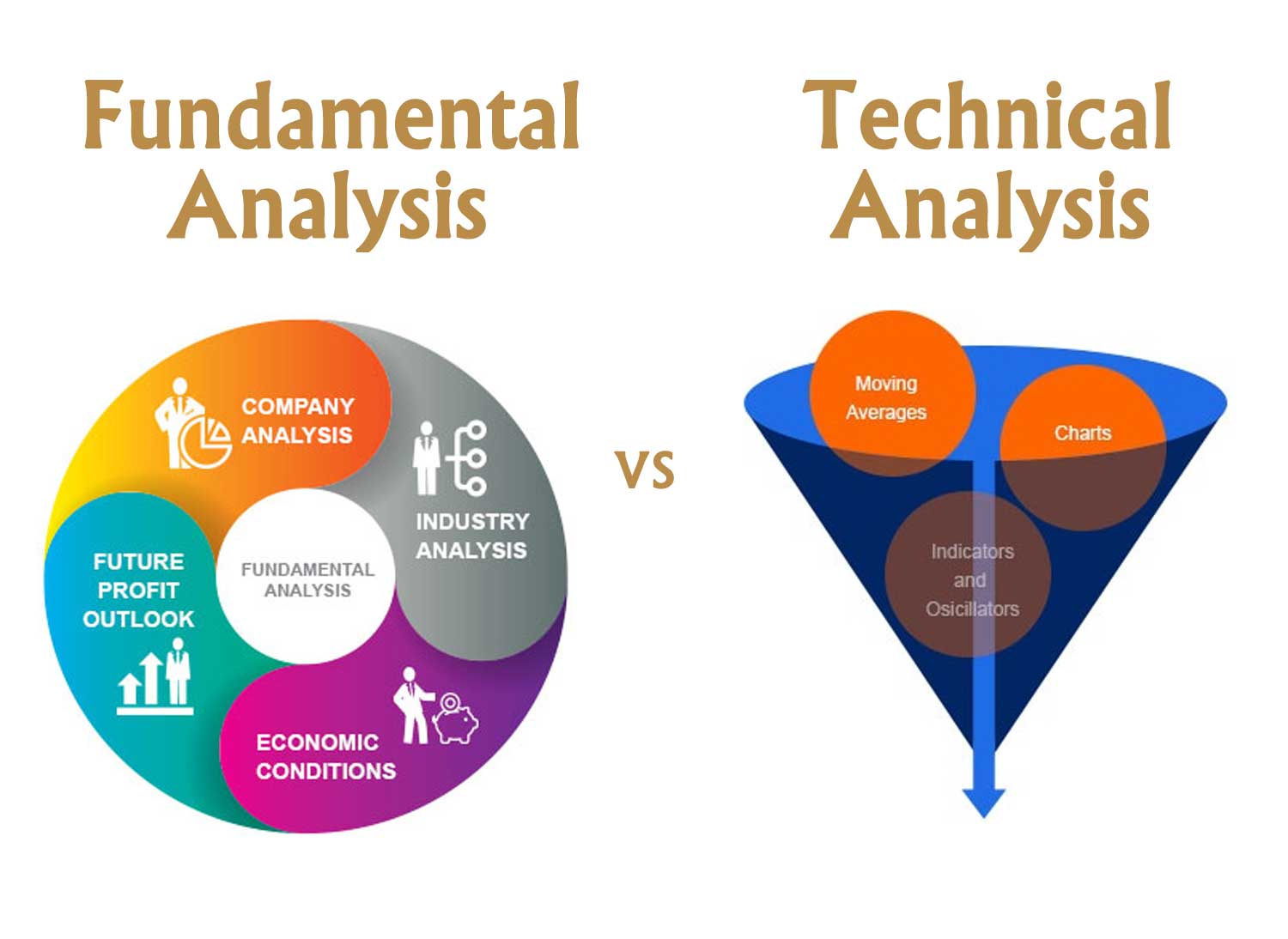
**Abstract:**

Making predictions about the stock market is challenging, since there are many factors that influence how the market behaves. For my project I’m predicting the Istanbul Stock Exchange (ISE) using the data from the UCI machine learning repository. There are several features in the dataset such as a, b, c.., and the primary attribute is the ISE feature, which is used to predict stock values using the machine learning models. The goal of this project is to compare how well two model forecast the ISE index: Long Short-Term Memory (LSTM) and Autoregressive Integrated Moving Average (ARIMA). Finding the model with the highest accuracy while utilizing the least amount of data input is one of the goals. By creating the model architecture, assess the model performance and figure out the best model for short-term stock prediction by studying the relevant literature on predicting stock using LSTM and ARIMA. The understanding of the stock exchange trends and the effectiveness of models in financial forecasting will be enhanced by the result of this study.

**Introduction:**

**Background**

Forecasting the stock market’s future using the past data and other influencing factors is referred as stock market prediction. The two main categories of analyzing stock market are fundamental analysis and technical analysis (figure1). By evaluating the company’s present financial performance and business environment, fundamental analysis projects the organization’s potential profitability. On the other hand, technical analysis involves analyzing charts i.e. is stock’s historical prices and utilizing statistical data to find market trends. In this project we will concentrate on the technical analysis to forecast the stock market.



**Figure 1: Fundamental Analysis and Technical Analysis comparison**

The ability to make better decisions and control risk is one of the main advantages that offers the investors, financial analysts, and traders to reduce risk and make smarter judgements. Because of several factors that may impact market behavior in addition to the regular fluctuations in prices, predicting stock market values is a difficult and complex undertaking. The applications of machine learning and statistical models in financial forecasting has been developing with an aim of improving the accuracy and confidence of investing predictions.

One important financial market that represents Turkeys economy performance and investment opportunities is the Istanbul Stock Exchange (ISE). Both domestic and foreign investors can benefit from understanding and forecasting the ISE attribute. A dataset from the UCI machine learning repository, which has multiple financial variables, is used in this work.

