Czech University of Life Sciences Prague

Faculty of Economics and Management

Department of Information Technologies

Diploma Thesis

Forecasting Foreign Currency Exchange Rate Using Deep Learning

Author: Feven Mulugeta Shimelis

Supervisor: Ing. Pavel Šimek, Ph.D.

CZECH UNIVERSITY OF LIFE SCIENCES PRAGUE

Faculty of Economics and Management

DIPLOMA THESIS ASSIGNMENT

B.Sc. FEVEN SHIMELIS, BSc

Systems Engineering and Informatics
Informatics

Thesis title

Forecasting Foreign Currency Exchange Rate

Objectives of thesis

Diploma thesis is focused on problem of forecasting foreign currency exchange rate. The main objective is to design and implement foreign currency exchange rate forecasting system using deep learning techniques. Specific objectives of this work are:

- to review various literature and understand the concept of Foreign Exchange Rate Forecast and Deep Learning,
- to study the general steps which are required to design and implement the forecasting system,
- to preprocess the data through different preprocessing step which can help to increase prediction accuracy,
- to propose RNN to train, predict exchange rate using the collected historical dataset,
- to check whether RNN is a good model for foreign exchange rate forecasting by training it on different currencies,
- to evaluate the performance of the deep learning algorithm with different parameters.

Methodology

Methodology of the diploma thesis is based on study and analysis of specialized information sources. The practical part is focused on forecasting foreign currency exchange rate using deep learning. Parts of methodology are:

- Literature review,
- Data Collection and Dataset Preparation,
- Data Preprocessing,
- Training and evaluation.

Based on a synthesis of theoretical knowledge and the results of own solution, the conclusions of the thesis will be formulated.

The proposed extent of the thesis

60 – 80 pages of text.

Keywords

Foreign Exchange Rate Forecast, Artificial neural network (ANN), Recurrent Neural Network (RNN)

Recommended information sources

Digital Finance : Smart Data Analytics, Investment Innovation, and Financial Technology, ISSN 2524-6186
International Journal of Engineering and Technology (IJET), ISSN : 0975-4024
NNS Conference on Big Data and Deep Learning 2018
World of Computer Science and Information Technology Journal (WSCIT), ISSN: 2221-0741

Expected date of thesis defence

2020/21 SS – FEM

The Diploma Thesis Supervisor

Ing. Pavel Šimek, Ph.D.

Supervising department

Department of Information Technologies

Electronic approval: 29. 7. 2020

Ing. Jiří Vaněk, Ph.D.

Head of department

Electronic approval: 21. 10. 2020

Ing. Martin Pelikán, Ph.D.

Dean

Prague on 14. 03. 2021

DECLARATION

I declare that I have worked on my diploma thesis titled Forecasting Foreign Currency Exchange Rate Using Deep Learning by myself and I have used only the sources mentioned at the end of the thesis. As the author of the diploma thesis, I declare that the thesis does not break copyrights of any person.

In Prague on 31.03.2021 Feven Mulugeta Shimelis

ACKNOWLEDGEMENT

First and foremost, I would like to thank almighty God. Without his mercy, grace, help and blessings, nothing would be possible. I would also like to thank the Ever-Virgin, St. Marry, Mother of our Lord.

My deepest gratitude goes to my thesis supervisor, Ing. Pavel Šimek, Ph.D. I am thankful for his support, insights, and constructive comments in this research work. His willingness to give time so generously has been very much appreciated. I would also like to thank Czech Ministry of Sports and Education for the masters study scholarship opportunity.

I would also like to thank my family for the lifetime support and love that keeps me motivated and focused. I would particularly like to thank my mom Sirgute Mulugeta who is the role model and strength in my life. None of this would have been possible without her love and prayer.

Finally, my deepest gratitude goes to my friends who are always there to help and support me with ideas in the process of writing this thesis work.

ABSTRACT

The foreign currency exchange rate(forex) is the most traded market in the financial world. Forex forecasting can be performed by analysing past data and gather patterns. Different methods for forex forecasting have been proposed in the research world. Every forex pair has it is own characteristics, and the forecast depends on the algorithms utilized. Thus, in this study a deep learning model which predicts a forex price has been developed using collected historical dataset only. The proposed system has four components: preprocessing, feature analysis and selection, normalization, and finally the forecasting phase. In this study, 22 years of a daily real-world historical dataset of CZK/Euro and CZK/USD forex pairs are collected from investing.com financial platform. Simple RNN and LSTM based forecasting models are proposed, the models are implemented using Keras in Python and tested using 20 days dataset. Mean squared error and accuracy are used to evaluate the performance of the models, MSE was taken as a metrics and calculate the error based on actual and predicted prices. Simple RNN achieved an accuracy of 89.91% and 82.81% for CZK/Euro and CZK/USD respectively. LSTM achieved forecasting accuracy of 97.64% and 93.68% for CZK/Euro and CZK/USD respectively. As a result, LSTM outperforms the simple RNN with higher accuracy and minimum mean squared error.

Keywords: Foreign currency, Exchange rate, Forecasting, Artificial Neural Network (ANN),Recurrent Neural Network (RNN), (LSTM) Long Term Shot Term Memory, (CZK) Czech Koruna, Preprocessing, Technical indicator, Deep Learning.

ABSTRAKTNÍ

Směnný kurz cizí měny (forex) je nejvíce obchodovaným trhem ve finančním světě. Forexové předpovědi lze provádět analýzou minulých dat a shromažďováním vzorů. Ve světě výzkumu byly navrženy různé metody předpovídání forexu. Každý forexový pár má své vlastní charakteristiky a předpověď závisí na použitých algoritmech. V této studii byl tedy vyvinut model hlubokého učení, který předpovídá cenu forexu pouze pomocí shromážděných historických datových sad. Navrhovaný systém má čtyři komponenty: předzpracování, analýzu a výběr funkcí, normalizaci a nakonec fázi předpovědi. V této studii je z finanční platformy investing.com shromážděno 22 let denních historických datových sad reálného světa v měně CZK / Euro a CZK / USD. Jsou navrženy jednoduché prognostické modely založené na RNN a LSTM, modely jsou implementovány pomocí Keras v Pythonu a testovány pomocí 20denní datové sady. K vyhodnocení výkonu modelů se používá střední kvadratická chyba a přesnost, MSE byla brána jako metrika a chyba byla vypočtena na základě skutečných a předpokládaných cen. Simple RNN dosáhly přesnosti 89,91% a 82,81% pro CZK / Euro, respektive CZK / USD. LSTM dosáhla přesnosti předpovědi 97,64%, respektive 93,68% pro CZK / Euro a CZK / USD. Výsledkem je, že LSTM překonává jednoduché RNN s vyšší přesností a minimální střední kvadratickou chybou.

Klíčová slova: Cizí měna, směnný kurz, předpovědi, umělá neurální síť (ANN), rekurentní neurální síť (RNN), (LSTM) dlouhodobá zastřelená termínová paměť, (CZK) česká koruna, předběžné zpracování, technický ukazatel, hluboké učení.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS

AF	Acceleration Factor
ANN	Artificial Neural network
CZK	Czech koruna
USD	U.S. Dollar
MA	Moving Average
CPI	Consumer Price Index
RSI	Relative Strength Index
MAE	Mean Absolute Error
MLP	Multi-Layer Perceptron
MSE	Mean Squared Error
GRU	Gated Recurrent Unit
CCI	Commodity Chanel Index
PSAR	Parabolic Stop and Reverse
RNN	Recurrent Neural Network
PPL	Purchasing Power Parity
EMA	Exponential Moving Average
LSTM	Long Short Term Memory
ARMA	Auto-Regressive Moving Average
MACD	Moving Average Convergence Divergence

1. Introduction

From the start of bartering to the present day, money has played an essential role in human culture for nearly 3,000 years. Metal artifacts may have been used as money as early as 5,000 B.C., according to historians. The Lydians were the first Western civilization to produce coins around 700 B.C. Other countries and cultures quickly followed suit, minting their coins of specific denominations. King Alyattes of Lydia developed the first known currency. In 600 B.C., which is now part of Turkey, A roaring lion appears on the first coin ever minted. Coins were replaced by banknotes around 1661 AD. In 1946, the first credit card was issued. There are different types of money, such as Commodity money, receipt money, Fractional money, and Fiat money.Currency is any coin, note, piece, or another item that is presented as currency. Paper notes, coins, and plastic cards are the most popular forms of currency today (e.g., credit or debit cards). Bartering is a type of direct trading in which goods and services are exchanged directly. Over the centuries, a form of currency arose based on easily traded items such as animal skins, salt, and weapons. These exchanged goods were used as a form of currency. This trading system spread across the world, and it is still in use in some areas today.

The exchange rate is an important macroeconomic policy. A foreign exchange market is a market where a convertible currency is exchanged for another convertible cash or other convertible currencies[1]. The foreign exchange market is significant in a country's economic growth Since it performs many useful functions such as serving as a focal point for setting prices in various currencies, supporting the investment functions of banks and corporate traders who can take currency risks on behalf of their companies and helps investors to hedge or reduce the risk of losing money due to exchange rate fluctuations.

Forecasting is critical and necessary since we live, work, and make decisions in an unpredictable environment. One of the forecasting's functions is to turn uncertainty into risk. Uncertainty exists when we do not know what will happen in the future, while risk exists when we have a probability distribution for the outcome, and we act based on it. Forecasting is helpful because it reduces uncertainty and leads to a better decision.

Economists and investors forecast future exchange rates to calculate the monetary value based on the forecasts. There are many models used to predict a currency's future exchange rate discussed briefly in chapter 3. However, almost all forex forecasting models are riddled with uncertainties, and none of them can claim to be 100 percent accurate in predicting the exact future exchange rate.

