- ORIGINAL ARTICLE -

A Novel NLP-based Stock Market Price Prediction and Risk Analysis Framework

Un Novedoso Framework basado en PLN para Análisis de Riesgos y Predicción de Precios Bursátiles

Zain-ul-Abideen[1], Raja Hashim Ali[1,2], Ali Zeeshan Ijaz[1], and Talha Ali Khan[2]

[1]*A.I. Research Group, Faculty of Computer Science and Engineering, Ghulam Ishaq Khan Institute of Engineering Sciences and Technology, 23460 Topi, Khyber Pakhtunkhwa, Pakistan.*
Zain-ul-abideen@giki.edu.pk (Z.A.); hashim.ali@giki.edu.pk (R.H.A.); ali.zeeshan@giki.edu.pk (A.Z.I.)
[2]*Department of Technology and Software Engineering, University of Europe of Applied Sciences, Potsdam, Germany*
hashim.ali@ue-germany.de (R.H.A.); talhaali.khan@ue-germany.de (T.A.K.)

Abstract

The prediction of stock market prices represents a significant challenge due to its volatile nature, influenced by unpredictable economic factors, company performance, and market sentiment. The assurance of these forecasts or the associated risk with these price estimations plays a pivotal role in the decision-making process. Existing models have either focused on stock price prediction or risk analysis but rarely integrate both, leaving a gap in providing a comprehensive tool for investors. In the current work, we present a novel framework for investment analysis designed to create ease for investors and provide a confidence measure along with the stock price to depict the risk involved in investing in stocks of a particular company. The model uses a stock price dataset depicting the original scores as numerals and textual data extracted from Reddit news articles as input. The stock price is predicted by LSTMs on individual stock prices, while the confidence is represented by a risk value calculated using XGBoost and LSTM output. We performed sentiment analysis and subjectivity analysis to extract features for further investigation in the study. The results show that an accuracy of 94% for stock trend prediction can be achieved using PCA as the feature extractor with tuned parameters for XGBoost and around 76% accuracy for stock price prediction with a tuned LSTM. Our study demonstrates the effective integration of risk analysis with stock price forecasting, illustrating that deep learning techniques are suitable for melding risk assessment with the prediction of stock prices.

Keywords: Long Short-Term Memory (LSTM) network, Reddit, Natural Language Processing, Deep Learning, Stock Price Analysis

Resumen

La predicción de los precios del mercado de valores representa un desafío importante debido a su naturaleza volátil, influenciada por factores económicos impredecibles, el desempeño de las empresas y el sentimiento del mercado. La seguridad de estas previsiones o el riesgo asociado a estas estimaciones de precios juega un papel fundamental en el proceso de toma de decisiones. Los modelos existentes se han centrado en la predicción del precio de las acciones o en el análisis de riesgos, pero rara vez integran ambos, lo que deja una brecha a la hora de proporcionar una herramienta integral para los inversores. En el trabajo actual, presentamos un marco novedoso para el análisis de inversiones diseñado para facilitar a los inversores y proporcionar una medida de confianza junto con el precio de las acciones para representar el riesgo que implica invertir en acciones de una empresa en particular. El modelo utiliza un conjunto de datos de precios de acciones que representa las puntuaciones originales como números y datos textuales extraídos de artículos de noticias de Reddit como entrada. Los LSTM predicen el precio de las acciones sobre los precios de las acciones individuales, mientras que la confianza está representada por un valor de riesgo calculado utilizando la salida de XGBoost y LSTM. Realizamos análisis de sentimiento y análisis de subjetividad para extraer características para una mayor investigación en el estudio. Los resultados muestran que se puede lograr una precisión del 94% para la predicción de la tendencia de las acciones utilizando PCA como extractor de características con parámetros ajustados para XGBoost y alrededor del 76% de precisión para la predicción del precio de las acciones con un LSTM ajustado. Nuestro estudio demuestra la integración efectiva del análisis de riesgos con la previsión de precios de las acciones, lo que ilustra que las técnicas de aprendizaje profundo son adecuadas para fusionar la evaluación de riesgos con la predicción de los precios de las acciones.

Palabras claves: Red de memoria larga a corto plazo (LSTM), Reddit, procesamiento del lenguaje natural, aprendizaje profundo, análisis del precio de las acciones

1 Introduction

Forecasting has gained prominence in recent times, with several applications in decision-making in many vital fields, e.g., in the energy sector [1], in the spread of pandemic diseases [2], flood forecasting [3] and motion forecasting in autonomous vehicles [4], etc. Forecasting is concerned with the prediction of future events using historical data, and is of three types [5]: Short term forecasting, which predicts events within a few seconds to some weeks; medium-term forecasting, which is concerned with the prediction of events to occur within the next one to two years; and Long term forecasting that predicts events happening after more than 2 years. A significant application of forecasting is in the field of stock market analysis, which employs various tools to analyze the stock market data and look for trends, resulting in predictions of stock prices going up or down in the future [6, 7]. From the viewpoint of investment, such an analysis helps users identify stocks that are low risk and may give high profits in the future. However, it is difficult to predict the stock markets because of several uncertainties, such as general economic conditions and sociopolitical factors at regional, national and international levels [8]. The central theme for working in stock market analysis and forecasting is to help the end user understand which stocks are expected to go up (or down) soon so they can buy (or sell) stock shares in the market.

Several classical machine learning models [9, 10, 11] have traditionally been applied for forecasting stock prices [12] under different scenarios. Javed *et al.* [13] implemented linear regression, random forest and decision tree techniques to study the stocks of ten top companies, including historical stock prices, to reach an accuracy between 80% to 98% for fluctuation in stock prices. A hybrid machine learning model, based on multi-layer perceptron, was applied to preprocessed stock price data for five years between 2012 and 2017 and used financial ratios of Technology companies listed on Nasdaq to attain an accuracy of around 66% [14]. Similarly, a novel Bayesian neural network approach implementing the Markov chain Monte Carlo (MCMC) sampling technique [15, 16] was used to make reasonable predictions for stock prices despite the high volatility during the first peak for Covid-19 [17]. Kumar *et al.* [18] deployed a novel prediction model based on a radial basis function network and compared it with a single hidden layer feed-forward neural network to infer the stock market prices. Garcia *et al.* [19] employed a two-stage approach utilizing both generalized regression neural networks (also called kernel regression) and kernel adaptive filtering (KAF) to sequentially predict the stock prices and showed a higher Sharpe ratio as compared to several other methods. Moreover, k-Nearest Neighbour (kNN) regression, Naive Bayes, Support Vector Machine (SVM), Random Forest, and Softmax were deployed by Kumar *et al.* [20] to compare the performance of these traditional machine learning methods to predict the prices of stocks.

On the contrary, several deep learning-based architectures [21, 22, 23] have gained prominence for making predictions. These approaches promise a higher accuracy than the traditional machine learning mechanisms for regression analysis [24, 25, 26]. In that aspect, Recurrent Neural Networks, especially Long-Short Term Memory (LSTM) networks, has shown promising results [27] and has become the de-facto state-of-the-art method to employ for forecasting [28]. Therefore, stock market analysis is one of the obvious applications of forecasting and one of the critical areas in which deep learning methods and models have shown promising results [29].

The availability and use of datasets are one of the main issues with stock price forecasting. Historically, forecasts have been made using historical data from various financial markets. Numerous stock markets have provided data sets that cover multiple days, weeks, months, or years of stock prices, which have been utilized in multiple studies for analysis and forecasting [30]. The New York Stock Exchange, Tokyo Stock Exchange, Australian Stock Exchange, Nasdaq, London Stock Exchange, and Karachi Stock Exchange are just a few of the stock exchanges from which historical stock price datasets have been frequently obtained. Moreover, Reddit news articles, tweets, and information from other social media can also impact the stock prices, and therefore, is an ancillary data for stock price prediction [31]. Typically, data are extracted from social media feeds using natural language processing techniques. The features may then be fed into a machine learning classifier or predictor to anticipate the class or predicted price of the stock after a given period.