The above features are essential in developing predictive models and fully understanding the connections between the targeted variable and other financial metrics.

**Problem Statement**

This study’s main goal is to use the historical financial data to anticipate the Istanbul Stock Exchange (ISE) with accuracy. Choosing the right model for prediction is important because of the complexity of financial markets. Even though they are widely used, traditional time series models like ARIMA might not be able to properly represent complex dependencies on time and non-linear relationships. On the other hand, advanced deep learning models such as LSTM network have shown progress in processing sequential data; however, they require sufficient training data and appropriate tuning. The objective of this research is to evaluate and compare the ISE indices forecasting performance of the LSTM and ARIMA models to identify which model requires less data to provide forecast that are more accurate.

**Justification of the study**

This study is justified by the potential benefits of improving investment forecasting accuracy. Accurate models may enhance financial decision-making, investment strategies and risk minimization. By evaluating the performance of LSTM and ARIMA models, identifying the benefits and drawbacks of each model, this project contributes to the ongoing research in financial projections. The findings will be useful to the investors, financial analysts and researchers who are interested in applying machine learning and statistical models in the financial market. And in cases where data is very limited, understanding which model performs best with less data might be extremely beneficial.

**Research Questions**

The research aims to provide explanation to the following questions:

1. When predicting the stock market with limited amounts of data input, which model LSTM or ARIMA provides the highest accuracy?
2. By studying market movements, daily returns, moving averages and correlations between stocks, how can we forecast future stock behavior?

**Aims and Objectives**

The aim of this research is to find out whether model like LSTM or ARIMA predicts the Istanbul Stock Exchange (ISE) index more accurate. The objective is to:

1. Explain how well the ARIMA and LSTM model predicts the ISE index.
2. Determine which model works best with minimum data input.
3. Describes some efficient financial forecasting techniques.