Time series analysis is a popular statistical tool for predicting and is commonly used in many statistical and economic applications. The dependent variable's behavior is predicted based on its past behavior. On the other hand, a modern method is more accurate and effective in forecasting that can use logic in their operations rather than the fixed relationship between variables known as artificial neural networks. The use of neural networks to forecast exchange rates has proved as a successful technique in forecasting. Exchange rates are characterized as non-linear, stochastic, and highly non-stationary financial time series [2].Due to this nature and forex data, artificial neural networks have proved to be a more powerful way to reveal systemic relationships among the different entities. Deep-learning-based forecasting models and their finance applications have attracted considerable attention in recent research [3]RNN is also a type of deep learning and can be applied for time-series forecasting.

LSTM is a type of RNN that overcomes the problem of vanishing gradient descent by having long memory.[4]Performed forex forecasting by comparing traditional recurrent network architectures, LSTM, and gated recurrent units (GRU) and concluded that LSTM and GRU showed promising performance in predicting forex.

Therefore, this research aimed to design and develop a foreign currency exchange rate forecasting model using technical analysis by applying the historical forex data and perform future predictions. This study does not consider fundamental factors for building the forecasting model. The model is evaluated using the historical forex dataset for CZK against Euro and USD currency pairs.

2. Objectives and methodology

2.1. Objectives

General Objectives

The general objective of this study is to design and implement a foreign currency exchange rate forecasting system using deep learning techniques.

Specific objectives

To meet the general objective, the following specific objectives are formulated:

- To review various literature and understand the concept of foriegn currency forecast and deep learning.
- To study the general steps which are required to design and implement the forecasting system.
- To preprocess the data through a different preprocessing step which can help to increase prediction accuracy.
- To propose RNN to train, predict exchange rate using the collected historical dataset.
- To check whether RNN is a good model for foreign exchange rate forecasting by training it on different currencies.
- To evaluate the performance of the deep learning algorithm with different parameters.

2.2. Methodology

To achieve the objectives of the study, the following methods and techniques are applied in the process of conducting the study.

Literature review

Various materials, such as journals, books, articles, research papers, and on-going conference papers on the subject of forging exchange rate prediction and related fields, are examined to select an efficient forecasting algorithm. Possible solution finding methods also are studied.Basic concept of Foreign currency exchange rate forecasting and state-of-the-art deep learning approaches used for analyzing historical exchange rate data are examined for a better understanding of the study. Different approaches and techniques of foreign currency exchange rate forecasting including ppl,econometric model,time series and deep learning models are presented. Detailed description of ANN,RNN and LSTM are discussed thoroughly. Detailed analysis of different works related to the the problem domain , exchange rate forecasting using different approaches is perfomed.

Data Collection and Dataset Preparation

Dataset collection and preparation is the first task of the exchange rate forecasting system. Historical dataset for Czech Koruna (CZK) against Euro and USD is collected from investing.com website, the most known global financial website. As we are dealing with forex price, the collected dataset has numerical values only. This dataset is used for training and testing the algorithm.

Data Preprocessing

Preprocessing is the essential step in forecasting. The aim of preprocessing is to remove noise and to make the subsequent step simple. The preprocessing step includes noise removal, calculating technical indicators such as moving average and relative strength index for the dataset exchange rate variable. These calculated indicators are integrated with the collected dataset as an additional input. We have also applied the data normalization technique to our dataset. The new dataset is scaled using the appropriate method in a fixed range to feed the model with normalized data. For this thesis, work normalization is used to scale the collected dataset in the field of [0,1].

Feature Analysis and Selection

After the preprocessing, the next step is feature analysis and Selection. In this research RandomForestRegressor algorithm is used to select the essential features from the listed parameters. After this step, we have classified our preprocessed dataset into training, testing, and validation datasets accordingly.

Training and evaluation

After designing the RNN, the network is trained with the training and validation dataset. The performance of the proposed model is evaluated on the test dataset using mean squares error and accuracy.

3. Literature Review

3.1. Foreign currency Exchange rate market (Forex)

FOREX is derived from Foreign Exchange and is the largest financial market globally that deals with approximately $1.8 trillion transactions every day[5]. The foreign exchange market is a financial market where currencies are bought and sold simultaneously[6]. It is the most important economic indicator in the international monetary market. It consists of multiple international participants, including professionals and individuals who invest and speculate for profit due to its robust liquidity[7]. The market is fundamentally classified as a liquid market where the information is public and accessible for all traders equally. International banks, central banks, commercial companies, hedge funds, retail forex brokers, and investors are critical participants in this vast market. Investors can make high profits at minimum cost in the market of frequent changes in the exchange rate. Therefore, the prediction of time series data for exchange rate has become a hot issue in financial market research[8] An estimated $1 trillion is traded every day in the foreign exchange market, making it the largest and most liquid of the financial market. Foreign exchange rates are among the world's most significant economic indicators in the monetary markets[9]

3.2. Exchange Rate Regime

The exchange rate regime is the way a nation manages its currency in the enormous foreign exchange market. The country's monetary policy is closely related to the regime. Floating exchange, fixed exchange, and pegged float are the basic exchange rate regimes.

3.2.1. Floating Exchange Rate

A floating exchange rate is one in which a currency's value is permitted to fluctuate under the foreign exchange market. A floating currency is a currency that uses a floating exchange rate. The euro and dollar are an example of a floating currency. According to many economists, floating exchange rates are the best possible exchange rate system because they automatically adapt to economic conditions. These regimes allow a country to mitigate the effects of shocks and international business cycles and avoid a balance of payments crisis. As a consequence of their dynamism, they often generate unpredictability.

3.2.2. Pegged Float Exchange Rate

Pegged floating currencies are tied to a band or value that is either set or modified regularly. These regimes are a mix of fixed and floating.

3.2.3. Fixed Exchange Rate

A fixed exchange rate mechanism is where governments attempt to keep their currency constant against a specific currency. In this scheme, a country's government determines the value of its currency by comparing it to a fixed weight of an asset, another currency, or a basket of currencies. A country's central bank remains committed to buying and selling at a fixed price at all times.

The central bank of a nation maintains foreign currency and gold reserves to ensure that a currency's "pegged" value is maintained. They will sell these reserves to intervene in the foreign exchange market to meet excess demand for the country's currency or to absorb excess supply. The gold standard is the most well-known fixed-rate scheme, a scheme in which each unit of currency is bound to a particular amount of gold. Many currencies are also pegged by regimes. These countries have the option of pegging to a single currency or a "basket" of currencies representing the country's main trading partners.

3.3. Czech Republic forex market

In the sense of liberalized capital movements, the Czech Republic has mixed inflation targeting policy with a floating exchange rate system since 1998. To prevent unnecessary koruna volatility, foreign exchange interventions have been made on occasion. The monetary authorities outlined their preparations for the financial integration process in the second half of 2003. The CNB specifically proposed that koruna remains outside the ERM II for a period after entering into the E.U. in 2004, owing to the fiscal consolidation calendar and the need for structural reforms.

The government approved a proposal to convert to the euro in April 2007. Although 2012 was seen as a reasonable date for the transition, the Prime Minister did not set a specific timetable.

Czech Republic is still a candidate for eurozone membership. The Czech Republic's exchange rate was pegged until early 1996, when it was effectively abolished by a significant expansion of the fluctuation band. The Czech economy now operates in a so-called managed floating system, in which the exchange rate is floating. Still, the central bank may intervene if severe fluctuations occur. As compared to the currencies of other major European countries, Central European currencies, like the Czech koruna, have relatively low liquidity and high volatility.

3.4. Foreign Exchange Rate Forecasting Approaches

There are many directions in which research on the foreign currency exchange rate has been carried out during the past years. Fundamental and technical forecasting methodologies are widely applicable in use in financial forecasting[10]. And this methods can be used for Forex as well. In the subsections below, we discussed these two significant methodologies that have been emerged as a result of different researches.

3.4.1. Fundamental Approach

Since the market is open foreign exchange rates are affected by many highly correlated economic, political, social, and even psychological factors. The interaction of these factors is exceptionally complicated.Therefore, forecasting the changes in foreign exchange rates is generally very diffcult[10]

The fundamental approach focuses on analysing the affecting factors to forecast the future price or direction. Most of the researchers who use the fundamental approach focus on studying those factors to predict the future forex price. Primary factors which can affect of FOREX market of a country are listed below.

1. Inflation
2. Rate of interest
3. Capital account balance
4. Role of speculators
5. Cost of manufacture
6. Debt of the country

7. Gross domestic product
8. Political stability and economic performance
9. Employment data
10. Relative strength of other currencies
11. Macroeconomic and geopolitical events

Figure 3. 1 Currency price influencing factors [11]

3.4.2. Technical Approach

Technical analysis examines past market data by identifying and making a chart of patterns produced by the forex signals to forecast future changes in forex trend lines. Technical Indicators such as positioning surveys, moving-average trend-seeking trade rules, and Forex dealers' customer-flow data are used in this approach. As a type of non-stationary time-series data, financial trading data is highly volatile and complex. Technical analysis can smooth out noise and help identify trends and become more prevalent in trading research[10].

Technical analysts believe that price fluctuations are not random and are not unpredictable by nature. Once a particular type of trend is established, it is likely to continue for a certain period. Technical Indicators are spicy of technical analysis. Now let discuss some basic concepts on technical indicators used in the technical analysis approach.

3.4.2.1. Technical Analysis Indicators

A technical indicator is a time series obtained from mathematical formula(s) applied to another time series, which is typically a price[10]. These indicators are calculated by utilizing the close, open, high, low, and volume data. The indicators can be used on stocks and commodities since they are also traded in an open market.

