Data analytics for Level-2 financial-market data

1st Samiksha Jain
MSc Data Science
University of Bristol
qi22058@gmail.com

2nd Yujie Wang
MSc Data Science
University of Bristol
ni22398@bristol.ac.uk

3rd Xiaoyu Wang
MSc Data Science
University of Bristol
oq22541@bristol.ac.uk

Abstract*—This report presents an in-depth analysis of Level-2 financial-market data using limit order book (LOB) information and the application of Autoregressive Integrated Moving Average (ARIMA) and Long Short-Term Memory (LSTM) machine learning algorithms. The study aims to investigate the predictive accuracy and effectiveness of these algorithms in forecasting short-term price movements in the financial market. The data sample comprises of high-frequency LOB financial data provided by HSBC over a specified period. The methodology entails pre-processing the LOB data and extracting essential features before implementing ARIMA and LSTM models for prediction. Evaluation metrics such as mean squared error (MSE), mean absolute error (MAE), and coefficient of determination (R-squared) are used to compare the performance of various models. The results show that the LSTM model outperforms the ARIMA model in terms of predictive capabilities, with much lower error rates and higher R-squared values. The study's results add to the existing body of knowledge by highlighting the proficiency of machine learning algorithms, specifically LSTM, in forecasting financial market trends using Level-2 data. This has potential implications for market players, including traders and investors, as it offers valuable guidance for their decision-making endeavours. Additionally, the research emphasizes on the significance of incorporating high-frequency limit order book data in financial market assessments, as it enables a more detailed comprehension of market behaviour.

I. Introduction

Financial market analysis has evolved considerably in recent years and more and more high frequency trading is taking place, prompting researchers to turn their attention to more sophisticated analytical techniques in order to gain a deeper understanding of market behaviour. Machine learning algorithms have emerged as a popular method of forecasting financial market movements in this context. This type of data is highly volatile and requires a time-series analysis.

The purpose of this research is the analysis of Level 2 Financial Market Data from HSBC, which includes data of financial assets in a limit order book. Level 2 financial market data, also known as a limit order book, is an extended set of trading information that includes details on the order book, such as buy and sell orders at different price levels, providing insights into market depth, liquidity, and supply and demand dynamics for a particular financial instrument. Unlike traditional methods, for high frequency trading, the LoB looks beyond historical data to provide a more comprehensive set of market characteristics by focusing on recent data related to intermediate prices. This approach provides a deeper understanding of market dynamics, which is essential for accurate predictions in high frequency trading environments.

This report aims to answer the following questions: What are the efficiencies of ARIMA and LSTM machine learning algorithms when predicting short term price moves using Level Financial Market2 data? The hypothesis is that, given the fact that LSTM models are able to detect complex timescale patterns, they will be more accurate in predicting accuracy than an ARIMA model.

Before implementing the predictive models, an exploratory data analysis has been carried out to gain a detailed understanding of the data. This early analysis provides an overview of data patterns, trends and possible outliers as well as a detailed description of the underlying characteristics or relationships among variables. In order to ensure that accurate and efficient prediction models are developed, the decisions are made based on the correct pre-processing steps and selected features based on the initial exploratory data analysis.

In addition to implementing ARIMA and LSTM machine learning algorithms, the proposed method includes a simulation environment to assess its efficiency. The simulator enables controlled testing of the predictive models under various market conditions, allowing for a comprehensive evaluation of their performance and robustness. This simulation offers valuable insights into the potential applicability of the algorithms in live trading situations by attempting to replicate realistic market scenarios.

II. Literature Review

Before diving into this analysis, three main areas of literature were studied: Level 2 Analysis of Financial Market Data, Traditional Time Series Forecast Methods and Machine Learning Algorithm for Financial Market Forecasting. It is important to note that trading signals should be used with caution and cannot be solely relied upon for making investment decisions. It is also important to consider the costs associated with trading, such as transaction fees and bid- ask spreads, when evaluating the profitability of a trading strategy.

1. Level-2 Financial-Market Data Analysis:

By providing data on pending orders, market depth and liquidity, Level 2 Financial Market Data provides deeper insight into market dynamics, including information on the limit order book. Over time, research in the area of data analysis relating to LOB has evolved significantly. The book, Limit Order Books, provides an extensive overview of the limit order book [1], while Gould analyze the LOB data from a quantitative finance perspective [2] . Further exploring the world of financial markets through an examination of trades, quotations and prices is done in their complete book which demonstrates the importance of LOB data for understanding market behaviour.

2. In the timer series forecasting domain, the ARIMA model has been massively used to predict time series problems like stock price prediction. In their paper, used ARIMA model to predict stock prices [3]. They used daily closing prices on the Nigerian Stock Exchange AllShare IndexNSE Nigeria, they showed that the ARIMA model can be applied and efficiently used for stock price forecasting. To determine the accuracy of the ARIMA model, performance metrics such as MSE and MAPE were used. Their research indicated that the ARIMA model can be successfully used to predict the time dependence of stock prices and provide accurate forecasts, thereby supporting the potential application of these models for financial market forecasting.

The LSTM is a special type of recurrent neural network for processing and predicting important events in time series with relatively long intervals and delays. The effectiveness of LSTM over simpler models such as Random Forest and Support Vector Machines is discussed in the study conducted by 2021 [4]. Results indicated that the LSTM model is superior to other algorithms, demonstrating its ability to learn long term dependencies and provide accurate stock price forecasts. It underlines the potential of LSTM models for financial market prediction and their superior performance compared to conventional algorithms like ARIMAs in some scenarios. Using extremely granular data like financial data to predict stocks is used in another research [5].