**Literature Review**

“Stock Predicting based on LSTM and ARIMA”

Qian, H. (2022), in their case study of “Stock Predicting based on LSTM and ARIMA” using LSTM and ARIMA model 95% of the total data for each stock is applied as the training set data, which is used to train the model parameters and rest is used as the test set data. The data used is Google Stock Price from April -2017 to April-2022. The model’s predicted outcomes on the test dataset validates the model’s benefits and drawbacks. To avoid the overfitting, incorporate several dropout layers and six LSTM layers into the LSTM model architecture. The output’s dimensions are 64 units, and the return sequences parameter represent stock. Using LSTM and ARIMA 487 to predict whether to return the entire series or only the last output in the output sequence, with input\_shape serving as the shape of the training set. The dense layer with a specified output of one unit will be added, and the drain layer with an output value of 0.2 will be eliminated. Lastly, the model works with a batch size of 32 and can be used for 100 iterations. But the study doesn’t train RNN on the entire observation sequence, instead it uses a batch of small subsequences randomly selected from the training data. According to the prediction findings, the LSTM algorithm outperforms the alternative ARIMA in terms of MSE, MAE and RMSE. When projecting future prices, investors in the capital market may find the findings of this study useful. 97% of the data was collected, which was highly promising for this research project. During the training of the model values of 01.3071 at epoch 74 is the best validation performance. (NO DATASET link).

“Comparison of ARIMA, ANN and LSTM for Stock Price Prediction”

Ma, Q. (2022) examined the three models in this research. Through an analysis of the three model’s underlying assumptions and forecast outcomes, this paper particularly contrasts three models. Although the LSTM model is heavily influenced by the data processing, it is considered to have the best prediction performance in the end. In terms of performance, the ANN model outperforms the ARIMA model. Also, Ma, Q. mentioned it’s possible that the ANN is more responsible for the LSTM model’s performance. Additionally, by strengthening the white noise sequence, ARIMA-GARCH can rise the ARIMA model’s accuracy even further. In comparison with the other two models, the LSTM model adds additional variables to differentiate between abrupt changes and sudden fluctuations in markets. The dataset is “DELL's stock price” from 2010 used to analyze the models in this study.

“Time series forecasting of stock market using ARIMA, LSTM and FB prophet”

Sunki, A., SatyaKumar, C, Surya Narayana, G, S., Koppera, V., & Hakeem, M. (2024) concluded that, time series forecasting in the stock market is a challenging task requiring technical methods and careful research. Though no forecasting technique can accurately predict stock prices, time series forecasting offers helpful insights and helps investors make viable choices. These models incorporate a variety of factors, including trend, seasonality, and autocorrelation, to generate projections. As a result, it can be concluded that the ARIMA model provides a better match to the data than both LSTM and FBProphet models.

Based on these three models, ARIMA has the lowest RMSE (root mean square error) value 7.8919253, suggesting that it has the highest predicted accuracy for this dataset. Followed by LSTM model has the 10.33765 RMSE value and FBProphet 9.11863 also had the lower RMSE value which appears to be the least accurate in predicting the target variable. The dataset used in this paper is “Netflix”

“ARIMA vs LSTM on NASDAQ stock exchange data”

Kobiela, D., Krefta, D., Krol, W., Weichbroth, P. (2022) in their case study of using ARIMA and LSTM, which model works better in terms of the selected input data, parameters and the feature count has been shown in this research paper. Mean Square Error (MSE) and mean absolute percentage error (MAPE) were the relative metrics used to compare the selected models. And in regression problems usually use selected measures. By comparing the selected metrics in various models, the study reveals which model performs better.

The prediction of the stock prices performed over a range of time periods, from one day to nine months. Overall, ARIMA is better at processing single-feature (price) data, and it generally performs better that LSTM, especially over a long period of time. Due to absence of more features, LSTM struggled with learning and only performed better for one-day forecasts. Future study involves adding more features and exploring hybrid models which potentially increase LSTM performance. The study concluded that, for single-feature stock price prediction on “NASDAQ data”, ARIMA is now more successful.