Risk analysis is another essential feature of stock prediction that quantifies the degree of certainty in stock price predictions and ought to be considered for each inference drawn. However, very little research has been done to measure the confidence and risk associated with stock price predictions. Most studies do not include confidence values in their inference. Additionally, there hasn't been much research that has merged data from social media and social market sources, despite the abundance of studies that have used deep learning techniques on multiple datasets. Platforms like Investopedia and MSN Finance have considered risk analysis. Yet, they do not present the crucial insights from risk analysis, maximum profit statement, and/or social media analysis to the investor. Therefore, a new framework is required based on multiple sources, which uses risk analysis in conjunction with prediction of change in stock prices.

To fill this gap, we introduce a framework designed specifically for investors keeping in mind their requirements in this study. The framework is built on soft-

ware engineering principles with requirement gathering, use case dia-grams, and other software design requirements. We have performed social media, news, and sentiment analysis to determine the different indicators for this framework. We have used LSTM and XGBoost-based deep learning models for the machine learning model, which have been trained and tested on merged data from Reddit sources and stock price data. The system is evaluated on all necessary metrics to facilitate the investor in knowing how safe the investment is and what the profit margin is after a certain amount of time on the investment. While evaluating a particular stock, the investors have complete transparency and autonomy over the system. The system provides investors with the necessary background analysis. In comparison to the existing systems forecasting stock price prediction, our system incorporates social media analysis in addition to the stock price data, as well as performs risk analysis that are generally lacking in an existing system. With an accuracy of approximately 94.1% on the validation merged data set, the system can identify if a stock price is expected to go up or down and associate the model's confidence with the prediction using a risk level indication. The study successfully shows how a reliable, simple, and dependent model can analyze the risk involved and whether the stock price will increase or decrease soon to help secure the investment.

Our main contributions and novelty are summarized in Figure 1. Our model is the first attempt to merge data sources consisting of actual stock values and Reddit data for stock price prediction. Then a novel hybrid pipeline composed of results of XGBoost and LSTM was applied to this dataset after data preprocessing. Finally, the risk analysis component is another uniqueness of this work which acts as a proxy for how much the system is confident in its predictions. Another contribution is the advanced machine learning and deep learning techniques for forecasting the change in stock values and simultaneously in the risk evaluation.

2 Background

This section discusses the fundamental concepts necessary for predicting stock prices. Since we have used LSTMs in this study, we briefly review the applications of LSTMs for stock price prediction. The literature review inspired our methodology and settings for LSTM and decision trees (Extreme Gradient Boosting XGBoost).

2.1 Types of Models

Note that the following three models are generally used for stock market prediction.

2.1.1 Fundamental Analysis

Fundamental analysis is an investment analysis technique where economic factors such as sales, profits, earnings, etc., are used, which has shown impressive results for problems categorized as Long-Term prediction.

2.1.2 Technical Analysis

Short-term predictions are made using Technical Analysis. Typically, people use a moving average for such analysis.

2.1.3 Time Series Analysis

The Time Series method has two further models: the linear Model – one of which is variance-variance compo-site-based method known as Auto-Regressive Integrated Moving Average (ARIMA); However, it does not account for the dynamic trends. The other way is to use nonlinear models – such as Long Short-Term Memory (LSTM). The LSTM is, in general, a suitable algorithm that uses LSTM cells in place of hidden layers (thus deviating from general Recurring Neural Network (RNN) Models), which allows it to explore hidden trends. The model judges which input to keep and which to drop based on the Sigmoid activation function, and the tanh activation function is used to generalize the inflow of data.

2.2 Types of LSTMs

As discussed earlier, the literature generally shows a better performance of LSTM than other techniques for predicting stock prices and analyzing hidden trends, both in Long Term and Short Term. Note that three types of LSTMs are discussed in the literature.

2.2.1 Simple LSTM

Simple LSTM was developed to extend the memory of RNN, for it works for a more extended sequence of inputs.

2.2.2 Bidirectional LSTM

Bidirectional LSTM uses trends in two ways; first, it traverses the trends to look for patterns in ascending order of dates, and then it does the same in descending order to find any hidden pattern, hence the name Bi-Directional LSTM.

2.2.3 Stacked LSTM

The third type, Stacked LSTM, uses multiple stacks of LSTM with their outputs linking to the inputs of the other LSTM. This is done to explore more complex trends in the data.

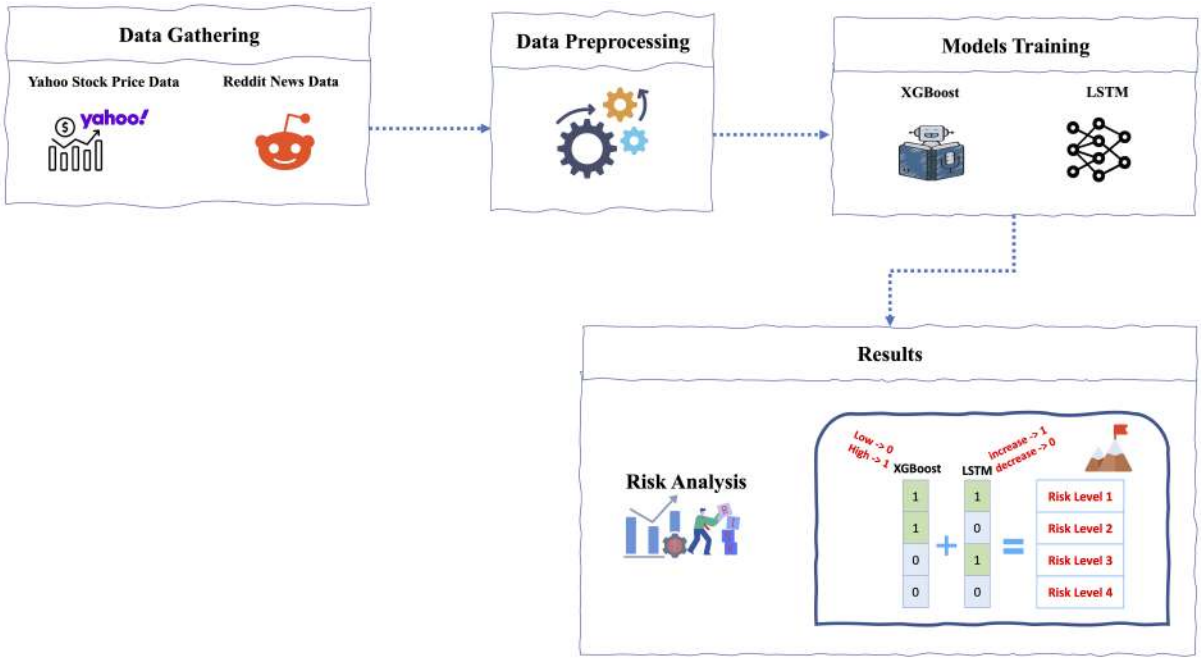

Figure 1: Summary of this work and our contribution to Stock Price Prediction – The figure summarizes our unique contributions in the field of stock price prediction, where we have successfully merged two different sources (Reddit news articles and actual stock prices) to infer the price of stocks using a novel hybrid approach with LSTMs and XGBoost. We also combined stock price prediction with risk analysis, which is lacking in contemporary methods.

3 Literature Review

3.1 Linear vs. Nonlinear Models

In a study by Selvin *et al.* [32], traditional machine learning algorithms performed well when identifying short-term trends but generally failed to identify long-term patterns and trends. The authors compared non-linear models (deep learning models including RNN, Convolutional Neural Networks (CNN) and LSTM) and linear models (autoregressive (AR), moving average (MA), and Auto Regressive Integrated Moving Average (ARIMA)) and recorded their per-formance on short term, medium term and long-term stock data taken from three companies (InfoSys, Cipla, and TCS) using error percentage as the evaluation metric. Their analysis showed that all deep learning-based models performed significantly better than the linear models, as shown in Figure 2, where a lower error percentage depicts a better performance.

A similar yet more comprehensive work has been done by Chhajer *et al.* [33], who performed a detailed review of literature for Artificial Neural Networks, a linear classifier (SVM), and LSTM techniques. Although a direct comparison has not been performed in the paper, a closer view of the reported accuracy reported for different papers suggests that LSTMs and ANNs perform significantly better than their linear counterparts.

Figure 2: Performance of Sliding Window vs. Deep Learning Methods – A lower error percentage indicates a better method. The deep learning models show significantly better performance than ARIMA.