3. There has been some research in the automated trading world, for example, Bristol Stock Exchange or BSE which ”is a minimal simulation of a financial exchange running a limit order book (LOB) in a single tradeable security” [6]. Although it ignores complexities that comes with a financial trade, it is a useful tool that adapts to the ever changing financial data and learns from it to make predictions.

III. Methodology

Two models are used to do the initial predictions for stock prices with the OHLC data. The models leverage the closing stock prices in the data to simulate and make simple predictions.

A. ARMA

A moving average model uses past forecast errors in a regression-like model instead of using the past values of the forecast variable in a regression [7]. The ARMA Model is the combination of Autoregressive (AR) model and Moving average (MA) model. Autoregressive models suppose a linear relationship between current returns and their own history. The hyperparameters of the autoregressive models could be observed from PACF(Partial Autocorrelation Function) plots. Also for the moving average model, the new signals therefore have not only an immediate effect but also a delayed effect [11].

A key aspect of the ARMA model is the Autoregressive (AR) component. The basic assumption underlying the Autoregressive Component is that past values of a series have information which might be useful in predicting subsequent value levels. The hyper-parameters of the moving average models could be observed from ACF(Autocorrelation Function) plots. A time series model x_t is an autoregressive moving average model of order p, q, ARMA(p, q), if:

$$x_t = c + \varepsilon_t + \sum_{i=1}^{p} \varphi_i x_{t-i} + \sum_{i=1}^{q} \theta_i 1_{t-i} \tag{1}$$

where c is the constant, φ is the autoregressive model's parameters, θ is the moving average model's parameters, and ε is the white noise. Additionally, p and q are separately the orders of the autoregressive polynomial and the moving average polynomial.

In addition, returns describes what the percent change in stock price between one day and the next, then used for doing the simulation.

Because the ARMA's autocorrelation structure do not change over time, this model is used on the OHLC data instead of the level-2 data.

B. ARIMA

In real world applications, time series data is often non-stationary and demonstrate such behaviour through trends or seasonality. In this case, the ARIMA model is more suited as it includes an extra step called ”differencing” to transform the non- stationary data to stationary data. The ARIMA model is differentiated from ARMA model with an extra component, ’I’. The ”I” in this model stands for Integrated measures how many non-seasonal differences are needed to achieve the stationarity [12] [13]. The projected value for the ARIMA model is a homogeneous combination of earlier values and random errors, as follows [12]:

$$\begin{aligned} Y_t = \theta + \phi_1 Y_{t-1} + \phi_2 Y_{t-2} + \cdots + \phi_p Y_{t-p} + \\ \varepsilon_t - \theta_1 \varepsilon_{t-1} - \theta_2 \varepsilon_{t-2} - \cdots - \theta_q \varepsilon_{t-q} \end{aligned} \tag{2}$$

It can be interpreted as:

Prediction = constant + linear combination lags of Y + linear combination of lagged forecast errors

ARIMA is a combination of 2 models AR(Auto Regressive) and MA(Moving Average). It has 3 hyperparameters; p(auto regressive lags),d(order of differentiation),q(moving average) which comes from the AR, I and MA components respectively. The I component works to link the AR and MA components in a model so they can be integrated effectively by applying differencing. Differencing, d, involves calculating the difference between consecutive observations in a time series [14]. It can be represented as:

$$\Delta X_t = X_t - X_{t-1} \tag{3}$$

where, ΔX_t is the first difference at time t, X_t is the original value at time t, and X_{t-1} is the value at time t-1.

Note that, in some cases, a single differencing step may not be enough to make the time series stationary, and higher-order differencing may be required.

The data contains less noise compared to when higher orders of differencing are applied, which results in an increase in noise. Therefore, the choice of first-order differencing for the model seems most suitable. This can also be confirmed by examining an autocorrelation (ACF) plot. This is how the process of obtaining the "d" value can be carried out [14] [15].

A questions thus arises, "How can we choose the best values of p, d and q?" The model selection process in time series analysis aims to find the best combination of autoregressive (AR), moving average (MA), and differencing orders (p, d, and q) that yields the most accurate model for forecasting.

AIC, is a model selection criterion that estimates the relative quality of a statistical model by considering both the likelihood of the model given the data and the number of parameters. It is represented as:

$$\text{AIC} = -2 \cdot \log(L) + 2 \cdot k \tag{4}$$

Where, L is the maximum likelihood of the model, and k is the number of estimated parameters in the model. The goal is to minimize the AIC value, so the model with the lowest AIC is considered the best fit.

After model selection, it is crucial to verify the model's validity by analyzing the residuals to prevent over-fitting of data. Residuals can be denoted as:

$$\epsilon_t = Y_t - \hat{Y}_t \tag{5}$$

Here, ϵ_t represents the residual at time t, Y_t is the actual observed value of the time series at time t, and $\hat{Y}_t$ is the predicted value of the time series at time t based on the fitted model.

In principle, the residuals of a good fitted time series model should be similar to white noise with the following properties: constant mean, constant standard deviation or variation and correlation between lags is 0 (no seasonality).