“A Comparative Study of Future Stock Price Prediction Through Artificial Neural Network and ARIMA Modelling”

Varshney, S., & Srivastava, P. (2024) this study compares and calculates the stock price forecasting using ARIMA and ANN. The outcomes showed that the ANN approach to stock price forecasting is more accurate that ARIMA modelling. The NIFTY 50 index closing price from the NSE India served as the source of data for this research. The Levenberg–Marquardt algorithm was used to simulate and train the network. Given the forecast error arising from these two models is rather small, it is evident that they reached nearly forecast performance. These results are comparable to those of Adebiyi et al. (2014) ANN results are predicted more accurately than ARIMA. The ANN model’s predicted line and the NSE index actual values nearly overlap. On the other hand, the ARIMA model’s prediction appears to diverge rather than exactly overlap. Towards the end of the study period, the ARIMA results show rising deviation as it shown a linear pattern that is directed based on the ARIMA model results.

Talati, D., Patel, M., & Patel B. (2022) suggest that because LSTM can manage the sequential and nonlinear structure of financial data and the model is good at predicting the stock prices. The author believes that feature selection and data preparation can improve the model’s performance. The author has trained the model on three different datasets such as “Infosys (1996 to 2022), Microsoft (1986 to 2022) and TCS (2002 to 2022)”.

Huang, W. (2023), concludes that LSTM model can learn from the sequential patterns of stock data and include long term dependencies, as they are quite predictive at predicting the prices of the stock. In the study the author compared the model’s performance with the existing models, gives a 35.18% improvement in predicting the stock market. The author uses the historical stock price data of “Google stocks “.

“Comparison of ARIMA and Artificial Neural Networks Models for Stock Price Prediction”

https://onlinelibrary.wiley.com/doi/full/10.1155/2014/614342

Adebiyi, A. A., Adewumi, A. O., & Ayo, C. K. (2014) in this study the understanding of ARIMA and ANN models for predicting “Dell stock prices” were examined. Based on the minimal forecast errors and close match to actual prices, ARIMA (1,0,0) was found to be the optimal configuration. Additionally, the ANN model with a 10-17-1 architecture demonstrated great accuracy and minimal mistakes. Although both models worked well, test results showed that the ANN model typically produces forecast that were more accurate than those made by the ARIMA model. Though the ANN model performed somewhat better overall, statistical tests revealed little variance between the actual and predicted values for either model.

Whereas the forecasts of the ANN model closely matched the actual values, the ARIMA model had a directional pattern. To increase forecast accuracy even further using new market indexes and recent stock data, future research may investigate hybrid models.

**Methodology**

**Overview**

The goal of this project is to use machine learning techniques to predict the Istanbul Stock Exchange. This will be executed in few steps (Figure 2). Collecting and reading the data is the first step, and the comes data preprocessing, to handle the missing values and identify the features from the dataset. And then data is split into training and testing and trained using LSTM and ARIMA models. Subsequently we will test the models, and in the end, we’ll assess model performance by tuning the parameters to see which model forecast the future stocks more accurately.

|  |
| --- |
| * Clean data (missing values, handle outliers). * Normalize data. * Feature Selection * Install “**ucimlrepo”** * Load fetch” unci repo(id=247) * Here Id is Dataset for Uci repo. * Time series (ARIMA) * LSTM (Long Short-Term Memory) * ANN (Artificial Neural Network) * Update Model architecture. * Update parameters to get better accuracy.   **Dataset**: Istanbul Stock Exchange  End   * SVM (Support Vector Machine) * Time series (ARIMA) * LSTM (Long Short-Term Memory) * ANN (Artificial Neural Network) * Random Forest   Predict the model.  Apply model (Test Dataset)  Predict the future stock market.  Optimize model (Fine-tune model)  Optimize model (Fine-tune model)  Define the ML model architecture.  Train the Model &  Validattion  Split data (Train & Test)  Pre-processing Data  Data Collection  Load & Read Data  Start |

* Training
* Testing
* Validation

**Figure 2: Data management plan flow chart**

**Long Short-Term Memory (LSTM):**

Long Short-Term Memory networks are a type of Recurrent Neural Network (RNN) that process sequential data and collect long-term dependencies. “Hochreite”r and “Schmidhuber” created LSTM to address the issues identified by conventional RNN and machine learning methods. When handling time-series data, like stock prices, which are generally non-stationary and show trends and changing circumstances, LSTM prove to be particularly useful.