3.2 Using Hybrid Models

Predicting the equivalent real currency for the virtual currency (cryptocurrency) is also as big a challenge as predicting the stock prices. The proposed models can theoretically be applied for stock market analysis [34]. Jay *et al.* [35] coupled a neural network with a stochastic process to predict the price of multiple cryptocurrencies (Bitcoin, Ethereum, and Litecoin) and showed more accurate results than the deterministic models. Similarly, Parekh *et al.* [36] used LSTM and VADER algorithms to forecast the price based on

historical cryptocurrency data as well as the sentiment analysis of tweets. However, in both studies, the dependence between cryptocurrencies is not considered. This drawback was addressed by Schumaker *et al.* [37], who used graph convolutional networks (GCN) and LSTMs for stock prediction over multiple metrics, e.g., the overall stock market trend and the mutation point among different markets. Similarly, Ding *et al.* [38] also deployed LSTMs to predict stocks simultaneously over multiple metrics.

3.3 Using Textual Analysis for Stock Price Prediction

As opposed to the works done directly on stock prices by Chhajer *et al.* [33], Schumaker *et al.* [37] used textual analysis on 9,211 financial news articles and over 10 million stock quotes using SVMs with an accuracy of around 57%. The authors also compared several representations of texts, including noun phrases, named entities, and bag-of-words. They concluded that the bag-of-words is not enough for stock price prediction. In another similar approach, Huang *et al.* [39] used contextual information from the news to predict stock prices using RNN. However, Akita *et al.* [40] noted that RNNs perform well when applied to a medium-term problem but performed poorly, especially on long-term problems, as they could not predict the long-lasting effects of words such as 'financial crisis'. However, LSTMs could capture trends using variable input data (text analysis) and performed significantly better for long-term forecasting. Therefore, LSTMs were proposed for stock price prediction, especially when working with texts.

3.4 Using Sentiment Analysis for Stock Price Prediction

For stock price forecasting, a different approach to predicting stock prices is using sentiment analysis on textual data, e.g., tweets or news articles. Mishev *et al.* [41] deployed sentiment analysis for stock price forecasting. They used Natural Language Processing (NLP), starting with simple, specific financial lexicons and then upgrading the complexity of learning by including sentiments based on words, sentences and finally, the NLP transformers. Xu *et al.* [42] deployed reinforcement learning (RL)-based gated recurrent unit (GRU) network and contextual semantics in the news for stock prediction. The authors used addition and dot operation based on two attention mechanisms to decrease the significant amount of noise in the raw data to get more accurate results.

3.5 Using LSTMs for Stock Price Prediction

Althelaya *et al.* [43] compared the performance of bidirectional LSTMs and stacked LSTMs for short-term and long-term prediction on stock data. Interestingly, bi-directional LSTM performed slightly better than stacked LSTM, and both performed well for making short-term predictions. The recent literature consists of papers on bi-directional LSTMs for stock prediction, where Chen *et al.* [44] proposed a novel hybrid Bi-LSTM-based model for stock price prediction, while Chen *et al.* [45] also used attention networks and bi-directional LSTM for attentive multi-view news learning (NNML), and stock's con-textual information respectively. Similarly, Wu *et al.* [46] developed a hybrid framework for stock market prediction using auto-encoders, a powerful learning machine, and a discrete wavelet transform that performs 5% better (on average) than its singular counterparts. Moreover, Ding *et al.* [38] designed another hybrid technique based on CNN (for feature extraction), Bi-LSTM (for stock prediction) and attention mechanism (for capturing feature effect in stocks). As shown earlier, the hybrid models generally tend to outperform the single models on inference accuracy and are known to make more accurate predictions.

3.6 Using XGBoost for Stock Price Prediction

Unlike other techniques mentioned in this section, Chou *et al.* [47] deployed a particle swarm-based optimization method to forecast stock prices. Kumar *et al.* [48] use XGBoost integrated with SARIMA to present a hybrid technique for stock prediction. Our proposed solution has taken inspiration from these studies, and we have developed an AI-based solution using tree-based XGBoost and LSTM. Other tree-based models present in literature for stock market prediction problems include Ampomah *et al.* [49], who compared several techniques like the random forest, AdaBoost, voting classified and XGBoost based on performance scores on accuracy and errors.

4 Methodology

4.1 Dataset Generation

Data was primarily gathered from Yahoo Finance and Reddit. For Yahoo Finance, records for 500 companies were collected for eight years. The subreddit World News was chosen for information extraction as the subreddit allowed for ranking news based on user scores. These two data sources were then merged by utilizing tickers, a unique identifier for each company in the stock market. The top 25 daily news (ranked from top to bottom based on how hot they are) for the weekdays in the eight years were collected to generate news data for 1989 days.

4.2 Overall System Workflow

Figure 3 illustrates the overall activity of the system from the back-end perspective, where steps 1-4 depict

the process followed for data extraction, steps 5-9 indicate the procedure followed for machine learning modules, and steps 10 and 11 show the process followed for calculating the risk analysis values. Each step is shown in detail with parameter values in the flowchart.

Figure 4 illustrates the process of collection of raw data and the functioning of the Machine Learning module.

4.2.1 Data extraction and preprocessing

Raw data is collected from its sources, preprocessed, and then merged to form a unified source. For preprocessing, we checked for null values in the combined data and replaced them with empty strings. There are several non-word tags in the headlines that would bias the sentiment analysis so we replaced them with empty string as well. After performing preprocessing steps, we have a dataset containing stock information regarding stock prices (represented by 7 features) and top 25 daily news for 1989 days.

Sentiment analysis was performed on the dataset to ascertain the company's general perception. This consisted of two parts, namely polarity lexicon analysis and subjectivity analysis. Python's library nltk or Natural Language Tool Kit was utilized for the former, where lexicons were extracted and compared with previously stored lexicons, clustered in groups to classify the dataset into three categories, namely, positive, negative and neutral. The sentiment analysis was performed using vader and gave a compound score between -0.5 (most negative) and 0.5 (most positive). The sub-jectivity of each headline was calculated using the output from TextBlob package. For example, the polarity score of the sentence "he is not a good boy" is -0.3412 – a value close to the bottom value of -0.5 indicating a highly negative sentence. Against every news item out of the top 25 daily news, a subjectivity score, compound polarity score, and positive, negative, and neutral polarity scores were obtained from TextBlob library. Mean was then calculated for each category.

Finally, a dataset consisting of stock data, subjectivity, compound polarity score, positivity, negativity, neutrality and labels is generated, which would be used as input to predict labels. Then, we deployed a two-pronged approach; the stock data is passed to LSTM to infer the predicted stock prices and the risk value, while the merged data from stock and Reddit are passed to XGBoost to infer the risk value. The dataset was then divided into three parts, the train set, the validation set, and the test set, with 60% of the data used for training, 20% for validation, and the remaining 20% used to test the system performance.

4.2.2 Classification using the XGBoost library

First, the merged data is used to infer the XGBoost parameters (n_extraction and max_depth) by passing a complete data set (without any feature reduction) for training and validating on XGBoost with an array of values for both pa-rameters. After the best set of parameters has been determined based on higher accuracy, the complete merged dataset, the dataset after feature extraction using PCA and the dataset after feature extraction using exploratory data analysis (EDA), namely correlation analysis, are then passed through XGBoost to infer the risk values. The merged dataset and the reduced dataset after EDA were both scaled using the standard scaler, while the reduced dataset after PCA was passed without scaling. All three datasets were then divided into test and training datasets using the 80-20 split rule. The best of the three datasets on evaluation on the test dataset was selected as the possible solution for getting XGBoost output. The output from XGBoost is a binary value - a number indicating if the stock will go up (0) or down (-1) or remain unchanged (0) in the given time. The number indicates the confidence (or risk level) in prediction.