They are empirical assistants widely used to identify future price trends and measure volatility[10]. The technical indicators help us to analyze the pattern of the past historical data makes perform future predictions. According to their functionalities, the indicators can be grouped into four categories: trend, momentum, volatility, and volume. The forex dataset used in this research work doesn't have the volume of the exchange rate. That being said, let us discuss each of the first three indicators in detail.

3.4.2.1.1. Trend Indicators

Because of signals in forex data often conflict, it is difficult to separate the price pattern from the background noise. As with a moving average, price data is smoothed, and a single line represents the trend. The smoothing method causes the indicators to lag in price shifts, which is why they are sometimes referred to as trend tracking indicators. Lagging indicators also referred to as trend indicators, follow the past price action [6]. Trend indicators help to measure the direction of the trend. The most known types of trend indicators are listed below.

Moving Average

By smoothing price data, moving averages provide a way to measure of trend direction. The moving average is typically calculated using closing prices, but it can also be computed using median, normal, weighted closing, open, high, and low prices. The trend of M.A. only changes when there is a significant change in price[12].

Moving average can be calculated for various time intervals. For longer cycles, 100 to 200 Day moving averages are primarily used, for intermediate cycles 20 to 65 Days and 5 to 20 Days for short cycles. Since our data set holds short-term cycle 7- and 14-days moving average is calculated in this research. Moving average is calculated by adding close prices for the chosen period and dividing it by the day.

$$X \text{ days MA} = (A1+A2+\ldots\ldots+Ax) / X \quad \text{Where A is close price} \tag{3.1}$$

In this study, three moving average types are calculated, which are simple moving average that takes into account only the last n observations. In contrast, cumulative moving average considers all priority observations, and exponential moving average gives the most recent data points more weight and meaning.

Moving Average Convergence/Divergence (MACD)

Moving Average Convergence Divergence (MACD) is one of the most widely used type of trend indicator. Gerald Appel presented the (MACD) trading method in his book The Moving Average Convergence Divergence Trading Method. In MACD, the distance between the two moving average lines is measured.[13] stated MACD as below:

$$EMA12 = 11/13 \times EMA12 \text{of last day} + 2/13 \times \text{the closing price of the day} \tag{3.2}$$

$$EMA26 = 25/27 \times EMA26 \text{of last day} + 2/27 \times \text{the closing price of the day}$$

$$DIF = EMA12 - 26$$

$$DEA = 8/10 \times DEA \text{ of last day} + 2/10 \times DIF$$

$$MACD = (DIF - DEA) \times 2$$

Parabolic stop and reverse (Parabolic SAR)

Parabolic stop and reverse (Parabolic SAR) are another type of trend indicator, J. Welles Wilder Jr. invented the parabolic SAR, which is presented in his book New Concepts in Technical Trading Systems. Parabolic SAR has convenient entry and exit points. By utilizing the previous day's data, a stop loss is calculated for each day. The position is long if the stop level is below the current price (Rising parabolic SAR), and the position is short if the stop level is above the current price (Failing parabolic SAR).

$$\text{Rising PSAR} = \text{Prior PSAR} + [\text{Prior AF (Prior EP – Prior PSAR)}] \tag{3.3}$$

$$\text{Falling PSAR} = \text{Prior PSAR} - [\text{Prior AF(Prior EP– Prior PSAR)}] \tag{3.4}$$

Where:

• AF = Acceleration Factor. It starts at 0.02 and increases by 0.02, up to a maximum of 0.2, each time the extreme point makes a new low (falling SAR) or high (rising SAR) ;
• EP = Extreme Point. The lowest low in the current downtrend (falling SAR) or the highest high in the current uptrend (rising SAR) ;

3.4.2.1.2. Momentum indicators

Momentum indicators measure the velocity and magnitude of price movement, regardless of whether the price is moving up or down, and signal if an instrument is being overbought or oversold. These indicators are helpful because they assist traders and analysts in identifying market reversal points. Three types of momentum indicators are discussed below.

Commodity Chanel index

The Commodity Channel Index (CCI) was introduced by Donald Lambert and presented in his book Commodities Channel Index: Tools for Trading Cyclic Trends. This indicator helps for determining where a price is in relation to its moving average. This can be used to indicate whether the market is overbought or oversold or when a pattern is fading. CCI similar to Bollinger Bands in nature, but instead of overbought/oversold levels, it is shown as an indicator line. The Commodity Channel Index calculation is fairly complicated. The formula for calculating N day CCI is presented below.

- TP= (High + Low + Close) / 3 (3.5)
- Find TPMA= n days simple M.A. for Typical Price(T.P.)
- Subtract TPMA from T.P. and divide the result by the calculations below
- Find the result of T.P. - today's TPMA from T.P.
- Sum up the absolute values and divide it by N.
- Multiply the result by 0.015.

Relative Strength Index

The Relative strength index (RSI) indicator is one of the types of momentum indicators. RSI compares closing prices of the current and previous candles for the up and down trends and then turns the outcome into EMA and afterward calculates how the uptrend EMA relates to the downtrend EMA. The formula for calculating RSI is presented below.

RelativeStrength = Avg. 14-Day Up Closes/Avg. 14-Day Down Closes (3.6)

RSI = 100 - (100/(1+Relative Strength)

Where:

- Avg. 14-Day Up Closes is 14 day periods average up trend for the close price;
- Avg. 14-Day Down Closes is 14 day periods average down trend for the close price;

Stochastic Oscillator

Stochastic Oscillators are momentum indicators developed by George Lane in the late 1950s. This indicator compares a closing price to a range of its prices over a given period.

According to [14][12]Stochastic Oscillator is calculated as follows.

$\%K = (C14 -- L14/H14- L14) \times 100,$ (3.7)

Where:

- C is the latest closing price;
- L14 is the lowest price in the past 14 days;
- H14 is the highest price in the past 14 days
- %K is the slow stochastic indicator.

3.5.2.1.3. Volatility

Volatility indicators determine how far an asset deviates from its mean directional value. As per [15]three types of volatility indicators are reported. These indicators are listed and explained below.

Bollinger bands

Bollinger Bands are plotted at a standard deviation level above and below a simple moving average. Bollinger Bands (B.B.s) can be applied in all financial markets and used in most time frames[16]. The formula of Bollinger Bands[17] is presented below, 20 days period is used:

$$\text{Upper band} = \text{MA} + 2 \times \text{Standard deviation} \quad (3.8)$$
$$\text{Middle band} = \text{M.A.}$$
$$\text{Lower band} = \text{MA} - 2 \times \text{Standard deviation},$$

Where: standard deviation = $\sqrt{\sum n\ j=1 (Xnj - MA)2}$, n=20

Average true range

Average true range (ATR) is introduced by J. Welles Wilder, Jr. ATR is the mean of actual ranges over a given time and shows the price volatility.

$$TRi = \max[(\text{high} - \text{low}), \text{abs}(\text{high} - \text{previous close}), \text{abs}(\text{low} - \text{previous close})] \quad (3.9)$$
$$\text{ATR (first)} = 1/N \sum t\ i=1\ TRi$$
$$\text{ATR} = Previous\ ATR \times \text{t-1/t} + \text{TR / t}$$

Where: t= time

Sample standard deviation

The sample standard deviation (SSD) is used to quantify the current price's dispersion from its mean. The standard deviation used N, whereas SSD uses N − 1.

$$\text{SDD} = \sqrt{\sum n\ j=1 (nX - j - 1MA)2}, \text{ n=20} \quad (3.10)$$

In this research work, a technical approach is applied by studying historical exchange rate data and the computed technical indicators. The technical indicators used in this study are described in chapter 3. we built a neural network model for the prediction, by utilizing the technical indicators and the initially recorded data.

3.5. Models of Forex forecasting

A currency exchange rate forecast can help brokers and companies make informed decisions to reduce risks and increase profits. There are different methods of forecasting currency exchange rates. Here, we will look at a few of the most popular methods: purchasing power parity, relative economic strength, econometric models, and time series models.

3.5.1. Purchasing Power Parity Model

The Purchasing Power Parity forecasting approach stated that the same goods in different countries should have the same prices taking into account the exchange rate and keeping out transaction and shipping costs. As per PPP, a pen in the Czech Republic should be the same as a pen in Ethiopia. This approach predicts that the exchange rate will adjust by balancing the price changes occurring due to inflation.

3.5.2. Relative Economic Strength Model

The relative economic strength model reveals if a currency is going to appreciate or depreciate, it does not forecast the future exchange rate. The approach is more of classification, not regression. In this approach, the interest rate is a fundamental factor since a high rate will attract more investors. The currency's power will increase, which would result in the country's currency appreciation.

3.5.3. Econometric Models

In the econometric model, all necessary factors that affect a currency are gathered and used to forecast the future price direction. The factors are initially from economic theory. However, any necessary variable can be added as well if it is required. GDP and interest are the know economic factors. In this approach, the trader can predict the future price movement by connecting these factors and building a suitable econometric model for the prediction.

3.5.4. Time Series Prediction Model

The time series model is entirely technical, with no economic theory or element to consider in the forecasting process. The most popular time-series approach is the autoregressive moving average (ARMA).

The underlying principle is that the past behaviour and patterns in the price can affect the future price movement and behaviour. Selected parameters from the historical data are used to create a prediction model in this approach. Since time series prediction problems are one of the most challenging types of predictive modelling problems, a time series forecasting model's ability to predict the future is determined by its results.[17] indicated that traditional statistical techniques for forecasting had reached their limitation in applications with nonlinearities in the data set, such as stock indices[17]

3.6. Deep Learning Techniques

Neural networks are a computational model with certain parallels to the human brain, in which a large number of simple units operate in parallel with no centralized control unit. The weights between the units are the primary means of storing information in neural networks over the long term. The feedforward multilayer neural network is the most well-known and easiest for understanding the neural network. It has an input layer, a single output layer, and one or more hidden layers. The facets that differentiate deep learning networks in general from feed-forward multilayer networks are as follows:

- Automatic feature extraction
- More neurons than previous networks
- More complex ways of connecting layers

Neural networks are adaptable and can model nonlinearly. There is no need to define a model while using ANN. Instead, the model is dynamically adjusted based on the data's characteristics. Neural networks like Long Short-Term Memory (LSTM) recurrent neural networks can seamlessly model problems with multiple input variables. This is a great benefit in time series forecasting, where classical linear methods can be challenging to adapt to multivariate or multiple input forecasting problems[18].