In order to evaluate the performance of a model and gain insight into specific aspects of its behaviour, a number of metrics can be used, such as root mean squared error (RMSE) and mean absolute percentage error (MAPE) have been used.

The MAPE can be denoted as:

$$\text{MAPE} = \frac{1}{n}\sum_{t=1}^{n}\left|\frac{A_t - F_t}{A_t}\right| \tag{6}$$

Here, n represents the number of data points, A_t represents the actual value at time t, and F_t represents the forecasted value at time t [16].

The mathematical formula for RMSE is as follows:

$$\text{RMSE} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2} \tag{7}$$

Here, n represents the number of observations, y_i is the actual observed value, and $\hat{y}_i$ is the predicted value based on the model [16].

C. Models For Mid-Price Forecasting

Three distinct tasks are undertaken to predict the mid-price of the LOB data. These tasks include predicting the mid-price itself, a two-class classification of mid-price changes, and a three-class classification of mid-price changes. A variety of nueral networks have been used to accomplish this task:

1) RNN: Initially, a model utilizing a recurrent neural network (RNN) consisting of a single layer of straightforward RNN units is employed for the two-class classification problem of the mid-price. This is then followed by a time-distributed dense layer featuring a softmax activation function. Although simple RNNs are useful, they suffer from the vanishing gradient problem, where the gradient diminishes significantly during backpropagation. To overcome this issue, the LSTM model with 32 units is used to improve the task's results.

2) LSTM: The process of mid-price prediction mainly uses the LSTM model, which is an RNN variation that can learn long-term dependent information while avoiding overfitting. An LSTM unit has three gates that assist the unit in removing or adding information: a forget gate, an input gate, an output gate, and a cell. The forget gate selects which information from the unit should be deleted, the input gate determines which information should be saved, the output gate gives the output information, and the cell records the unit's status information. In this case, the LSTM model utilizes deep learning and comprises an LSTM layer and a time-distributed dense layer. The LSTM layer is designed to handle sequential data and consists of 32 units with a hyperbolic tangent activation function and a hard sigmoid recurrent activation function. [17]. In addition, the time-distributed dense layer is used to apply the same dense layer to every time step of the output sequence from the LSTM layer.

3) ConvRNN: When it comes to predicting mid-price changes in a three-class classification problem, the ConvRNN model is a reliable choice. This model combines convolutional and recurrent layers. This architecture is useful for time-series data with spatial features and can accurately classify different types of data.

4) ConvLSTM: The ConvLSTM [18] model is also used in the three-class classification problem, which consists of several convolutional layers of varying kernel and filter sizes, followed by a single LSTM layer and a fully connected output layer with a softmax activation function. The convolutional layers are responsible for extracting features from the input data, which is a 3D tensor that represents a time series of 2D feature maps. The extracted features are then processed by the LSTM layer, which produces a fixed-length output vector that is fed to the final dense layer for classification. The model is optimized using the Adam optimizer with a specified learning rate and epsilon value [19].

D. Trading Simulator

The LSTM model predicts the midrice for LimitOrderBook, and the prediction results can be used as observation indicators for market fluctuations. The trading simulator selects trading strategies based on the predicted market fluctuations.

For HSBC, buying at a low price when the market trend is down and selling at a high price when the market trend is up can obtain more profits. The market is constantly fluctuating, but the magnitude of these fluctuations are different. Trading when the fluctuation at its maximum can lead to more profits. Therefore, setting a trading threshold where trades are executed when the price increases or decreases by 5% can be beneficial.

IV. Data Description / Preparation

The data for this research is of two forms; tapes and LOB data.

A. OHLC: Open, High, Low, Close

Each data point in the OHLC dataset consists of four components, which provide a comprehensive view of the price movements during that period:

1. Open: When the market opens for a given period, such as the beginning of the day, hour or minute, the opening price refers to the first traded price of the asset.

2. High: The highest price in this period represents the maximum value that an asset has been traded at. It reflects a strong buying pressure or an optimistic outlook, indicating that the asset's peak value has been achieved.

3. Low: The lowest price is the minimum traded price of the asset.

4. Close: The closing price the last traded price of the asset for when that period comes to a close.

OHLC data for time series forecasting is useful as they are capable of providing a consistent view of the market activities in one specific period and can give insights into future price movements. The candlestick chart below shows the price fluctuations over a period of time. When prices continue to move higher, the current market trend is up. When prices are consistently lower, the current market trend is down. Changes in market trends can present some good trading opportunities [20].

OHLC data also includes many features. To explore the relationships among these features, a correlation analysis was conducted on the OHLC data, and a heatmap was generated to visualize the correlations. This graph shows the correlation between the data is between 0.97 and 0.99, proving that they are strongly correlated with each other.