1. **Classification using the LSTM model**: At the same time, the raw stock price data is scaled using MinMax scaling and then divided into test and training datasets using the 80-20 split rule. The LSTM model is then trained on this data, and the stock price is forecast. The predicted value is both used as an output and transformed to a binary value, one(1) if the stock price is to go up or remain unchanged or zero (0) if the stock price is to go down as compared to the current price.
2. **Determining the confidence in predicted stock price**: This transformed binary value is then merged with the output from XGBoost output to choose the confidence in the stock price. In our framework, risk levels 1 through 4 are employed to quantify and communicate the degree of uncertainty associated with stock price predictions. These risk levels serve as a metric for investors to gauge the confidence or reliability of the forecasted stock price movements, al-lowing for more informed decision-making. If both classifiers identify an increase in stock price, it is called risk level 1. However, if XGBoost predicts a decrease in stock price but the LSTM suggests otherwise, then a risk level 2 is assigned. Otherwise, if LSTM predicts a stock price reduction and the XGBoost suggests otherwise, then a risk level 3 is set. Finally, if XGBoost and LSTM predict a decrease in the share's price after the recommended time, a level 4 risk value is assigned to this share. Level 1 share has the least risk involved, while the level 4 share is the riskiest as all indicators are against investing in it.

4.3 Implementation

The first two modules, the Scraping module and Machine Learning module, were developed in python,

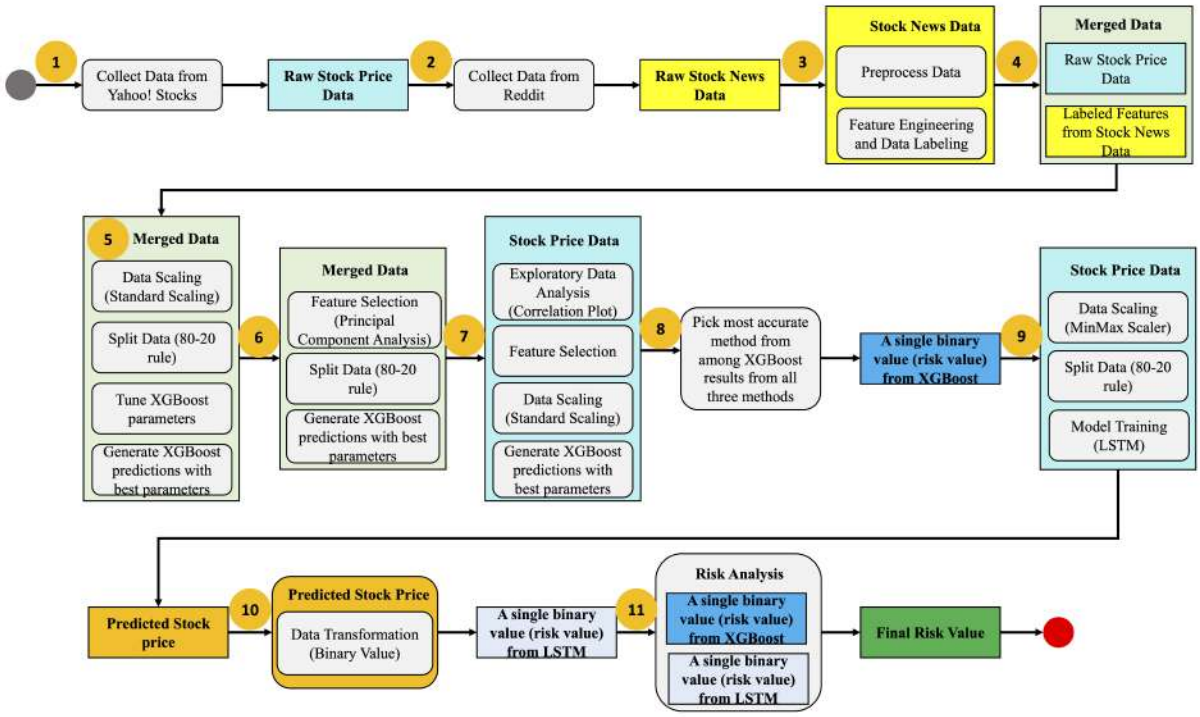

Figure 3: Overall activities of the system, in particular, data extraction and machine learning modules – The figure shows the overall activities available in the system. The system consists of 11 main steps from data extraction from Yahoo! Stocks and news from Reddit to the predicted stock price from LSTM and the final risk analysis value after combining results from XGBoost and trans-formed binary value from LSTM.

which is the popular platform for deep learning-based development. The data collection was performed using the Pandas library in Python and Reddit's API. Additionally, sentiment analysis was performed using a nltk library, which augmented the data by acquiring sentiments expressed in the news. This was further enhanced by using a bootstrap method to analyze the text for subjectivity and then merged with stock information to predict the rise and fall of stock price. The XGBoost machine learning algorithm was used for the said purpose. Furthermore, the combined dataset was fed to an LSTM, a deep learning algorithm, for stock price prediction.

4.4 Evaluation Metrics

In our research, the evaluation of the proposed model's performance employs a comprehensive set of metrics: accuracy, precision, recall, and F1 score. These metrics were selected due to their relevance and importance in assessing the quality of classification models, especially in the context of stock price prediction. Accuracy is used to measure the overall correctness of the model across all predictions, providing a general sense of model performance. Precision and recall are critical in understanding the model's ability to correctly identify positive instances among the predicted and actual positives, which is crucial for investment decisions where the cost of false positives and negatives can be high. The F1 score, a harmonic mean of precision and recall, offers a balance between the two, providing a single metric to assess the model's performance when false positives and false negatives are equally costly. Together, these metrics provide a holistic view of the model's predictive capabilities, ensuring its reliability and effectiveness in stock market forecasting.

5 Results and Discussion

5.1 Exploratory Data Analysis

After preprocessing the data, a detailed exploratory data analysis was performed. A general data summary included variance, mean, mode and median. To ensure that the difference between mean and median was not sig-nificant, an additional check was performed. As shown in Figure 5, bar plots were generated to ensure equal dataset distribution.

Co-relation was checked against each combination of fields, as shown in Figure 6, to remove any fields that were highly co-related amongst themselves. We found several fields that are mutually correlated and removed them. For example, Low, High, Adj Close, and Open features were closely correlated, and only Low was kept in the analysis. Similarly, Positive, Negative, and Neutral were highly negatively correlated, and only Neutral was kept in the features. The final five features after removing the highly correlated columns

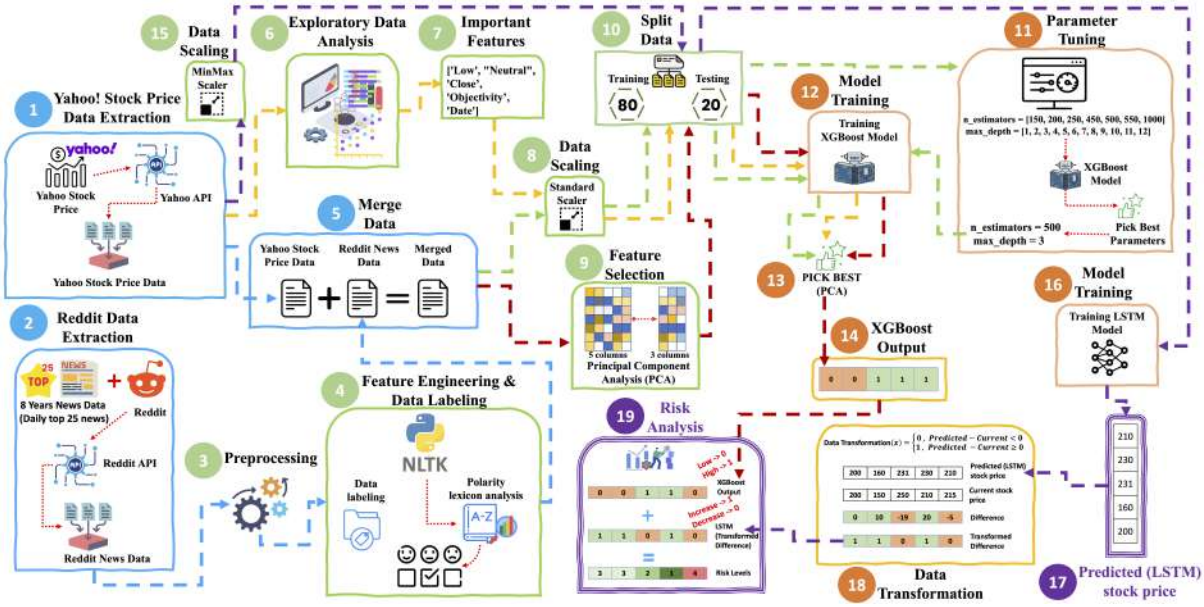

Figure 4: Illustration of Data Collection, Extraction and Model Training – The figure displays the stepwise process followed for data extraction and model training modules with their parameter settings and sample outputs. The steps for the data extraction mod-ule are shown in blue col-our, the fundamental processing steps are shown in green colour, the process followed for artificial intel-ligence modules is shown in orange colour, and the outputs are shown in purple colour. The parameters and sub-steps for each step are shown within a box in the figure.

are" Low"," Neutral"," Close"," Objec-tivity", and" Date". These five columns can now generate results from the XGBoost algorithm. The exploratory data analysis helped us identify important features and aid in removing any inherent bias in the columns of the datasets.