In recent years, deep learning tools, such as long short-term memory (LSTM), have become popular and effective for many time-series forecasting problems[19]. To further provide color to the definition of deep learning, we define and discuss the significant architectures of deep neural networks: Recurrent Neural Networks (RNN) and extension of RNN, namely LSTM, which we applied in this research work.

3.6.1. Recurrent Neural Network

Recurrent neural networks are robust and commonly used neural network architecture class built to model sequence data[20]. RNNs have a similar structure in comparison with the feed-forward neural networks. With a slight contrast, RNNs have a recurrent hidden state whose activation at present is dependent on that of the previous time. The ability of recurrent neural networks with temporal dependency makes them fit for the tasks when input and output consist of a sequence of dependent points. In feed-forward networks, inputs are fed to the network, and an output is produced. Feedforward networks can be used for classification purposes.

A common assumption of a neural network is that all units of the input vectors are independent of each other. Therefore, the traditional neural network cannot use sequential knowledge. The RNN model, by comparison, introduces a hidden state produced by sequential information from a time series, with the output depends on the hidden state.

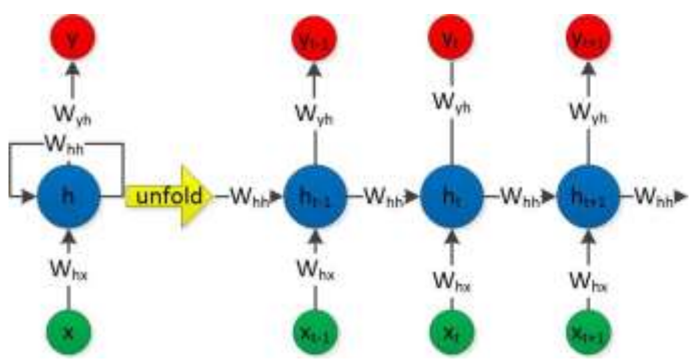

Figure 3. 2 Recurrent Neural network

[21]shows the above architecture of RNN.

Where:

- x: is input
- h: hidden state
- y: output
- w: weight

RNN has been effectively employed to several problems, such as natural language processing, speech recognition, generation of image descriptions, and machine translation[22]. The basic idea behind RNN models is that to teach new elements in the sequence to contribute some further information, which updates the current state of the model.

RNN has been widely used in time series modeling [23]. The prediction result of the recurrent neural network makes now somewhat depends on the prediction it got in the previous moment. Hence, the present output of the network depends on the previous output as well as the current inputs[23].

Input layer: the input is the row vector input, and the input to neurons of the other layers is the output (activation) of the previous layer's neurons[23]. The number of neurons in an input layer is typically the same as the network's input feature. Input layers are followed by one or more hidden layers[24].

Hidden layer: There are one or more hidden layers in a feed-back neural network. The weight values on the layers' connections are how neural networks encode the learned information extracted from the raw training data. Hidden layers are the key to allowing neural networks to model non-linear functions.

Output layer: output (prediction or classification) of our model is answered from the output layer. The output layer gives us an output based on the input from the input layer [23]. Depending on the setup of the neural network, the final output may be a real-valued output (regression) or a set of probabilities (classification). This is controlled by the type of activation function we use on the neurons in the output layer[25].

Connections between layers: In a fully connected feed-back network, the connections between layers are the outgoing connections from all neurons in the previous layer to all of the next layer's neurons. These weights are progressively changed as the algorithm finds the best solution with the backpropagation learning algorithm [26].

Each neuron receives an array of inputs and produces a single output. The output of a neuron in the input layer will be input for the neuron in the hidden layer. Similarly, the output of the neuron in the hidden layer will be input for the output layer. Each neuron in all the layers processes its input by a mathematical function known as the neuron activation function (transfer function). The neurons in the input layer connect with the neuron in the hidden layer, while the neuron in the output layer is only connected to the neuron in the hidden layer. There is no direct connection between the neuron in the input layer and the neurons in the output layer. In feed-forward networks, the signal flow is from input units to output units, directly in a feed-forward direction. The data processing can extend over multiple units, but no feedback connections [27]. The popular way to train RNN is backpropagation through time. This makes the problem of the vanishing gradients, which causes the parameters to capture short-term dependencies while the information from earlier time steps decays.While RNN performs sequence modeling, the long-term dependence within the sequence is difficult to learn due to the vanishing gradients[28].LSTM, initially proposed by[29], is an effective solution for tackling vanishing gradients using memory cells.

3.6.2. LSTM

Long Short Term Memory networks (LSTM) are a special kind of RNN explicitly designed to avoid the long-term dependency problem[30]. LSTM is capable of learning long-term dependencies and it has feedback connections. It can also process the entire sequence of the data. LSTM units consist of memory blocks and memory cells, along with the gate units they contain. The LSTM system consists of an input gate, forget gate, and an output gate. By using the gates to selectively retain information that is relevant and forget information which is not relevant.

LSTM overcome the problem of vanishing gradient descent. LSTM network is Lower sensitivity to the time gap which makes it better for analysis of sequential data than simple RNNs. LSTMs hold information separately from the normal flow of the RNN in a gated cell. Information are often kept in, read, or written from a cell, like data during a computer’s memory. The cell makes judgments via gates that open and close about, when to allow reads and writes, what to store, and removals. Yet, these gates are analog, applied with element-wise duplication by sigmoid, unlike the digital storage on computers, which are all in the range of 0-1. LSTMs excel in predicting, classifying,learning, and processing sequential data. It can be applied on video analysis, caption generation, weather forecasting, stock market prediction[29].

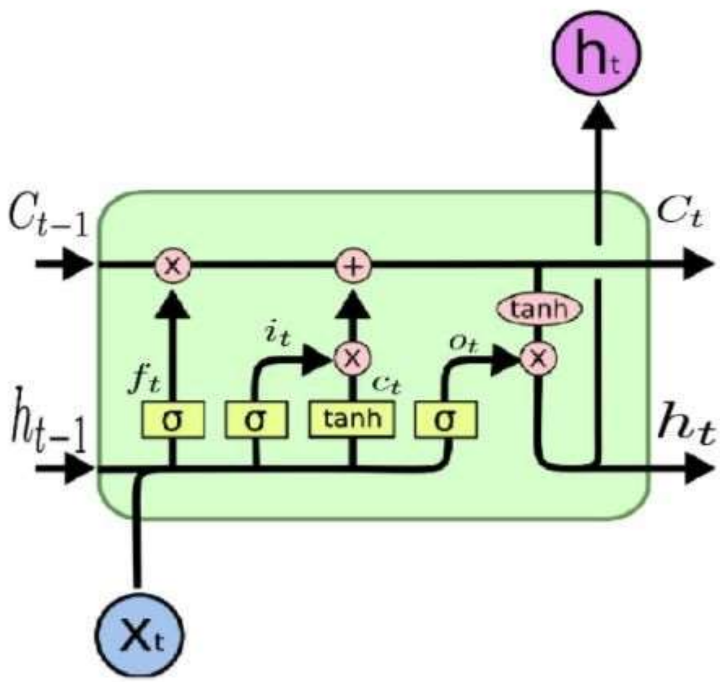

Figure 3. 3 LSTM Architecture by [29]

[31]The LSTM calculates the hidden states as follows :

$$i_t = \sigma(W_f[h_{t-1}, x_t] + b_i) \quad (3.11)$$

$$f_t = \sigma(W_f[h_{t--1}, x_t] + b_f) \quad (3.12)$$

$$\text{Sigmoid}== \frac{1}{1+e^{-1}} \quad (3.13)$$

$$o_t = \sigma(W_o[h_{t-1}, x_t] + b_o) \quad (3.14)$$

$$\bar{c}t = tanh(W_c[h_{t-1}, x_t] + bc) \quad (3.15)$$

$$c_t = f_t * c_{t-1} + i_t * \bar{c}t \quad (3.16)$$

$$h_t = o_t * tanh(c_t) \quad (3.17)$$

Where:

- σ is the logistic sigmoid function;
- *i*, *f*, and *o* are the input, forget and output gate respectively;
- h is a hidden vector that is the same size in each layer;
- W is a weight matrix for the transformation of information from cell to gate vector.

LSTM Recurrent Neural Networks have proven their capability to outperform in the time series prediction problems. When it comes to learn from the previous patterns and predict the next pattern in the sequence, LSTM models are best in this task. The analysis of the future as well as the past of a given point in the series is useful for several sequence processing tasks[28].

3.7. Time series models and Forex forecasting

[32]Gaussian Mixture Model Initialized Neuro-Fuzzy (GMMINF) is proposed for foreign currency exchange rate prediction. The proposed approach's offline training process consists of two stages considering three currency exchange pairs, namely AUD/USD, EUR/USD, and GBP/USD. The historical currency exchange price is obtained from HistData.com from 02 January 2002 to 30 December 2016. The performance of the model is evaluated by Root Mean Squared Error (RMSE), Mean Absolute Percentage Error (MAPE), and Non-Dimensional Error Index (DEI). They compared the proposed model using two versions of the autonomous learning multi-model (ALMMo) predictor, namely, ALMMoEvolving and ALMMo Offline. The proposed GMMINF predictor can provide outstanding prediction results after the offline training without any modification on the model itself.,overall results proved that the combination of GMM data space partitioning and ALMMo neuro-fuzzy forecasting system could be explored further in forecasting FOREX rates. And they stated that in the future, they would extend their work by applying the proposed approach to more different problems and further investigate its performance and real-time feature of the proposed model can be another solution for the future.