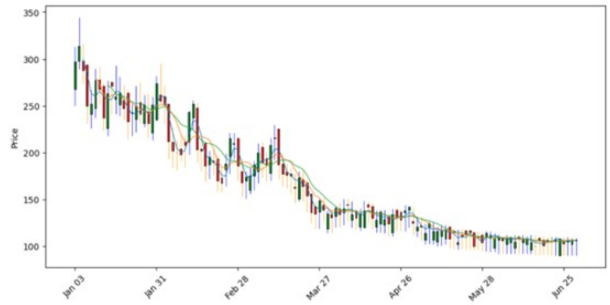

Fig. 1. OHLC Data Candlestick Chart

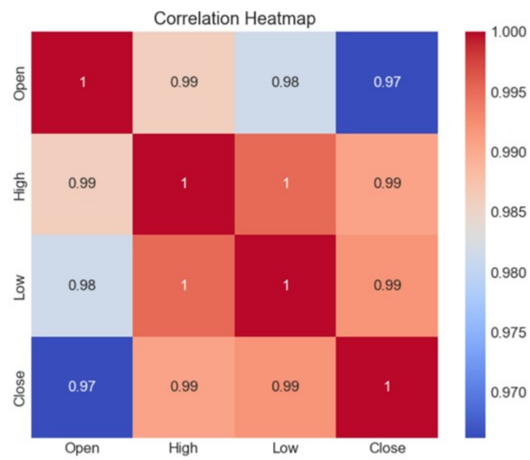

Fig. 2. OHLC Data Correlation Heatmap

B. Limit Order Books

The LOB file contains a series of exchange (Exch0) at different time points. Each entry in the data consists of three elements:

1. Timestamp: The first element represents the time of the data in seconds, relative to an arbitrary starting point (e.g., 0.000, 0.093, 1.240 etc).

2. Exchange: The second element indicates the name of the exchange where the LOB is recorded (eg., Exch0).

3. Bid and Ask Orders: The third element is a list containing two sub-lists, one for bid orders and one for ask orders. Each sub-list has a label (either 'bid' or 'ask') and a list of orders. Each order is represented as a list with two elements: the price and the quantity of shares available at that price level.

For example, if an entry is [2.666, Exch0, [['bid', [[1, 6]]], ['ask', [[239, 2], [248, 4], [615, 5]]]]], it displays the state of the limit order book on the Exch0 exchange at 2.666 seconds, with one bid order and three ask orders at different price levels and quantities.

The Limit Order Book (LOB) displays a wealth of bidding and asking price information, but not all of this data is useful. In practical work, data from the first and last hours is often discarded. The figure below shows the relationship between transaction prices and mid-prices. The horizontal axis represents cumulative time, and the vertical axis represents prices. This is a plot illustrates that most of the transaction prices are close to the mid-price. Therefore, the mid-price can to some extent reflect the market's changing trends and serves as an important feature for market trend prediction.

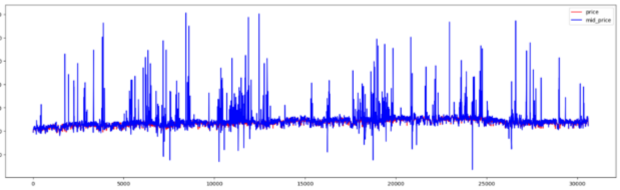

Fig. 3. Relationship between transaction price and mid-price.

C. Feature Extraction

To predict the mid-price, this experiment employs the basic feature set [8] for the mid-price prediction task, as depicted in the below figure.

Feature Set	Description
Time Intervals	$u_1 = \{t_{i+1} - t_i\}_{i=1}^{k}$
5(=n)-level LOB Data	$u_2 = \{P_i^{ask}, V_i^{ask}, P_i^{bid}, V_i^{bid}\}_{i=1}^{n}$
Spread	$u_3 = \{(P_i^{ask} - P_i^{bid})\}_{i=1}^{n}$
Weighted Mid-Price	$u_4 = \{(V_i^{b}P_i^{a} + V_i^{a}P_i^{b})/(V_i^{b} + V_i^{a})\}_{i=1}^{n}$

Fig. 4. Basic Feature Set

In the original limit order book (LOB) data, the maximum level of the LOB can reach up to 15 levels. However, in this experiment, only the first five levels of the LOB data are used due to the fact that deep-level data has less impact on the mid-price and contains numerous null values. Moreover, the null values in the other levels of data are handled by generating fake prices using a specific method [21] and subsequently filling them in, except for the first level of data, which is removed from the data set.

In addition, two additional features, namely Spread [8] and Weighted Mid-Price [9], are added to the feature set. These two features belong to the category of time-insensitive features [8] and have a high correlation with Mid-Price. Therefore, they are added to the feature set to enhance the predictive accuracy.

For the prediction of the mid-price changes, there are two types of classification problems: two-class classification problems with two states, which are Up and $Down$ [10], and three-class classification problems with three states: Up, $Stationary$, and $Down$ condition for the mid-price movement. The changes in the mid-price are defined by means of the following calculations [22]:

$$l_t = \begin{cases} [1,0,0], & \text{if } \dfrac{S(t)}{M_m(t)} < -\alpha \\ [0,1,0], & \text{if } -\alpha < \dfrac{S(t)}{M_m(t)} < \alpha \\ [0,0,1], & \text{otherwise} \end{cases} \tag{8}$$

where

$$S(t) = M_p(t) - M_m(t) \tag{9}$$

$$M_p(t) = \frac{1}{K}\sum_{i=1}^{K} MP(t+1) \tag{10}$$

$$M_m(t) = \frac{1}{K}\sum_{i=1}^{K} MP(t-1) \tag{11}$$

where $MP(t)$ is the mid-price at time t, $M_p(t)$ is the average of the future mid-price events with window size K [22], $M_m(t)$ is the average of the previous mid-price events with window size K, and α determines the significance of the mid-price movement [22]. The Eq.(9) indicates the encodes for the three states, where $[1,0,0]$,$[0,1,0]$,$[0,0,1]$ separately stand for the states $Down$, $Stationary$, and Up [17].

