Evidence from the foreign exchange market. *Journal of Money, Credit and Banking*: 2135-58.
13. Hochreiter S, Schmidhuber J (1997) Long short-term memory. *Neural Computation*. 9: 1735-80.
14. Koreisha SG, Fang Y (1999) The impact of measurement errors on ARMA prediction. *Journal of Forecasting*. 18: 95-109.
15. Siami-Namini S, Tavakoli N, Namin AS (2018) A comparison of ARIMA and LSTM in forecasting time series. In 2018 17th IEEE International Conference on Machine Learning and Applications. 1394-1401.
16. Khashei M, Bijari M (2011) A novel hybridization of artificial neural networks and ARIMA models for time series forecasting. *Applied Soft Computing*. 11: 2664-75.
17. Kim SH, Kim D (2014) Investor sentiment from internet message postings and the predictability of stock returns. *Journal of Economic Behavior & Organization*. 107: 708-729.
18. Xu Y, Cohen SB (2018) Stock movement prediction from tweets and historical prices. In Proceedings of the 56th Annual Meeting of the Association for Computational. 1: 1970-9.
19. Yang L, Zhang Z, Xiong S, Wei L, Ng J, et al. (2018) Explainable text-driven neural network for stock prediction. In 2018 5th IEEE International Conference on Cloud Computing and Intelligence Systems. 441-5
20. McKinney W (2010) Data structures for statistical computing in Python. In Proceedings of the 9th Python in Science Conference. 51-6
21. Abadi M, Barham P, Chen J, Chen Z, Davis A, et al. (2016) Tensorflow: A system for large-scale machine learning. In 12th Symposium on Operating Systems Design and Implementation 16: 265-83
22. Loria S (2018) Textblob Documentation. Release 0.15, 2.
23. Hutto CJ, Gilbert E (2014) Vader: A parsimonious rule-based model for sentiment analysis of social media text. In Eighth international AAAI conference on weblogs and so-

cial media.

24. Urologin S (2018) Sentiment Analysis, Visualization and Classification of Summarized News Articles: A Novel Approach. *IJACSA0 International Journal of Advanced Computer Science and Applications*. 9
25. Kraus M, Feuerriegel S (2017) Decision support from financial disclosures with deep neural networks and transfer learning. *Decision Support Systems*.104: 38-48.
26. Fischer T, Krauss C (2018) Deep learning with long short-term memory networks for financial market predictions. *European Journal of Operational Research*. 270: 654-69.
27. Gers FA, Schmidhuber J, Cummins F (2000) Learning to forget: Continual prediction with LSTM. *Neural Computation*. 12: 2451-71.
28. Graves A, Schmidhuber J (2005) Framewise phoneme classification with bidirectional LSTM and other neural network architectures. *Neural Networks*. 18: 602-10.
29. Sak H, Senior AW, Beaufays F (2014) Long short-term memory recurrent neural network architectures for large scale acoustic modeling. *Fifteenth annual conference of the international speech communication association*.
30. Xu Y, Cohen SB (2018) Stock movement prediction from tweets and historical prices. In *Proceedings of the 56th Annual Meeting of the Association for Computational*. 1: 1970-9.
31. Yang L, Zhang Z, Xiong S, Wei L, Ng J, et al. (2018) Explainable text-driven neural network for stock prediction. In *2018 5th IEEE International Conference on Cloud Computing and Intelligence Systems*. 441-5.
32. Gal Y, Ghahramani, Z (2015) A theoretically grounded application of dropout in recurrent neural networks. *arXiv preprint arXiv:1512.05287*.
33. Cline DB, Brockwell PJ (1985) Linear prediction of ARMA processes with infinite variance. *Stochastic Processes and their Applications*, 19: 281-96.
34. Arun KE, Kalaga DV, Kumar CMS, Chilkoor G, Kawaji M, et al. (2021) Forecasting the dynamics of cumulative COVID-19 cases (confirmed, recovered and deaths) for top-16 countries using statistical machine learning models: Auto-Regressive Integrated Moving Average (ARIMA) and Seasonal Auto-Regressive Integrated Moving Average (SARIMA). *Applied Soft Computing*. 103: 107161.
35. Malki Z, Atlam ES, Ewis A, Dagneu G, Alzighaibi AR, et al. (2021) ARIMA models for predicting the end of COVID-19 pandemic and the risk of second rebound. *Neural Computing and Applications*. 33: 2929-48.
36. Musa Y, Joshua S (2020) Analysis of ARIMA-Artificial Neural Network Hybrid Model in forecasting of stock market returns. *Asian Journal of Probability and Statistics*. 6: 42-53.
37. Zhu SP, Lian GH (2012) An analytical formula for VIX futures and its applications. *Journal of Futures Markets*, 32: 166-90.
38. Ahoniemi K (2006) Modeling and forecasting implied volatility-an econometric analysis of the VIX index. Helsinki: Helsinki Center of Economic Research.
39. Chollet F (2015). Keras, GitHub. <https://github.com/fchollet/keras>
40. Deng Y, Bao F, Kong Y, Ren Z, Dai Q (2016) Deep direct reinforcement learning for financial signal representation and trading. *IEEE Transactions on Neural Networks and Learning Systems*. 28: 653-64.
41. Froot KA, Scharfstein DS, Stein JC (1992) Herd on the street: Informational inefficiencies in a market with short-term speculation. *The Journal of Finance*. 47: 1461-84.
42. Hao J, Zhang JE (2013) GARCH option pricing models, the CBOE VIX, and variance risk premium. *Journal of Financial Econometrics*. 11: 556-80.
43. Walt SVD, Colbert SC, Varoquaux G (2011) The NumPy array: a structure for efficient numerical computation. *Computing in Science & Engineering*. 13: 22-30.
44. Zhang GP (2003) Time series forecasting using a hybrid ARIMA and neural network model. *Neurocomputing*. 50: 159-75.

Submit your manuscript to a JScholar journal and benefit from:

- ¶ Convenient online submission
- ¶ Rigorous peer review
- ¶ Immediate publication on acceptance
- ¶ Open access: articles freely available online
- ¶ High visibility within the field
- ¶ Better discount for your subsequent articles

Submit your manuscript at
<http://www.jscholaronline.org/submit-manuscript.php>