5.2 Performance of Machine Learning module

The system accuracy was calculated using the Confusion Matrix. Principle component analysis was performed to select the best variables, and the system parameters were tweaked with the estimator of 3. This yielded an accuracy of 0.941 (94.1%). The precision, recall, and f1 scores were 0.96, 0.91 and 0.94, respectively, for the label '0' and 0.92, 0.97 and 0.95, respectively, for the label '1', as illustrated in Table 1.

Table 1: System Performance

Label	Type	Precision	Recall	F1 Score
0	Loss	96%	91%	94%
1	Rise	92%	97%	95%

Initially, the XGBoost algorithm yielded an accuracy of 0.589 (58.9%) with the default parameter settings with the merged dataset. However, further parameter tuning was done to get precision, recall and f1-score, and several values (150, 200, 250, 450, 500, 550, and 1000) covering the whole parameter space for the number of estimators and maximum depth (1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, and 12) parameters were considered and evaluated for accuracy before settling on the number of estimators as 500 and maximum depth at 3. The highest accuracy for the whole dataset with XGBoost at 500 number of estimators at maximum depth 3 was observed to be 0.601(60.1%). Note that the system's accuracy is relatively low for a binary variable classifier. We hypothesize that this is due to the highly correlational values among the columns of the dataset. Therefore, we used exploratory data analysis and PCA analysis to improve the accuracy and remove noisy inputs. The accuracy of the data was significantly enhanced with the feature engineering technique PCA to 94.1%. Table II displays the precision, recall, and f1-scoresfor each label, respectively. Thus, data treated with feature engineering through PCA with the best parameter settings were chosen as the most suitable input for XGBoost to determine whether the stock price will go up, go down or remain unchanged three days from now. Based on the results for labeling and the comprehensive nature of the evaluation metric, we use F1 score as it simultaneously describes the precision and recall of our system. A sample run of XG_Boost component with empirical data is shown in Figure 7.

Once the labels were accurately predicted, stock price prediction was performed using LSTM. With several rounds of parameter tuning, the architecture and parameter settings for LSTM were finalized. The LSTM architecture consisted of three layers - an 'LSTM' cell layer with 1000 units and 'tanh' activation function, a 'dropout' layer with 20% con-nectivity, and a 'dense' layer with 'linear' activation function with

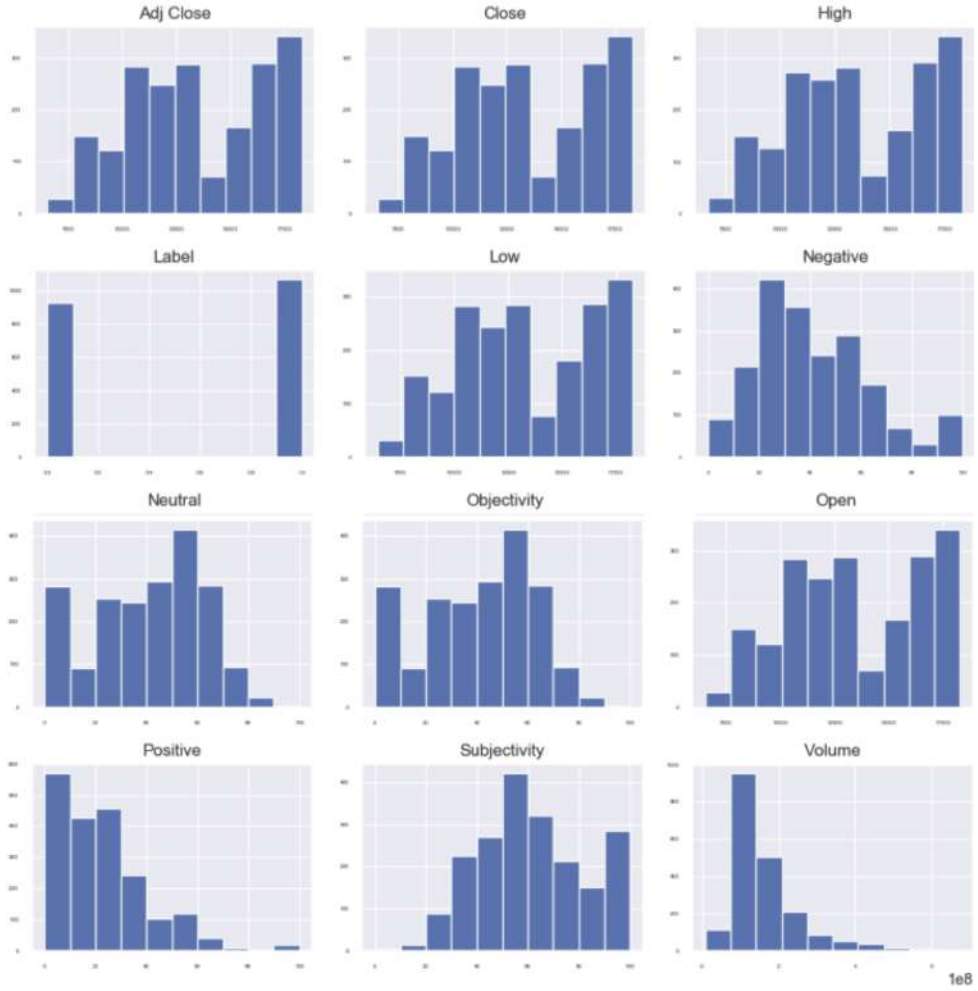

Figure 5: Exploratory Data Analysis for values in the datasets – The figure shows the results of exploratory data analysis with standard statistical measures like mean, mode and median, etc., for all variables. The results are depicted as bar plots.

1 unit. The 'mean squared error function' was used to calculate loss, and 'adam' optimizer optimized the learning. With fine-tuned parameters of '50 iterations' and '16 as the batch size', the yielded accuracy was 76.2%. Four Risk levels were established, with level 1 being the lowest and level 4 being the highest. This was done using a combination of both LSTM and XG-Boost predictions. Labels were calculated from LSTM predictions, where 1 represented an in-crease or equal prediction of the stock price compared to the previous day, while 0 meant a decrease in stock price prediction. A sample run of LSTMs module is shown in Figure 8.

When both XGBoost and LSTM predicted an increase, the risk level was 1. When the boosting algorithm predicted a rise and LSTM predicted a fall, the risk level was 2 and vice versa for risk level 3. When both models predicted a fall, the risk level rose to 4. This is illustrated in Table 2. Max profit was calculated for the next 3 days using LSTM predictions. These results were then uploaded to the database. Although social media platforms such as Reddit may contain noise due to malicious bots, our results demonstrate that the data is sufficiently reliable for performing stock price analysis.