D. Data Normalisation

In order to ensure accurate and reliable predictions from machine learning algorithms, normalisation is a crucial step in data preprocessing that helps in improving the performance, interpretability, and reliability of machine learning models. It ensures that all features have the same impact on the model's predictions and that the model is less sensitive to outliers and extreme values by scaling the data to a standard range.

For the purpose of this study, z-score normalisation is used because it effectively standardizes the data, is well matched with different types of distribution, reliably normalizes outliers to improve comparability between datasets [23]. The z-score method aims to standardize the data by transforming it to have a mean of 0 and a standard deviation of 1. After normalising the limit order data, the mid-price is used to create labels that represent the direction of price changes.

$$pt = \frac{p_a^{(1)}(t) + p_b^{(1)}(t)}{2} \tag{12}$$

Where, $p_a^{(1)}(t)$ is the best ask price (the lowest selling price) at time t $p_b^{(1)}(t)$ is the best bid price (the highest buying price) at time t.

Although no order can be exact at the mid price, it expresses a broad market value of an asset and is often quoted when one is trying to represent asset prices in a single number [23].

V. Results and Discussion

The experimental results of the study are discussed below in sections.

A. ARIMA Model with OHLC Data

The analysis began with the use of Open, High, Low, and Close (OHLC) data to forecast stock prices with the ARIMA model. To identify the optimal ARIMA model, the following analysis was performed:

1) Differencing the time series to achieve stationarity

2) Examination of the autocorrelation function (ACF) and the partial autocorrelation function (PACF) of the time series to obtain an initial understanding of the underlying patterns and dependencies.
3) Utilization of model selection criteria, such as Akaike Information Criterion (AIC), to determine the optimal combination of autoregressive (AR) and moving average (MA) components (p, d, and q).

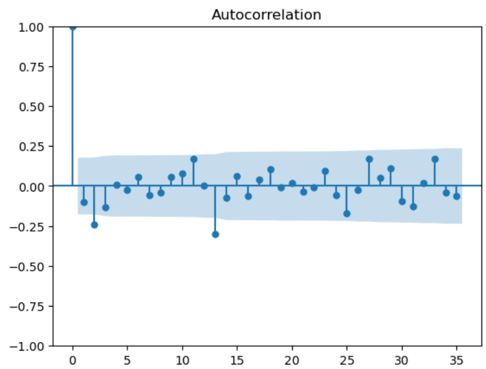

Fig. 5. Examination of the autocorrelation function (ACF)

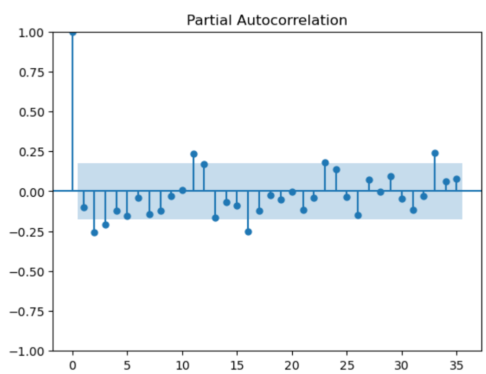

Fig. 6. Examination of the partial autocorrelation function (PACF)

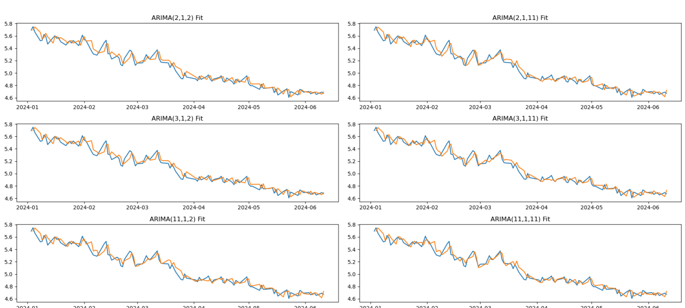

Fig. 7. Model selction for the data

Notable spikes beyond the blue boundary in the PACF plot indicate possible values for p. In this instance, the text highlights the selection of orders 2, 3, and 11 based on the PACF plot.

The objective of this step is to predict stock returns for the following day with an AR(2,2) model. A decision strategy is used, based on the forecast return. The stocks shall be purchased the following day, if the expected return does not exceed a preset threshold. Conversely, stocks will be sold on the next day. This approach, aimed at exploiting the expected short term market trends, is a combination of time series analysis and systematic trading strategies.

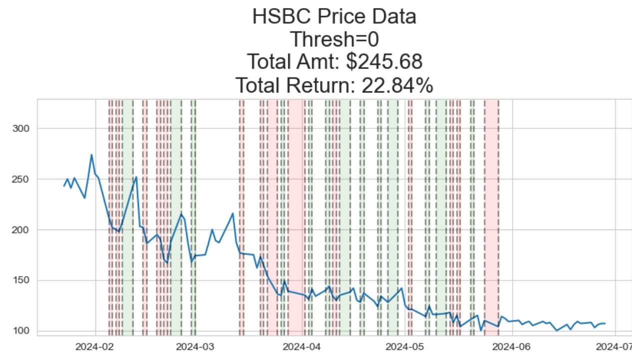

Fig. 8. The simulation using ARMA(2,2) Model and its returns

Following the selection of the best ARIMA model, the residuals have been analysed to ensure that they show white noise characteristics, indicating that the model has adequately captured the underlying data. Based on Fig 9, the residuals obtained from the current model exhibit stationarity, indicating that the current model has been well-fitted.