3.8. Artificial Neural Network and Forex Forecasting

[33]Proposed Artificial neural network-based foreign currency exchange rate forecasting model. The authors presented a comparison between 5 different performances for predicting four foreign currency P.S., USD, EURO, and JYEN exchange rates against the Indian rupee. The system states 1205 daily base historical data from the bank, of which 820 as input and 205 for evaluating the proposed model. The authors evaluated the prediction accuracy with the MSE(mean squared error) for each algorithm in different epochs of the four currencies. The Prediction Performance was measured by three statistical Performance Metrics Root Mean Square Error (RMSE), Mean Absolute Error (MAE), and Mean Absolute Percentage Error (MAPE). They concluded that neural network algorithms performed closer prediction better than other models for all currencies with smaller MRE and faster convergence. However, they tested the model with a few parameters, which decrease the accuracy of the prediction. And the proposed method consumed more time in the process and high memory usage.

[34]Presented Forex exchange rate forecasting using deep recurrent neural networks. The authors performed a comparison between long short-term memory networks (LSTM) and gated recurrent units (GRU) to traditional recurrent network architectures by considering four exchange rates against the U.S. Dollar. The data set consists of 12,710 daily exchange rates. The authors deemed logarithmic loss (Log loss), predictive accuracy (ACC), and the area under the receiver operator characteristic curve (AUC) as a measure of accuracy. They concluded that LSTM and GRU have better accuracy in the prediction. The authors acknowledged that different ways to set up the forecasting tasks could construct a more advanced network design.

[35]presented a Neuro-evolutionary algorithm based on Cartesian Genetic Programming evolved Artificial Neural Network (CGPANN) to predict foreign currency exchange rate. The model used 500 days of historical data sets for the U.S. dollar and the network's performance evaluated on 1000 days for five different currencies again Australian dollar. They presented a comparison between 8 recurrent neural network based models. The authors concluded that RCGPANN outperformed with an average accuracy of 98.872% and MAPE value of 1.1280 %. The proposed model achieved better forecasting accuracy compared with other research works based on neural

networks. The authors suggested that a neural network model that implements feature selection is a promising candidate for forex prediction.

[36]Designed foreign currency exchange rate forecasting model using artificial neural network (AFERFM). The proposed method performance evaluated or USA dollar, European Currency (EURO), Great Britain Pound (G.B.), and Japanese Currency (Yen) against Nigerian Money (Naira) and compared with the existing Hidden Markov model HFERFM. 800 Daily average datasets were collected from the Oanda website as input in the backpropagation to evaluate and forecast foreign exchange rates and predict the subsequent 100 daily rates. The proposed system was tested using mean square error and standard deviation AFERFM outperformed with an accuracy of 81.2 compared with 69.9% accuracy of HFERFM . The authors implied that improvement of the work could be made by extending the work to other country foreign exchange rates and by developing another suitable artificial neural network model.

Table 3. 1 Summary Of Related Works. Source [own]

Forex Forecasting Related Works Summary			
Authors	**Proposed Algorithm**	**Currency Used**	**Conclusion**
[36]	Artificial neural network (AFERFM)	USD,EURO,Great Britian japanese Yen against Nigerian Naira	Extending the work to other country foreign exchange rates and developing another suitable artificial neural network model
[32]	Gaussian Mixture Model Initialized Neuro-Fuzzy (GMMINF)	AUD/USD,EUR/USD,and GBP/USD.	Applying the proposed approach to more different problem and further investigate its performance

[33]	Artificial neural network	P.S., USD, EURO, and JYEN exchange rates against the Indian rupee.	They tested the model with a few parameters, which decrease the accuracy of the prediction. Moreover, the proposed method consumed more time in the process and high memory usage.
[34]	Traditional recurrent network architectures, LSTM, and gated Recurrent units (GRU)	Four exchange rates against U.S. Dollar.	The authors acknowledged that different ways to set up the forecasting tasks could construct a more advanced network design.
[35]	Neuro-evolutionary algorithm based on Cartesian Genetic Programming evolved Artificial Neural Network (CGPANN)	U.S. Dollar against the Australian dollar.	The authors suggested that a neural network model that implements feature selection is a promising candidate for forex prediction.

4. Practical Part

4.1. Implementation Tools

This research work is implemented on Spyder IDE which is a package under Anaconda distribution. Python is used as a programming language for the implementation with utilization of keras, tensor flow and NumPy.

4.1.1. Programming Language

Python

Python[1] widely used general-purpose programming language. Wel known for its simplicity and code readability, It allows the programmer to write ideas in fewer lines of code using predefined libraries.

4.1.2. Development Tools

Anaconda

Anaconda is a tool that is used to develop machine learning, deep learning, and artificial intelligence models. It is a Python and R- programming distribution used for data analysis and scientific computing. It has composed of many packages including Spyder and Jupiter notebook.

Spyder

Spyder is a free scientific python integrated development environment, that is included with Anaconda. It provides editing, testing, debugging, and introspection features. It also has the functionalities of a comprehensive development tool with the data exploration, interactive execution, deep inspection, and good visualization capabilities of a scientific package and model results.

[1]https://www.python.org/

Tensorflow

Tensorflow is an open-source software library developed by Google in 2015 that allows developers to design, build, and train deep learning models. TensorFlow provides various degree of concepts to select the the sutiable one to build and deploy machine learning models.

Keras

To design and test deep learning models, Keras is a solid and easy-to-use free Python open-source library. It wraps up the Theano and TensorFlow powerful numerical computation libraries and allows you to define and train models of neural networks in just a few lines of code.

NumPy

NumPy is a python programming language kit for scientific computing that provides robust data structure .This library gives a good performance multidimensional array object and tools for working with these arrays.

4.2. Overview of the Proposed System

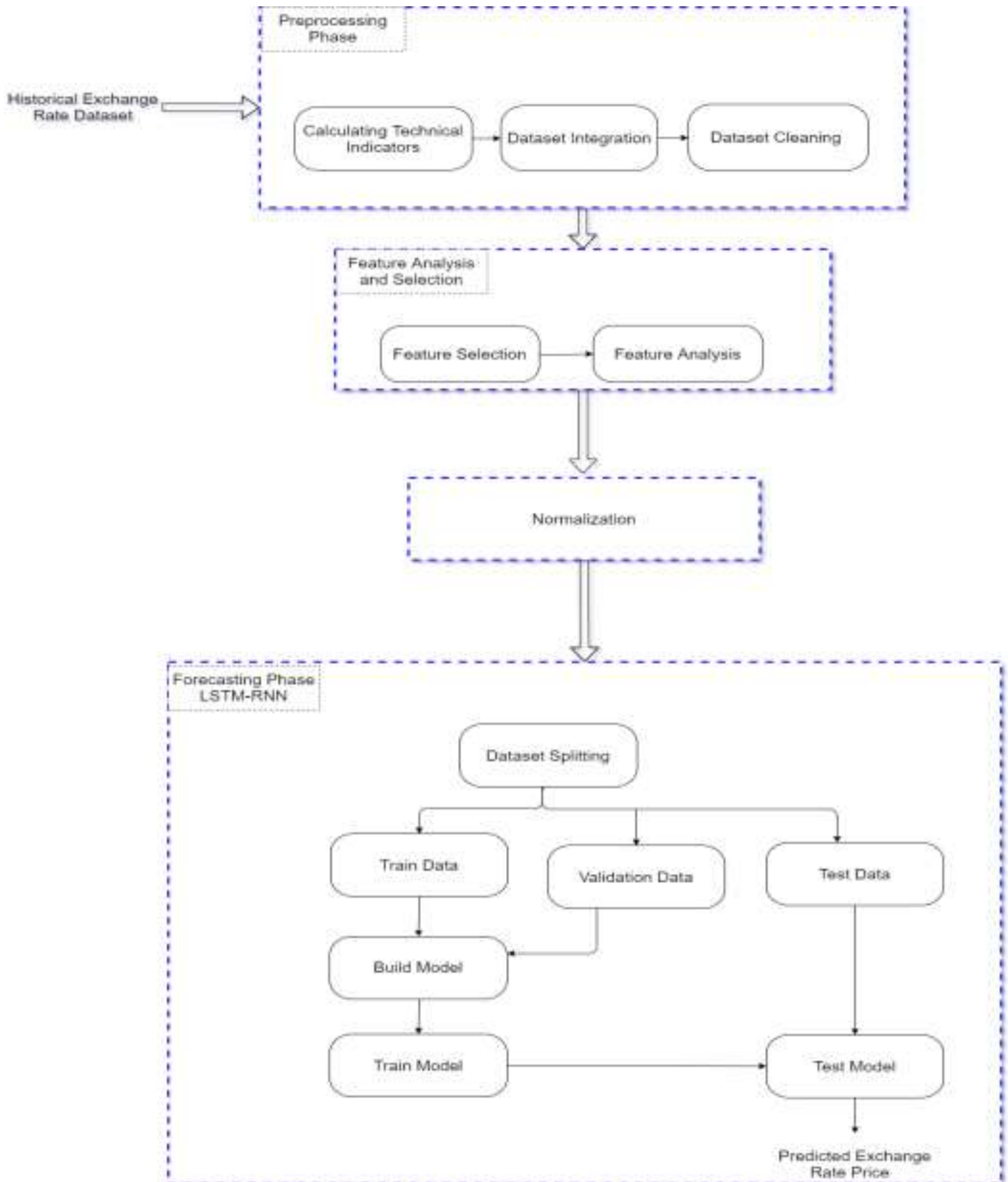

Figure 4. 1 Proposed LSTM-Based Model Architecture. Source [own]

1. Input dataset: this is the first component that is responsible for accepting the input dataset. The dataset contains data before preprocessing.