Table 2: Risk Level Assessment Based On Machine Learning Algorithm Outputs

Risk Level	XG-Boost Prediction	LSTM Prediction
1	Rise	Rise
2	Rise	Fall
3	Fall	Rise
4	Fall	Fall

However, in addition to the evaluation metrics deployed in this study, the importance of translating the model's predictive accuracy into tangible economic value for investors has not been studied in this study. Consequently, we propose an additional experiment to further validate the practical utility of our system. This experiment will simulate the evolution of a diversified stock portfolio, which is periodically rebalanced according to the recommendations generated by our predictive system. The performance of this

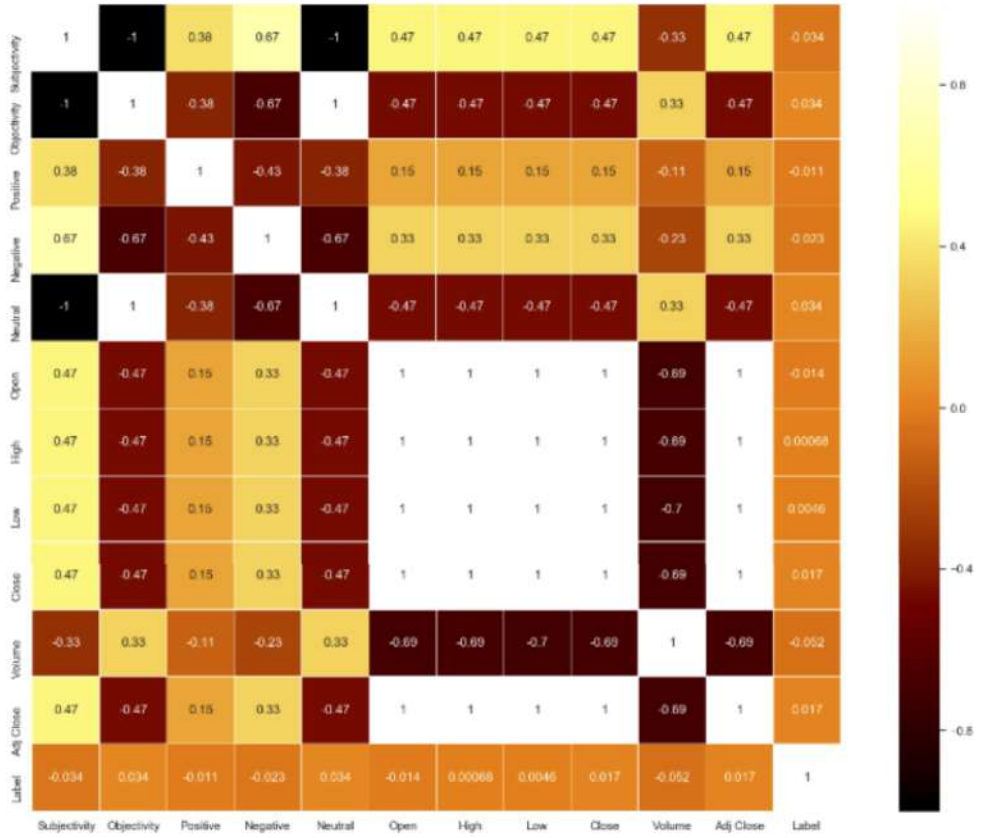

Figure 6: Correlation between all features for the merged dataset (including labels) – The figure displays the heat map showing the correlation between all features used for the combined dataset. The columns with high correlation values can be removed for downstream analysis.

simulated portfolio will then be benchmarked against two alternative strategies: a buy-and-hold approach, representing a passive investment strategy, and a random trading strategy, which will serve as a baseline for assessing the added value of our system's insights. The portfolio's performance will be evaluated based on standard financial metrics such as overall return, Sharpe ratio, and maximum drawdown, providing a comprehensive view of the risk-adjusted returns generated by following the system's recommendations. This comparative analysis will enable us to quantify the economic benefits of our predictive model beyond conventional accuracy metrics. However, it's crucial to note that this proposed simulation does not account for transaction costs associated with buying and selling securities. Transaction costs can significantly impact the net return of actively managed portfolios, especially those that entail frequent rebalancing. As such, while the results of this simulation will offer valuable insights into the potential economic utility of our system, they should be interpreted with caution. Future iterations of this work will aim to incorporate transaction cost estimates and economic evaluation inline with the earlier strategies to provide a more realistic assessment of the system's economic viability.

6 Conclusion

In this study, we introduce a novel financial advisory system that significantly outperforms existing models by achieving a stock market prediction accuracy of 94% and a stock price forecast accuracy of 76.2%. This advancement is the result of meticulous exploratory analysis and optimization of the LSTM and XGBoost models, aimed at enhancing investment decision-making processes. The integration of social media sentiment analysis with traditional financial data analysis provides a comprehensive market insight, a feature notably absent in current models. Additionally, our system innovatively incorporates risk analysis with stock price forecasting, addressing a critical void in market prediction tools. Future enhancements may include leveraging additional social media platforms and financial datasets for even more refined analyses. Furthermore, potential expansions into real estate forecasting highlight the system's versatility. Ultimately, this work not only showcases a sophisticated approach to merging disparate data for stock prediction but also emphasizes the importance of risk assessment in financial analyses, marking a significant leap forward in financial advisory services.

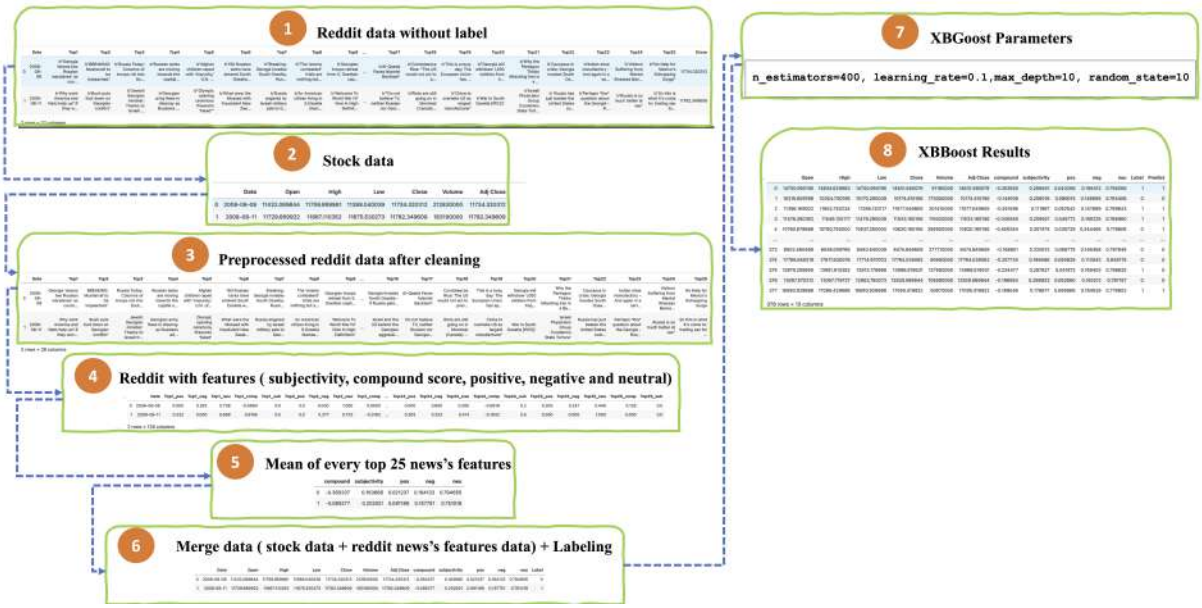

Figure 7: Demonstration of the work performed by XG_Boost module on empirical dataset – The figure displays the step-by-step approach and transformation on the stock data as well as on the reddit data to infer risk values and other results from XG_Boost module.

Competing interests

The authors have declared that no competing interests exist.

Authors' contribution

The authors confirm contribution to the paper as follows: Conceptualization, Z.A.; methodology, A.Z.I. and Z.A.; software, Z.A.; validation, Z.A. and T.A.K.; data curation, T.A.K.; writ-ing—review and editing, R.H.A., and T.A.K.; visualization, Z.A.; supervision, R.H.A.; project administration, T.A.K. All authors reviewed the results and approved the final version of the manuscript.

References

[1] T. Hong, P. Pinson, Y. Wang, R. Weron, D. Yang, and H. Zareipour, "Energy forecasting: A review and outlook," *IEEE Open Access Journal of Power and Energy*, vol. 7, pp. 376–388, 2020.

[2] S. Boccaletti, W. Ditto, G. Mindlin, and A. Atangana, "Modeling and forecasting of epidemic spreading: The case of covid-19 and beyond," *Chaos, solitons, and fractals*, vol. 135, p. 109794, 2020.