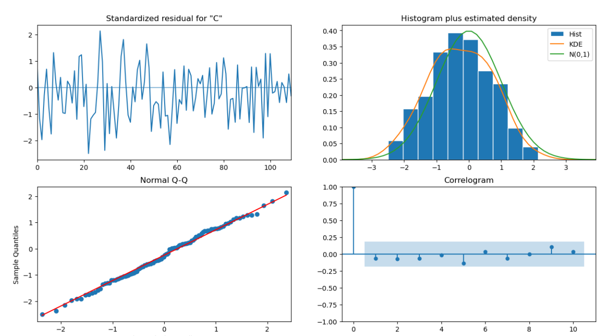

Fig. 9. The Residuals For The ARIMA(2,1,2) Model

The following results on stock price forecasting have been achieved by applying the ARIMA model to OHLC data:

The LOB data has been incorporated in this analysis to improve the accuracy of stock price forecasts. The same procedure was applied as for the OHLC data again, this time using a mid market price as one of the valuation features. After fitting the ARIMA model to the LOB data, the following results for the stock price forecasts were obtained:

A MAPE of 0.0060 means that, on average, the absolute percentage error between the predicted values and the actual values is 0.6. The square root of the average squared deviation between predicted values and actual values is an RMSE of 0.0715. In both cases, a lower value indicates better model

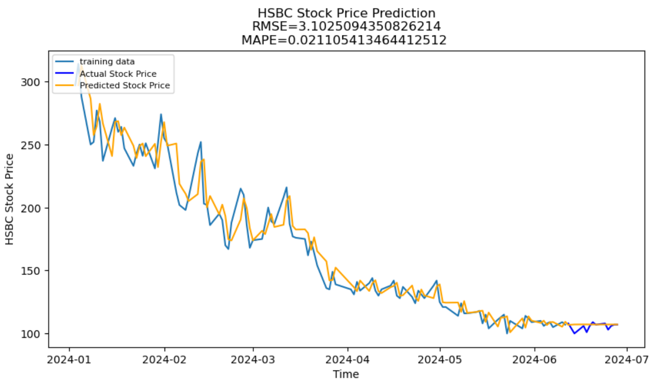

Fig. 10. The Comparison Between Actuals and Predictions

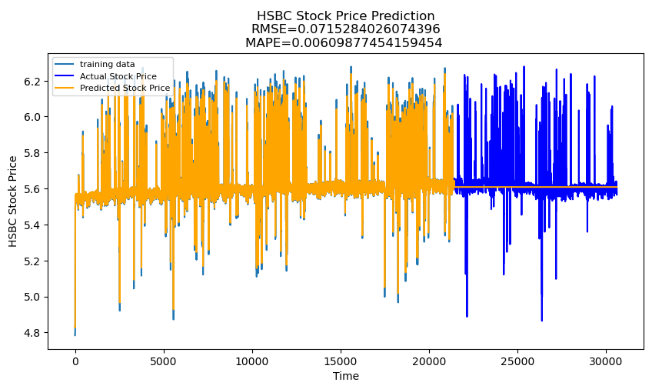

Fig. 11. ARIMA mid-price result

performance. The MAPE of 0.0060 and RMSE of 0.0715 can be considered as decent results, however it is essential to carry out a comparison with different models to determine their relative performance.

To explore the potential for improved forecasting performance, an LSTM model was implemented and its results were compared with those of the ARIMA model.

B. Mid-Price Tasks

1) The Prediction Of Mid-Price: The LSTM models are trained with 500 epochs, 100 batch sizes, and 30 time steps for the prediction of mid-price, and the results are depicted in the Fig 12. The findings suggest that the LSTM model with a linear activation function outperforms other models in predicting the mid-price of the Limit Order Book(LOB). It is noteworthy that this model's performance is evaluated on a testing set comprising 178,743 data points, while it is trained on a large training set containing 714,970 data points.

Activation Function	MAE	RMSE	MAPE
Linear	0.1839	0.4152	0.0006
Relu	18.6500	26.8959	0.0738

Fig. 12. Experiment Results For The Mid-Price Predictions

2) Two-class Classification Of Mid-Price Change: Under the same conditions as described in the previous section, the RNN and LSTM models are applied to perform binary classification for the mid-price, predicting whether it will increase or decrease in the next time step. The LSTM model achieves a slightly higher accuracy than the RNN model. However, the LSTM model does not reach its optimal training performance, indicating potential for further improvement.

Model	F1-Score		Precision		Recall		Accuracy
	Down	Up	Down	Up	Down	Up	
RNN	0.7567	0.7562	0.7563	0.7562	0.7562	0.7563	0.7563
LSTM	0.7591	0.7590	0.7591	0.7590	0.7590	0.7591	0.7591

Fig. 13. Experiment Results For The Mid-Price Binary Classification

3) Three-class Classification Of Mid-Price Change: The performance of the ConvRNN and ConvLSTM models under different parameters is shown in Fig 14. Although the ConvRNN model appears to have the highest accuracy at α=0.005, the F1-Score for the label "Stationary" is 0 due to the scarcity of such data in the dataset. Therefore, this experiment increases the value of α to 0.1. Under the same conditions, the ConvLSTM model has higher accuracy compared to the ConvRNN. The Fig 15 and 16 show the performance of the ConvRNN_7 and ConvLSTM_3 models, respectively. As shown in the figures, under the same conditions, the ConvLSTM model tends to converge more. Among all the experiments, the ConvLSTM model performs the best with K=30, α=0.1, and Timestep=50, with F1-Scores of 0.71, 0.43, and 0.62 for "Down", "Stationary", and "Up" labels, respectively.