2. Preprocessing: preprocessing is the second component that we used to clean the data we got from the first component. This component has three sub-components listed below.

   2.1. Calculating technical indicators: chosen technical indicators for the analysis calculated

   2.2. Dataset Integration: The calculated dataset is merged with the initial one.

   2.3. Dataset cleaning: this subcomponent helps to handle records which consist missing value.

   2.5. Data normalization: applying the data scaling mechanism. The data will be between 0&1.

   2.6. Splitting dataset: for training the model and validating and testing its performance,the data need to be split into a train, validation, and test data.

3. Train data: this is used to train the model.

4. Validation data is data that is holdback from training our model and is used to estimate the model. It evaluates loss and model metrics at the end of the epoch. It has two parameters; these are test data and test target.

5. Test data: this data helps to evaluate the model trained by using train data.

6. Build model: we built the simple RNN and LSTM, and the results from these two models are compared and presented as well.

7. Train the model: in this module, the train data will be feed to the implemented model, and the model will train based on the given data.

8. Test the model: in this module, the test data will be fed to the model to measure its performance.

9. Prediction result: this module shows the output of the prediction model. Our model predicts the close price for 20 days of unseen data.

4.3. Data Collection and Preparation

The primary task for research using deep learning is collecting the data needed and preparing it for further preprocessing. The exchange rate dataset for CZK against Euro[2] and CZK against USD[3] is publicly available.

For this research, 22 years (04 January 1999- 22 February 2021) historical dataset for the exchange rate of CZK/EURO and (04 January 1999- 17 November 2020) for CZK/USD is used to see the performance of the model in both currency pairs. A detailed description of the dataset is presented below, along with a table representing the dataset's source.

Table 4. 1 Collected Dataset Description. Source[own]

Forex Pair	**Parameters**	**Data Sources**	**Frequency Data Type**	**Frequency**	**Number of Observation**
EUR/CZK	Price,Close,High,Low,Change	[2]	Numerical	Daily	5774
USD/CZK	Price,Close,High,Low,Change	[3]	Numerical	Daily	5705

The collected dataset doesn't include weekend exchange rate prices. The currency exchange rate dataset consists of five parameters: price (close price), open, high, low, and change. These parameters are described as follows:

- **Price**: refers to the price of an individual forex price when the forex exchange shop closed for the day. It represents the last buy-sell order executed between two traders.
- **Open**: This is the price at which a forex trading started when the opening bell rang, It can be the same as where the exchange shops closed the night before.
- **High:** The highest price at which forex traded during a period.
- **Low:** The minimum price at which forex traded during a period.
- **Change:**The percentage diffrence of the close,open, high and low prices for a specific date.

[2]https://www.investing.com/currencies/eur-czk-historical-data

For every period, the close price is used as a reference point. After all of the activity during the day, traders decided on a price. Financial companies, regulators, and individual investors use the closing price as the traditional measure of the forex value as of a given date when researching historical forex price data.Thus we are interested to forecast the close price and the rest of the parameters taken as the model's input. To make more precise predictions, technical indicators(briefly described in 4.4.1) are also used as input for the model with the dataset's initial features.

Collected Dataset for CZK//EURO

```
[5774 rows x 5 columns]
              Price     Open     High      Low  Change %
Date
1999-01-04  35.1205  35.0870  35.5355  34.4880   -0.0007
1999-01-05  34.9050  35.0810  35.4500  34.7000   -0.0061
1999-01-06  34.7840  34.8810  35.3730  34.0720   -0.0035
1999-01-07  34.9210  34.7190  35.2200  34.5660    0.0039
1999-01-08  35.0385  34.8980  35.2095  34.7440    0.0034
...             ...      ...      ...      ...       ...
2021-02-17  25.8180  25.8140  25.9175  25.7795    0.0012
2021-02-18  25.8830  25.8490  25.9665  25.7455    0.0025
2021-02-19  25.8750  25.9165  25.9400  25.8145   -0.0003
2021-02-21  25.9050  25.8695  25.9085  25.8695    0.0012
2021-02-22  25.8905  25.9050  25.9965  25.8595   -0.0006
```

Figure 4. 2 Initial Collected dataset CZK/Euro. Source [own]

Figure 4. 3 Distribution of Collected dataset CZK/Euro. Source [own]

Figures 4.2 and 4.3 show the exchange rate price dataset of CZK/Euro for 5774 working days.

Collected Dataset for CZK//USD

[5705 rows x 5 columns]

Date	Price	Open	High	Low	Change %
1999-01-04	29.6965	29.9660	30.0915	29.1090	-0.0094
1999-01-05	29.6720	29.6620	29.9710	29.4200	-0.0008
1999-01-06	29.9410	29.6580	30.2790	29.0470	0.0091
1999-01-07	29.8250	30.1690	30.2370	29.5630	-0.0039
1999-01-08	30.2710	29.8020	30.4500	29.6690	0.0150
...	...	...	...	...	...
2020-11-11	22.4110	22.3620	22.5475	22.2995	0.0030
2020-11-12	22.4620	22.4310	22.5495	22.3555	0.0023
2020-11-13	22.3080	22.4660	22.5150	22.3200	-0.0069
2020-11-16	22.2820	22.3375	22.3645	22.1990	-0.0012
2020-11-17	22.3685	22.2685	22.3730	22.2445	0.0039

Figure 4. 4 Initial Collected dataset CZK/USD. Source [own]

[3]https://www.investing.com/currencies/usd-czk-historical-data

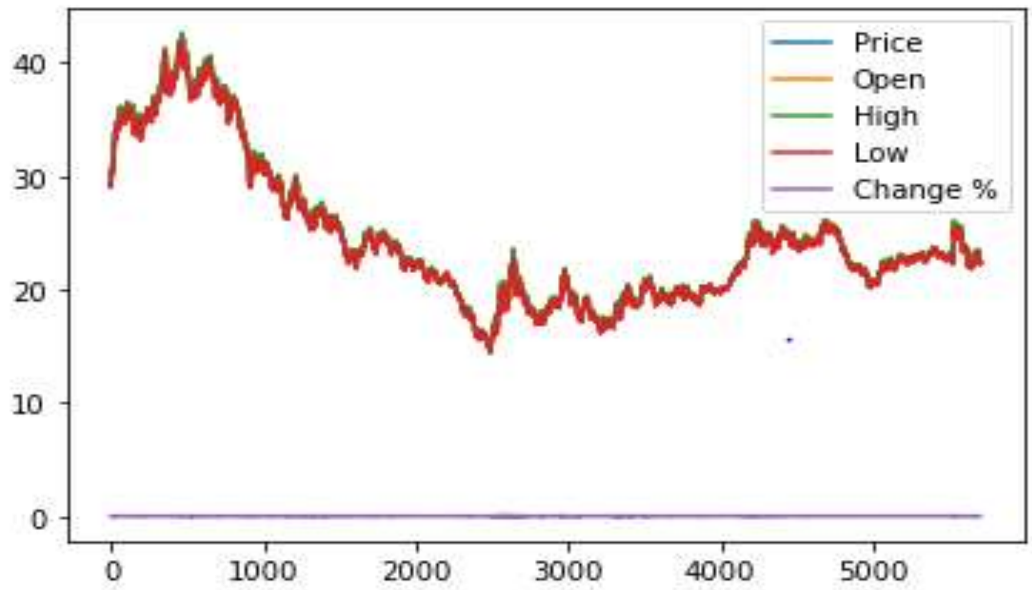

Figure 4. 5 Distribution of Collected dataset CZK/USD. Source [own]

Figures 4.4 and 4.5 depicts the exchange rate price dataset of CZK/USD for 5705 working days.

4.4. Data Pre-processing

Data Preprocessing is a crucial step in data mining and machine learning. In both fields, the expression "garbage in, garbage out" is applicable. Since data collection methods are often uncontrolled, out-of-range values, impossible data combinations, missing values, and other issues may arise. Analyzing and using data that has not been thoroughly screened can lead to errors in forecasting. In the preprocessing phase, there are a series of tasks done on the collected exchange rate dataset. These tasks can help us to prepare and enhance the dataset to be appropriate for the forecast. For the proposed system calculating technical indicators, dataset cleaning, feature selection, and normalization are used at the preprocessing step. We have the same type and number of features for both currency pairs (CZK/Euro and CZK/USD). Accordingly, the preprocessing step is the same for both pairs. The following sub-sections presented the pre-model steps for CZK/Euro.

4.4.1. Calculating Technical Indicators

In this study, technical analysis is performed by excluding other fundamental analysis factors such as economic, social, and political. The calculated technical indicators help us to smooth out price trends by filtering out noise from random short-term fluctuation. Those indicators are calculated using the initial exchange rate dataset. Technical indicators calculated in this research are listed below.

Table 4. 2 Technical Indicators. Source[own].

Group of technical indicators	**Technical Indicator**	**Accompanying indicators**
Trend Indicators	Simple Moving Average.	• 5 days MA • 10 days M.A.
	Cumulative Moving Average	• CMA
	Exponential Moving Average	• with 0.1 weight • with 0.3 weight
	Moving Average Convergence/divergence	• 12-days EMA • 26-days EMA • MACD • Signal Line
Momentum Indicators	Relative Strength Index	• Upward movement • Downward movement • Average 14-days up close • Average 14-days down close • Relative Strength and Relative Strength Index

While performing technical analysis, it is a great move to calculate technical indicators. As we are not taking other factors into consideration, the indicators' role in studying the price movement behavior is undeniable.