[3] M. Shehzadi, R. H. Ali, Z. u. Abideen, A. Z. Ijaz, and T. A. Khan, "Enhancing flood resilience: Streamflow forecasting and inundation modeling in pakistan," *Engineering Proceedings*, vol. 56, no. 1, p. 315, 2023.

[4] S. Casas, C. Gulino, S. Suo, K. Luo, R. Liao, and R. Urtasun, "Implicit latent variable model for scene-consistent motion forecasting," in *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XXIII 16*. Springer, 2020, pp. 624–641.

[5] P. Goodwin, "Integrating management judgment and statistical methods to improve short-term forecasts," *Omega*, vol. 30, no. 2, pp. 127–135, 2002.

[6] A. Thakkar and K. Chaudhari, "Fusion in stock market prediction: a decade survey on the necessity, recent developments, and potential future directions," *Information Fusion*, vol. 65, pp. 95–107, 2021.

[7] O. Bustos and A. Pomares-Quimbaya, "Stock market movement forecast: A systematic review," *Expert Systems with Applications*, vol. 156, p. 113464, 2020.

[8] J. Wang, J. He, C. Feng, L. Feng, and Y. Li, "Stock index prediction and uncertainty analysis using multi-scale nonlinear ensemble paradigm of optimal feature extraction, two-stage deep learning and gaussian process regression," *Applied Soft Computing*, vol. 113, p. 107898, 2021.

[9] M. H. Shah, M. A. Bakar, R. H. Ali, Z. U. Abideen, U. Arshad, A. Z. Ijaz, N. Ali, M. Imad, and S. Nabi, "Investigating novel machine learning based intrusion detection models for nsl-kdd data sets," in *2023 International Conference on IT and Industrial Technologies (ICIT)*. IEEE, 2023, pp. 1–6.

[10] A. Haider, A. B. Siddique, R. H. Ali, M. Imad, A. Z. Ijaz, U. Arshad, N. Ali, M. Saleem, and N. Shahzadi, "Detecting cyberbullying using machine learning approaches," in *2023 International Conference on IT and Industrial Technologies (ICIT)*. IEEE, 2023, pp. 1–6.

[11] A. B. Siddique, M. A. Bakar, R. H. Ali, U. Arshad, N. Ali, Z. U. Abideen, T. A. Khan, A. Z. Ijaz, and M. Imad, "Studying the effects of feature selection approaches on machine learning techniques for mushroom classification problem," in *2023 International Conference on IT and Industrial Technologies (ICIT)*. IEEE, 2023, pp. 1–6.

[12] A. Shabbir, R. H. Ali, M. Z. Shabbir, Z. U. Abideen, A. Z. Ijaz, N. Ali, M. H. Shah, S. Nabi, and K. Perveen, "Stock price forecasting using hidden markov models," in *2023 International Conference on IT and Industrial Technologies (ICIT)*. IEEE, 2023, pp. 1–6.

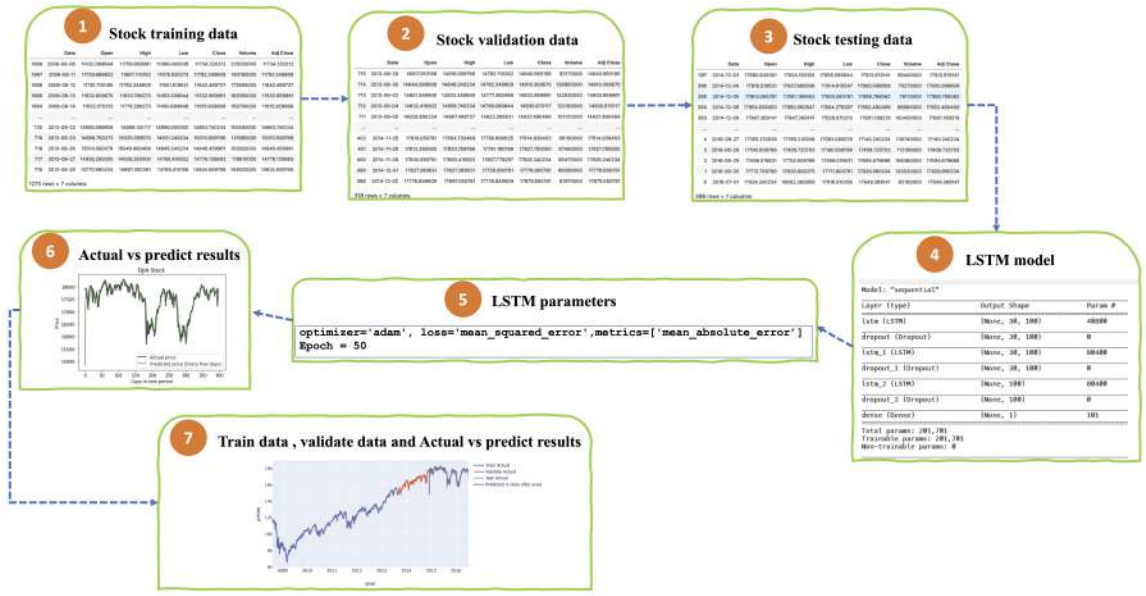

Figure 8: Demonstration of the work performed by LSTM module on the empirical data – The figure displays the step-by-step approach and transformation on the stock data to first learn and then infer stock price values and other results from LSTM module.

[13] M. Javed Awan, M. S. Mohd Rahim, H. Nobanee, A. Munawar, A. Yasin, and A. M. Zain, "Social media and stock market prediction: a big data approach," *Computers, Materials & Continua*, vol. 67, no. 2, pp. 2569–2583, 2021.

[14] A. Namdari and Z. S. Li, "Integrating fundamental and technical analysis of stock market through multi-layer perceptron," in *2018 IEEE technology and engineering management conference (TEMSCON)*. IEEE, 2018, pp. 1–6.

[15] R. H. Ali, "From genomes to post-processing of bayesian inference of phylogeny," Ph.D. dissertation, KTH Royal Institute of Technology, 2016.

[16] R. H. Ali, M. Bark, J. Miró, S. A. Muhammad, J. Sjöstrand, S. M. Zubair, R. M. Abbas, and L. Arvestad, "Vmcmc: a graphical and statistical analysis tool for markov chain monte carlo traces," *BMC bioinformatics*, vol. 18, pp. 1–8, 2017.

[17] R. Chandra and Y. He, "Bayesian neural networks for stock price forecasting before and during covid-19 pandemic," *Plos one*, vol. 16, no. 7, p. e0253217, 2021.

[18] R. Kumar, S. Srivastava, A. Dass, and S. Srivastava, "A novel approach to predict stock market price using radial basis function network," *International Journal of Information Technology*, vol. 13, no. 6, pp. 2277–2285, 2021.

[19] S. Garcia-Vega, X.-J. Zeng, and J. Keane, "Stock returns prediction using kernel adaptive filtering within a stock market interdependence approach," *Expert Systems with Applications*, vol. 160, p. 113668, 2020.

[20] I. Kumar, K. Dogra, C. Utreja, and P. Yadav, "A comparative study of supervised machine learning algorithms for stock market trend prediction," in *2018 Second International Conference on Inventive Communication and Computational Technologies (ICICCT)*. IEEE, 2018, pp. 1003–1007.

[21] O. K. Majeed, Z. ul Abideen, U. Arshad, R. H. Ali, A. Habib, and R. Mustafa, "Adaptivecloset: Reinforcement learning in personalized clothing recommendations," in *2023 18th International Conference on Emerging Technologies (ICET)*. IEEE, 2023, pp. 305–309.

[22] A. Mashhood, Z. ul Abideen, U. Arshad, R. H. Ali, A. A. Khan, and B. Khan, "Innovative poverty estimation through machine learning approaches," in *2023 18th International Conference on Emerging Technologies (ICET)*. IEEE, 2023, pp. 154–158.

[23] T. Ahmed, A. Maaz, D. Mahmood, Z. ul Abideen, U. Arshad, and R. H. Ali, "The yolov8 edge: Harnessing custom datasets for superior real-time detection," in *2023 18th International Conference on Emerging Technologies (ICET)*. IEEE, 2023, pp. 38–43.