Model	Epochs	K	Alpha	Timestep	Features	Accuracy %
ConvRNN_1	200	50	0.002	100	23	61.37
ConvRNN_2	100	50	0.005	100	23	65.23
ConvRNN_3	100	30	0.05	50	23	60.73
ConvRNN_4	100	50	0.1	50	23	58.17
ConvRNN_5	200	50	0.1	50	23	51.53
ConvRNN_6	80	50	0.1	50	23	60.44
ConvRNN_7	100	30	0.1	50	23	58.18
ConvLSTM_1	80	50	0.1	50	23	61.03
ConvLSTM_2	100	50	0.1	50	23	59.85
ConvLSTM_3	100	30	0.1	50	23	63.48
ConvLSTM_4	150	30	0.1	50	23	63.67
ConvLSTM_5	100	30	0.1	100	23	61.31
ConvLSTM_6	100	30	0.1	50	19	60.4
ConvLSTM_7	100	30	0.1	50	19	56.03

Fig. 14. Experiment Results For The Mid-Price Three-Class Classification

C. Trading Simulator

Based on the predicted result, it is possible to determine how the price trend changes over time in the time series. Set the initial parameters for the simulator: set the initial funding to 100,000 and the original holding to 10,000. when the trade signal is "buy", the simulator will select the lowest price to buy; when the trade signal is "sell", the simulator will select

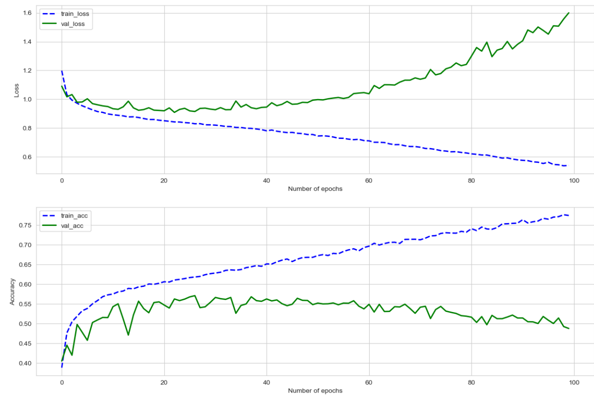

Fig. 15. Training and Validation Accuracy/Loss vs. Epochs For ConvRNN

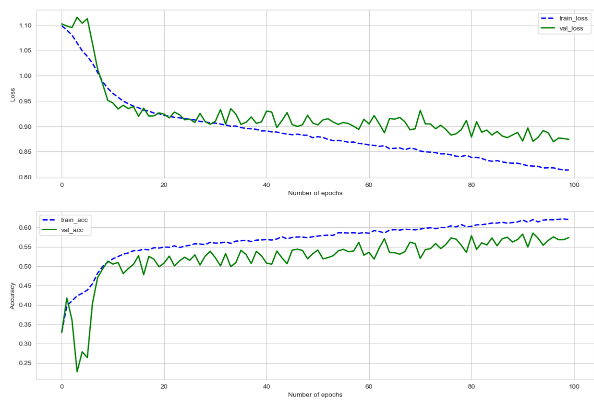

Fig. 16. Training and Validation Accuracy/Loss vs. Epochs For ConvLSTM

the highest price to sell. The result is a profit of 638,736 currency units.

VI. Further Work and Improvement

Although the selected models output decent results, to enhance the forecasting performance of the study, various modeling techniques can be explored, including GARCH, VAR, and advanced deep learning methods like Transformer networks and attention mechanisms.

It is also possible to consider hybrid models, which combine traditional time series models with deep learning techniques. Combining predictions from several models, e.g.ARIMA, LSTM and the rest of the prediction models can be used to achieve better overall accuracy by using ensemble models which employ techniques like stacks, bags or boosts.

By analysing trading volume, volatility and liquidity, the research may further explore the impact of various market structure factors on performance forecasts. In addition, an evaluation can be made of adaptive models that use real time updates through online learning techniques in order to assess their impact on forecasting performance.

The trading simulator in this study is very basic one, so one idea is to expand the simulator with more complex trading strategies, such as market making,pairing of trades, arbitrage and High Frequency Trading Algorithms.

VII. Conclusion

In conclusion, this report presents a comprehensive analysis of Level 2 financial market data, with a focus on forecasting stock prices using ARIMA and LSTM models. The ARIMA model is used to estimate stock prices at the start of this study, providing a baseline for comparison. The LSTM model has been subsequently implemented, demonstrating the potential to provide more accurate and reliable forecasts in different market conditions.

From there, the study goes on to compare ARIMA and LSTM models in order to demonstrate their adaptability and ability to predict stock prices. The report also stresses the need for data prioritisation, notably normalization of financial time series information, to improve machine learning algorithms' effectiveness in this context.