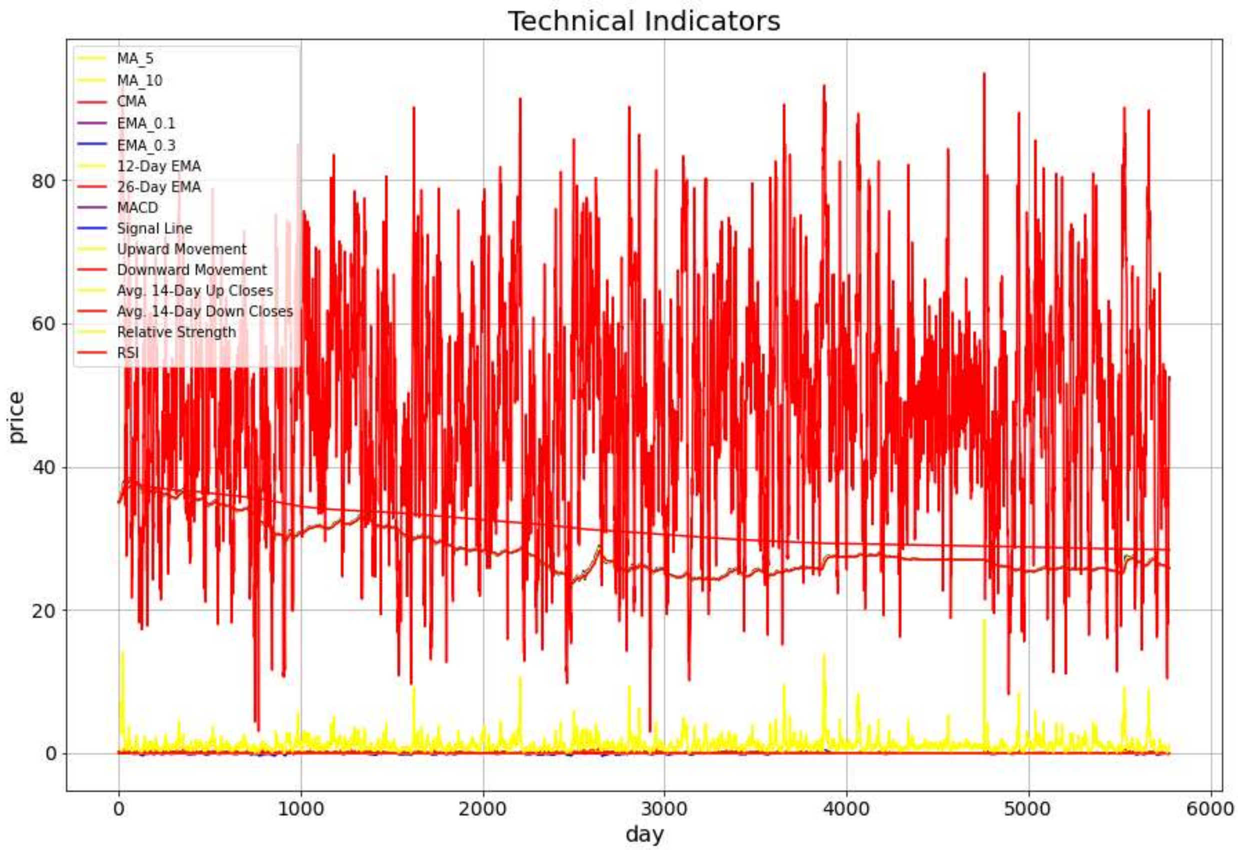

Figure 4. 6 Distribution of Calculated Technical Indicators. Source [own].

4.4.2. Dataset Integration

After calculating the indicators, the original dataset is merged with new calculated values to feed the neural network model. Figure 4.7 depicts the final dataset after merging the initial dataset with the technical indicators.

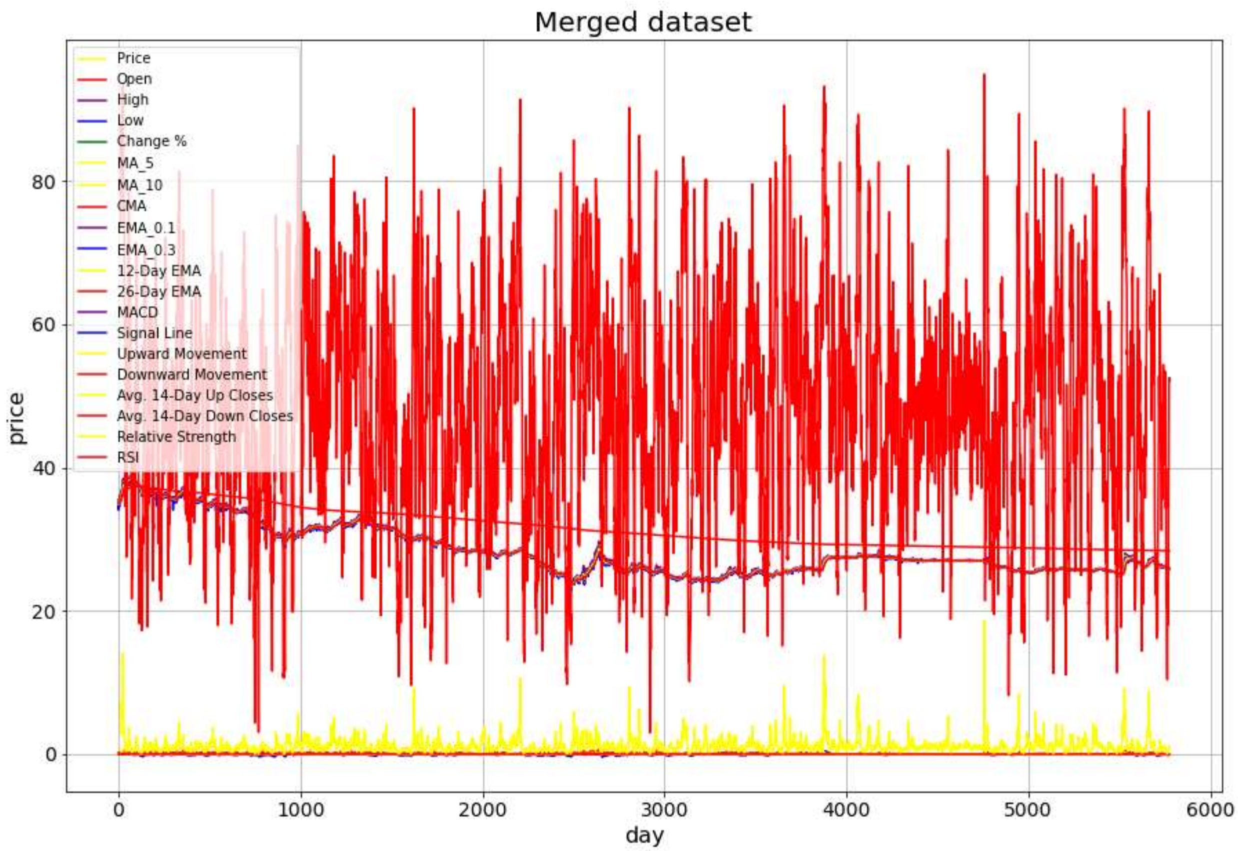

Figure 4. 7 Distribution of the Merged Dataset. Source [own]

All these calculated indicators are not equally valuable for forecasting, to select the crucial parameters we have performed a feature analysis and Selection, which is described in section 4.6 in detail.

4.4.3. Data cleaning

Missing values are checked twice for the initial dataset and the integrated dataset after calculating the technical indicators using Pandas Data Frame. The IsNull() function is used to check the null values. As per Figure 4.8, all the values are False, indicating no missing (NaN) value is found in the initial dataset.

	Price	Open	High	Low	Change %
Date					
1999-01-04	False	False	False	False	False
1999-01-05	False	False	False	False	False
1999-01-06	False	False	False	False	False
1999-01-07	False	False	False	False	False
1999-01-08	False	False	False	False	False
...	...	...	...	...	...
2021-02-17	False	False	False	False	False
2021-02-18	False	False	False	False	False
2021-02-19	False	False	False	False	False
2021-02-21	False	False	False	False	False
2021-02-22	False	False	False	False	False

For the integrated dataset, because we calculated two indicators with a window of 14, NaN values have existed, and our function returned a True value for 14 observations. We drop these rows then left with 5760 observations for CZK/Euro and 5691 observations for CZk/USD to feed our neural network fully organized dataset.

4.4.4. Visualizing dataset

To visualize the distribution of the dataset and identify anomalies, different measurements are calculated based on three categories: central tendency, variability, and measurement of position. Mean as a central tendency, Standard deviation for variability to show how closely the observation is around the mean, 25, 50, and 75 quartiles as a measure of position.