**Why use LSTM for predicting stock market?**

With the recent breakthroughs that have been happening in data science, it is found that for almost all these sequence prediction problems, long short-Term Memory networks have been observed as the most effective solution (Talati, D., Patel, M., & Patel B. (2022) IV methodology). The basic idea of traditional statistical models such as linear regression and Autoregressive Integrated Moving Average (ARIMA) is that the data is stationary, that its statistical features such as variance and mean don’t change over time.

Because LSTM has feedback connections, as compared to standard neural networks, it can handle complete data sequences as alternative to only handling single data points. In addition to this, it is very good at identifying and forecasting patterns in sequential data, such as speech, text and time series. By extracting important insight from sequential data, LSTM has developed into an effective tool in deep learning and artificial intelligence that has assisted advances across a range of industries.

Stock prices, on the other hand, show seasonality, patterns, and trends despite not being stationary. LSTMs are ideally suited for stock market prediction because they can manage this non-linear correlation within the data.

**LSTM Architecture**

By understanding the LSTM’s architecture, we shall see in the following section how it solves this issue. LSTM functions quite similarly to an RNN cell at a high level, and this is how the LSTM network operates internally. As seen in the diagram below, the LSTM network design is divided into three components, each of which carries out a distinct task.

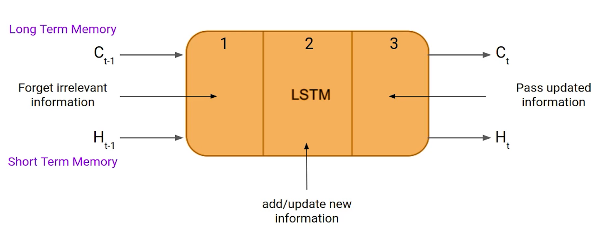


Fig: taken from (https://www.analyticsvidhya.com/blog/2021/03/introduction-to-long-short-term-memory-lstm/)

From the above figure as shown, an LSTM unit consists of three components referred to as gates. They regulate the information that enters and exit the memory cell, also known as the LSTM cell. The first gate is known as the “Forget gate”, where data from the preceding timestamp is either significant and should be remembered, or it can be ignored. Further, the cell attempts to learn new information form the input to this cell at the second gate, which is referred to as the “input gate”. The cell transfers the modified data from the current timestamp to the subsequent timestamp in the final gate, which is called the “output gate”. This LSTM cycle is considered as a single-time step.

An LSTM has a hidden state, just like a basic RNN, where “H(t-1)” is the hidden state of the timestamp that was previously recorded and “Ht” is the hidden state of the timestamp that is currently recorded. Furthermore, the cell state of an LSTM is denoted by “C(t-1)” for the past timestamp and “C(t)” for the present timestamp, respectively. In this case, long-term memory refers to the cell state and short-term memory to the hidden state.

Let’s use the equation to better understanding how these gates function in the LSTM architecture.

Forget Gate:

In this gate, we must figure out whether to keep or remove the data from the previous time step. Below is the forget gate equation.

FORGET EQUATION

Here, Let’s understand the working of the equation,

Xt: It’s the timestamp’s current input.

Uf: The input’s associated weight.

Ht-1: The previous timestamp’s hidden state

Wf: The weight matrix related to the hidden state.

It then undergoes the use of a sigmoid function. As a result, f and t will become a number 0 and 1. And then the cell state of the earlier timestamp is then multiplied by this f and t.

The input and output gate equation, which is essentially the same as that of the forget gate. Likewise, because of the sigmoid function, its value will also range between 0 and 1. The equations below can be used to feed new data into the input gate and extract the output from the output gate.

Input gate new information:

To handle new information, a sigmoid function is used to modify the existing data, and that’s how the neural networks operate rather than classifying information to be important or not. As a result, the data has been updated completely. LSTM, on the other hand, uses a mechanism to transmit the data known as cell states while executing minimal addition and multiplication changes to the data.