A trading simulator has been used to assess the real applicability and usefulness of the proposed LSTM model. In the simulation environment, it is possible to conduct controlled simulations of predictive models in different market scenarios so that they can be thoroughly evaluated on their performance and robustness. In its attempt to replicate realistic market conditions, the results obtained from a trading simulator have provided valuable information on the potential benefit of an LSTM model in live trade situations.

Not only does this report provide a solid basis for future stock price forecasting research using advanced machine learning techniques, but it also highlights the importance of practical evaluation in addition to traditional performance metrics.

References

[1] F. Abergel, M. Anane, A. Chakraborti, A. Jedidi, and I. Muni Toke, *Frontmatter*, ser. Physics of Society: Econophysics and Sociophysics. Cambridge University Press, 2016.

[2] J.-P. Bouchaud, J. Bonart, J. Donier, and M. Gould, *Trades, Quotes and Prices: Financial Markets Under the Microscope*. Cambridge University Press, 2018.

[3] A. A. Ariyo, A. O. Adewumi, and C. K. Ayo, "Stock price prediction using the arima model," in *2014 UKSim-AMSS 16th International Conference on Computer Modelling and Simulation*, 2014, pp. 106–112.

[4] Pramod and M. Pm, "Stock price prediction using lstm," *Test Engineering and Management*, vol. 83, pp. 5246–5251, 01 2021.

[5] S. Mehtab, J. Sen, and S. Dasgupta, "Robust analysis of stock price time series using cnn and lstm-based deep learning models," 11 2020, pp. 1481–1486.

[6] D. Cliff, "Bse: A minimal simulation of a limit-order-book stock exchange," in *30th European Modeling and Simulation Symposium (EMSS 2018)*, M. Affenzeller, A. Bruzzone, E. Jimenez, F. Longo, Y. Merkuryev, and M. Piera, Eds. Italy: DIME University of Genoa, Oct. 2018, pp. 194–203.

[7] R. J. Hyndman and G. Athanasopoulos, *Forecasting: principles and practice*, 3rd ed. Melbourne, Australia: OTexts, 2021, accessed on May 9, 2023. [Online]. Available: https://otexts.com/fpp3/

[8] A. Ntakaris, M. Magris, J. Kanniainen, M. Gabbouj, and A. Iosifidis, "Benchmark dataset for mid-price forecasting of limit order book data with machine learning methods," 2018.

[9] S. Stoikov, "The micro-price: a high-frequency estimator of future prices," *Quantitative Finance*, vol. 18, pp. 1959–1966, 2018. [Online]. Available: https://www.tandfonline.com/doi/full/10.1080/14697688.2018.1489139

[10] D. Palguna and I. Pollak, "Mid-price prediction in a limit order book," *IEEE J. Sel. Top. Signal Process.*, vol. 10, pp. 1083–1092, 2016. [Online]. Available: http://ieeexplore.ieee.org/document/7516563/

VIII. Project Repository

The project repository can be found at:
GitHub.

Appendix

[11] R. J. Hyndman and G. Athanasopoulos, *Forecasting: principles and practice*. OTexts, 2018.

[12] Y. Ensafi, S. H. Amin, G. Zhang, and B. Shah, "Time-series forecasting of seasonal items sales using machine learning – a comparative analysis," *International Journal of Information Management Data Insights*, vol. 2, no. 1, p. 100058, 2022. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S2667096822000027

[13] Jason, "How to remove trends and seasonality with a difference transform in python," Available: https://machinelearningmastery.com/remove-trends-seasonality-difference-transform-python/, July 2017.

[14] Yugesh, "Quick way to find p, d and q values for arima," Available: https://analyticsindiamag.com/quick-way-to-find-p-d-and-q-values-for-arima/, May 2022.

[15] ARINDAM, "Time series for beginners with arima," Available: https://www.kaggle.com/code/freespirit08/time-series-for-beginners-with-arima/notebook, May 2017.

[16] Konstantin, "https://towardsdatascience.com/time-series-forecast-error-metrics-you-should-know-cc88b8c67f27," Available: https://towardsdatascience.com/time-series-forecast-error-metrics-you-should-know-cc88b8c67f27, October 2021.

[17] X. Lv and L. Zhang, "Feature fusion learning based on lstm and cnn networks for trend analysis of limit order books," pp. 125–137, 2021.

[18] A. Xavier, "An introduction to convlstm," Available online at: https://medium.com/neuronio/an-introduction-to-convlstm-55c9025563a7, March 2019.

[19] T. Potdar, "Mid-price prediction in a limit order book using deep learning models," Available: https://github.com/tanvipotdar/LOBMidPricePrediction [Accessed: Apr 24, 2020], April 2020.

[20] H. Wang, W. Huang, and S. Wang, "Forecasting open-high-low-close data contained in candlestick chart," *arXiv preprint arXiv:2104.00581*, 2021.

[21] Jess, "Limit order book modelling with deep learning (lstm network) for price and market movement predictions," Available: https://github.com/jessgess/deep-learning-for-order-book-price-and-movement-predictions [Accessed: May 27, 2019], March 2019.

[22] A. Ntakaris, G. Mirone, J. Kanniainen, M. Gabbouj, and A. Iosifidis, "Feature engineering for mid-price prediction with deep learning," vol. 7, pp. 82 390–82 412, 2019.

[23] Z. Zhang, S. Zohren, and S. Robert, "Deeplob: Deep convolutional neural networks for limit order books," *arXiv preprint arXiv:1909.09243*, 2019.