[24] M. H. Ishaq, R. Mustafa, U. Arshad, Z. ul Abideen, R. H. Ali, and A. Habib, "Deciphering faces: Enhancing emotion detection with machine learning techniques," in *2023 18th International Conference on Emerging Technologies (ICET)*. IEEE, 2023, pp. 310–314.

[25] I. Mueed, U. Arshad, and R. H. Ali, "Revolutionizing campus exploration with gikilens: A deep learning-powered object detection app," in *2023 18th International Conference on Emerging Technologies (ICET)*. IEEE, 2023, pp. 315–320.

[26] M. Shehzadi, U. Arshad, Z. Abideen, R. H. Ali, A. A. Khan, and A. Z. Ijaz, "Identifying covid-19 through x-ray and ct scan images using machine learning," in *2023 18th International Conference on Emerging Technologies (ICET)*. IEEE, 2023, pp. 56–61.

[27] Y. Yang, S. Tu, R. H. Ali, H. Alasmary, M. Waqas, and M. N. Amjad, "Intrusion detection based on bidirectional long short-term memory with attention mech-

anism," *Computers, Materials & Continua*, vol. 74, no. 1, pp. 801–815, 2023.

[28] J. Verbraeken, M. Wolting, J. Katzy, J. Kloppenburg, T. Verbelen, and J. S. Rellermeyer, "A survey on distributed machine learning," *ACM computing surveys (csur)*, vol. 53, no. 2, pp. 1–33, 2020.

[29] X. Pang, Y. Zhou, P. Wang, W. Lin, and V. Chang, "An innovative neural network approach for stock market prediction," *The Journal of Supercomputing*, vol. 76, pp. 2098–2118, 2020.

[30] H. Rezaei, H. Faaljou, and G. Mansourfar, "Stock price prediction using deep learning and frequency decomposition," *Expert Systems with Applications*, vol. 169, p. 114332, 2021.

[31] I. K. Nti, A. F. Adekoya, and B. A. Weyori, "A comprehensive evaluation of ensemble learning for stock-market prediction," *Journal of Big Data*, vol. 7, no. 1, p. 20, 2020.

[32] S. Selvin, R. Vinayakumar, E. Gopalakrishnan, V. K. Menon, and K. Soman, "Stock price prediction using lstm, rnn and cnn-sliding window model," in *2017 international conference on advances in computing, communications and informatics (icacci)*. IEEE, 2017, pp. 1643–1647.

[33] P. Chhajer, M. Shah, and A. Kshirsagar, "The applications of artificial neural networks, support vector machines, and long–short term memory for stock market prediction," *Decision Analytics Journal*, vol. 2, p. 100015, 2022.

[34] D. Shah, H. Isah, and F. Zulkernine, "Stock market analysis: A review and taxonomy of prediction techniques," *International Journal of Financial Studies*, vol. 7, no. 2, p. 26, 2019.

[35] P. Jay, V. Kalariya, P. Parmar, S. Tanwar, N. Kumar, and M. Alazab, "Stochastic neural networks for cryptocurrency price prediction," *IEEE Access*, vol. 8, pp. 82 804–82 818, 2020.

[36] R. Parekh, N. P. Patel, N. Thakkar, R. Gupta, S. Tanwar, G. Sharma, I. E. Davidson, and R. Sharma, "Dl-guess: Deep learning and sentiment analysis-based cryptocurrency price prediction," *IEEE Access*, vol. 10, pp. 35 398–35 409, 2022.

[37] R. P. Schumaker and H. Chen, "Textual analysis of stock market prediction using breaking financial news: The azfin text system," *ACM Transactions on Information Systems*, vol. 27, no. 2, p. 1–19, Feb. 2009. [Online]. Available: http://dx.doi.org/10.1145/1462198.1462204

[38] G. Ding and L. Qin, "Study on the prediction of stock price based on the associated network model of lstm," *International Journal of Machine Learning and Cybernetics*, vol. 11, no. 6, p. 1307–1317, Nov. 2019. [Online]. Available: http://dx.doi.org/10.1007/s13042-019-01041-1

[39] H. Huang, X. Liu, Y. Zhang, and C. Feng, "News-driven stock prediction via noisy equity state representation," *Neurocomputing*, vol. 470, p. 66–75, Jan. 2022. [Online]. Available: http://dx.doi.org/10.1016/j.neucom.2021.10.092

[40] R. Akita, A. Yoshihara, T. Matsubara, and K. Uehara, "Deep learning for stock prediction using numerical and textual information," in *2016 IEEE/ACIS 15th International Conference on Computer and Information Science (ICIS)*. IEEE, Jun. 2016. [Online]. Available: http://dx.doi.org/10.1109/icis.2016.7550882

[41] K. Mishev, A. Gjorgjevikj, I. Vodenska, L. T. Chitkushev, and D. Trajanov, "Evaluation of sentiment analysis in finance: From lexicons to transformers," *IEEE Access*, vol. 8, p. 131662–131682, 2020. [Online]. Available: http://dx.doi.org/10.1109/access.2020.3009626

[42] H. Xu, L. Chai, Z. Luo, and S. Li, "Stock movement prediction via gated recurrent unit network based on reinforcement learning with incorporated attention mechanisms," *Neurocomputing*, vol. 467, p. 214–228, Jan. 2022. [Online]. Available: http://dx.doi.org/10.1016/j.neucom.2021.09.072

[43] K. A. Althelaya, E.-S. M. El-Alfy, and S. Mohammed, "Evaluation of bidirectional lstm for short-and long-term stock market prediction," in *2018 9th International Conference on Information and Communication Systems (ICICS)*. IEEE, Apr. 2018. [Online]. Available: http://dx.doi.org/10.1109/iacs.2018.8355458

[44] Q. Chen, W. Zhang, and Y. Lou, "Forecasting stock prices using a hybrid deep learning model integrating attention mechanism, multi-layer perceptron, and bidirectional long-short term memory neural network," *IEEE Access*, vol. 8, p. 117365–117376, 2020. [Online]. Available: http://dx.doi.org/10.1109/access.2020.3004284

[45] X. Chen, X. Ma, H. Wang, X. Li, and C. Zhang, "A hierarchical attention network for stock prediction based on attentive multi-view news learning," *Neurocomputing*, vol. 504, p. 1–15, Sep. 2022. [Online]. Available: http://dx.doi.org/10.1016/j.neucom.2022.06.106

[46] D. Wu, X. Wang, and S. Wu, "A hybrid framework based on extreme learning machine, discrete wavelet transform, and autoencoder with feature penalty for stock prediction," *Expert Systems with Applications*, vol. 207, p. 118006, Nov. 2022. [Online]. Available: http://dx.doi.org/10.1016/j.eswa.2022.118006

[47] J.-S. Chou, D.-N. Truong, and T.-L. Le, "Interval forecasting of financial time series by accelerated particle swarm-optimized multi-output machine learning system," *IEEE Access*, vol. 8, p. 14798–14808, 2020. [Online]. Available: http://dx.doi.org/10.1109/access.2020.2965598

[48] D. S. Kumar, B. Thiruvarangan, A. Vishnu, A. S. Devi, D. Kavitha *et al.*, "Analysis and prediction of stock price using hybridization of sarima and xgboost," in *2022 International Conference on Communication, Computing and Internet of Things (IC3IoT)*. IEEE, 2022, pp. 1–4.

[49] E. K. Ampomah, Z. Qin, and G. Nyame, "Evaluation of tree-based ensemble machine learning models in predicting stock price direction of movement," *Information*, vol. 11, no. 6, p. 332, Jun. 2020. [Online]. Available: http://dx.doi.org/10.3390/info11060332

Citation: Zain-ul-Abideen, Raja Hashim Ali, Ali Zeeshan Ijaz and Talha Ali Khan. *A Novel NLP-based Stock Market Price Prediction and Risk Analysis Framework*. Journal of Computer Science & Technology, vol. 24, no. 2, pp. 74–87, 2024.
DOI: 10.24215/16666038.24e07
Received: September 22, 2023 **Accepted:** April 28, 2024.
Copyright: This article is distributed under the terms of the Creative Commons License CC-BY-NC-SA.

© 2024. This work is licensed under https://creativecommons.org/licenses/by-nc-sa/4.0/ (the “License”). Notwithstanding the ProQuest Terms and Conditions, you may use this content in accordance with the terms of the License.