Here is the equation to calculate the new information.

NEW INFO equation

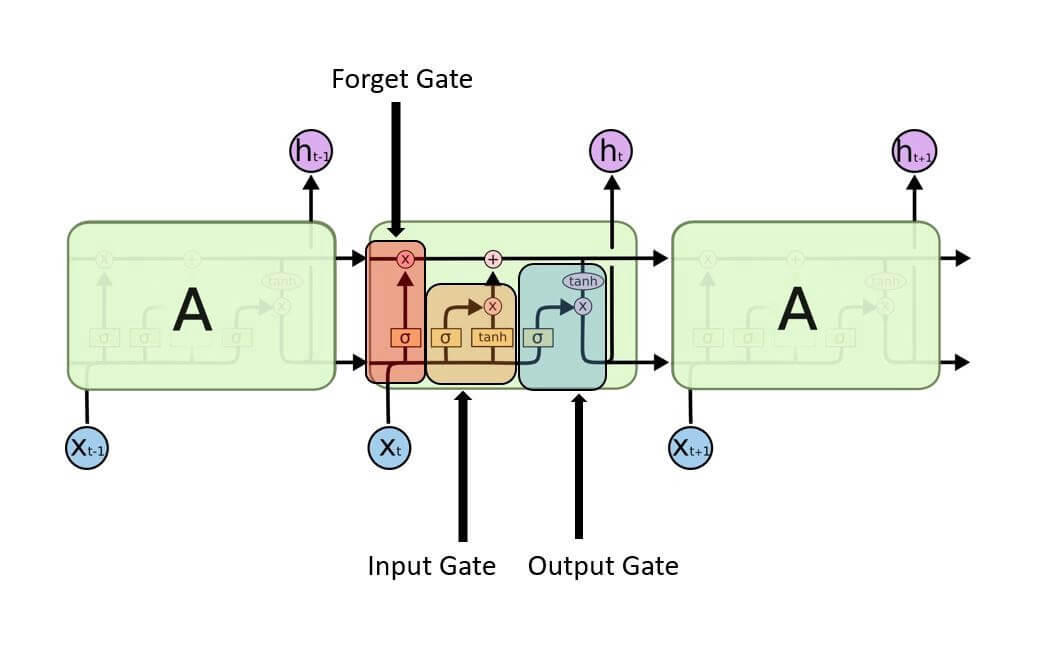
The function of a hidden state at timestamp t-1 and input x at timestamp t now identifies the new information that has to be provided to the cell state. And “tanh” is the activation function in this case. The value of new information will range from -1 to 1 as a result of the “tanh” function. Information is added to the cell state at the current timestamp if the value of Nt is positive, and subtracted from the cell state if it is negative. The cell state at the current timestamp is represented by Ct-1 in this case, and the other values are those we previously calculated.

Output gate:

In this gate we will use Ot and tanh of the updated cell information to compute the current hidden state as shown below.

OUTPUT gate equation

It turns out that the present output and long-term memory (Ct) determine the hidden state. Simply use the “SoftMax” activation on the hidden state Ht if you need to extract the current timestamp output. Finally, the token with the highest score in the output is used to make the prediction.



**Figure 3: Long Short-Term Memory Network**

**Essential elements of LSTM:**

* **Memory Capability**: Because LSTMs can remember information over time, they are perfect for capturing long-term dependencies in sequential data.
* **Gated mechanism**: LSTMs use gates to manage information flow, enabling them to store important data and eliminate unnecessary information.

**Data Collection**

**Data Preprocessing**

**Data Feature Selection**

Reference:

1. Stock Market Prediction Using LSTM Technique, <https://www.ijraset.com/research-paper/stock-market-prediction-using-lstm-technique>
2. Enhancing Stock Market Prediction Through LSTM Modeling and Analysis, <https://eudl.eu/pdf/10.4108/eai.2-6-2023.2334692>