Index	count	mean	std	min	25%	50%	75%	max
Price	5760	28.3847	3.53681	22.9725	25.7134	27.0725	30.5426	38.685
Open	5760	28.3625	3.53512	22.939	25.6915	27.0572	30.534	38.645
High	5760	28.4938	3.55295	23.1165	25.7924	27.157	30.6789	38.7195
Low	5760	28.2539	3.51088	22.881	25.6067	27.005	30.3935	38.349
Change %	5760	-4.95312e-05	0.00401683	-0.0407	-0.0018	-0.0001	0.0016	0.0461
MA_5	5760	28.3882	3.53797	23.1105	25.7105	27.0687	30.5385	38.5318
MA_10	5760	28.3925	3.53954	23.28	25.7064	27.0741	30.5477	38.4963
CMA	5760	31.5295	2.72885	28.4018	29.1079	30.7088	33.676	37.4785
EMA_0.1	5760	28.3997	3.53942	23.4995	25.701	27.0715	30.5562	38.2747
EMA_0.3	5760	28.3887	3.53743	23.1241	25.7133	27.0694	30.5453	38.4743
12-Day EMA	5760	28.3941	3.53869	23.3169	25.7053	27.0735	30.5432	38.3845
26-Day EMA	5760	28.4064	3.54286	23.6507	25.7039	27.0632	30.5612	38.1789
MACD	5760	-0.0122317	0.111048	-0.406919	-0.0739512	-0.0119027	0.0367	0.669023
Signal Line	5760	-0.012116	0.104276	-0.320671	-0.0711055	-0.0118712	0.0346658	0.550972
Upward Movement	5760	0.0362709	0.0742148	0	0	0	0.0445	1.1895
Downward Movement	5760	0.0380545	0.0699229	0	0	0.0015	0.050125	1.068
Avg. 14-Day Up Closes	5760	0.0364033	0.0295921	0.00103571	0.0171429	0.0289107	0.0477857	0.23675
Avg. 14-Day Down Closes	5760	0.038079	0.0268675	0.000678571	0.0191071	0.0336786	0.0509732	0.251107
Relative Strength	5760	1.17412	1.03697	0.0306122	0.604358	0.928034	1.38041	18.6842

Figure 4. 8 Statistical Representation of the dataset. Source [own]

Some parameters such as Change %, MACD, and RSI have different distribution and variability than other variables. Before starting to train our network, this value needs to be transformed into typical ranges, which is conducted in the last section of the preprocessing phase.

4.5. Feature analysis and Selection

There are 20 features in total when putting all the initial parameters and technical analysis indicators together, and all are considered to have a connection with forex analysis and prediction. All the prepared dataset features are not the most relevant features for the forex prediction, so it is essential to investigate, first, which features are the most influential. Feature Selection is a critical step in data analysis, and it can reduce data dimensionality, which often causes longer training time and overfitting.

To train our neural network effectively, it is crucial to remove the less important features for the prediction by using techniques that give input features a score based on how well they predict the target variable. A random forest is a meta estimator that fits several classifying decision trees on various sub-samples of the dataset and uses averaging to improve the predictive accuracy and control over-fitting[12]

In this research, The feature selection is performed after building forest trees from the training set. The model uses Random Forests to select the crucial features from the prepared training dataset. The algorithm is implemented in scikit-learn as the RandomForestRegressor[37]. After applying Random Forest-based selection, we have the coefficient value (importance scores) for each input feature.[12] There is no stated threshold for the importance score to depend on while selecting the best features. For this study, features with an importance score greater than or equal to 0.01372 are chosen. Table 3.4 shows that 11 features fulfilled the selection criteria, and the other nine features are less than the used threshold score value and exculded from the dataset. Eleven features are selected for the prediction from the overall 20 parameters. The feature are marked as 'Selected|' and 'Not selected'.

Table 4. 3 Features Importance Score. Source [own].

Feature	Status	Score
Price	selected	0.51709
Open	Selected	0.27908
High	selected	0.44365

Low	selected	0.47908
Change	selected	0.02007
5 day MA	Selected	0.01431
CMA	Not selected	0.01108
15 day MA	Selected	0.01606
EM_ 0.1	Selected	0.01372
EMA_0.3	Selected	0.02029
MACD	Selected	0.04778
12-Day EMA	Not selected	0.00107
26-Day EMA	Not selected	0.04778
Signal Line	Not selected	0.00107
Upward Movement	Not selected	0.00000
Downward Movement	Not selected	0.00002
Avg. 14-Day Up Closes	Not selected	0.00001
Avg. 14-Day Down Close	Not selected	0.00004
Relative Strength	Not selected	0.00005
RSI	Selected	0.01911

4.6. Normalization

Data transformation is done to fit and converge the model. The performance of a neural network is improved through data transformations (normalization).

The normalization phase involves rescaling our time series data by converting it to a consistent scale, such as the range between -1 and 1 or 0 and 1. For this study, the numerical data is normalized using the Minmaxscaler method from sklearn in the range of [0,1] to make it acceptable for our model and get a more accurate prediction. The normalized data have shown in the figure.

	1	2	3	4	5	6	7	8
1	0.846237	0.847305	0.843419	0.478111	0.840675	0.826706	0.821141	0.842707
2	0.849357	0.872621	0.856349	0.576037	0.852665	0.834395	0.827848	0.853714
3	0.869286	0.867301	0.8579	0.442396	0.860031	0.839068	0.833323	0.859796
4	0.863364	0.875441	0.85984	0.486175	0.865621	0.84426	0.838622	0.865129
5	0.86381	0.891143	0.870507	0.52765	0.872449	0.852044	0.844657	0.872517
6	0.884885	0.912998	0.864688	0.579493	0.883486	0.862552	0.852477	0.884587
7	0.902776	0.923669	0.884407	0.540323	0.893115	0.873508	0.861076	0.897541
8	0.916465	0.916811	0.9124	0.437788	0.902511	0.882002	0.868131	0.904635
9	0.908888	0.914888	0.908262	0.478111	0.91157	0.889425	0.874677	0.910167
10	0.9136	0.919631	0.9124	0.496544	0.919384	0.896845	0.881184	0.915818
11	0.900739	0.96302	0.916667	0.693548	0.932074	0.908868	0.891975	0.934021
12	0.963899	0.985291	0.96037	0.498848	0.943053	0.91931	0.902353	0.948689

Figure 4. 9 Sample Normalized dataset. Source [own]

The normalization sub-phase is the last task in the preprocessing phase, and now all points in our datasets are in the range of [0,1]. Our dataset is well prepared, and it is ready to feed to the neural network to perform the forecasting.

4.7. Forecasting Phase Using Simple RNN and LSTM

4.7.1 Dataset Splitting

Before developing the model, splitting the sample dataset into a training and the testing and validation dataset is the most common approach used in machine learning to build and evaluate regression and classification models. This study's entire dataset for both pairs CZK/EURO and CZK/USD is divided into three parts: training dataset and testing dataset.

The entire dataset is divided into three parts.

- **Training dataset**: 13 and half year daily data with the observation of 5000 is used for training the neural network model.
- **Testing dataset**: 20 days of data is used for testing the developed neural network model.

- **Validation dataset**: 2 years of data with 740 observations is used as a validation dataset to validate the training time model.

The recurrent layers predict the price for each frame in the input sequence. The proposed LSTM model consists of three layers: an input layer, the hidden layer, and the output layer. The signals from the input layer are connected to the given nodes of the hidden layer. The output of the hidden layer nodes is connected to the output layer. The output from the output layer will be fed back to the hidden layer nodes. LSTM cell used to remember the previous output feed as input for the current output. LSTM stores the information with three gates, namely input gate, forget gate, and output gate. The input gate holds the value that comes from the output layer and gives it to the output gate. The output gets to accept the data from the input gate then feeds it to the hidden layer, and the forget gate is used to forget the previous input gate data. So the input gate accepts new data from the output layer.

Supervised learning is performed in this study and the dataset has structure of input(X) train, output(Y) train,input(X) test,output(Y) test, input(X) validate, output(Y) validate. Where X is input for and Y is the label for the supervision. Model validation is performed using input test and output test data.

4.7.2. LSTM Model Architecture

Hyperparameter settings

Fixed and variable hyperparameters were set for the proposed network architecture. The network is evaluated using the hyperparameters mentioned below.

- Input shape: the input size is 11, holds the selected features for the prediction
- Activation function: relu and sigmoid is used as an activation function for each hidden layer node and the output layer node.
- Learning rate: The network is tested with the default Keras value of 0.0001.
- Learning rate optimizer: Adam has used an optimizer for error minimization.
- Batch size: defines the number of samples to work through before updating the internal model parameters.32 samples are utilized in each batch.

- Epoch: The number of epochs determines when the learning algorithm will work through the entire training. The model is trained until the epoch of index `epochs` is reached, which helps tackle overfitting. Ten epochs are used to train and validate the dataset with a reasonable test execution time.

```
regressor = keras.Sequential()
regressor.add(LSTM(4, activation='relu', return_sequences=(True),input_shape=(1,11)))
regressor.add(LSTM(4,activation='sigmoid',return_sequences=(False)))
regressor.add(Dense(1) )
regressor.compile(optimizer='adam',loss="mean_squared_error")
```

Figure 4. 10 Code Snippet of the proposed LSTM Model. Source [own]

4.7.3. Error metrics and Accuracy

Mean squared error metrics are used in this study to calculate the model's error and evaluate the model based on this esult.

Mean Squared Error

Mean-squared error is one of the most commonly used measures of error for numeric prediction. This value is computed by taking the average of the squared differences between each computed value and its corresponding actual value. MSE measure using the following formula:

$$MSE=1/n \sum_{k=0}^{n}[(actual-predicted)^2] \qquad (4.1)$$

Where: n is the number of rows.

The model error is calculated using the above formula, and the performance is evaluated as well. An error minimization or error optimization technique is needed to improve the model's efficiency. As a result, the Adam process is used as an optimizer.

Accuracy

Accuracy is the number of accurate predictions made by the model for all types of predictions. It can be computed in a variety of ways. The approach below is used to calculate both models' accuracy (simple RNN and LSTM).

$$\text{Accuracy} = 1 - \text{MSE} \tag{4.2}$$

5. Results and Discussion

5.1. Simple RNN Prediction

The results obtained from the simple RNN model for both currency pairs is presented below. Model loss evaluation and 20 days predicted vs actual forex price is also discussed.

5.1.1. CZK/Euro

The model loss and graphical representation of the predicted result versus actual values for CZK/Euro are stated below.

```
Epoch 1/10
157/157 [==============================] - 1s 3ms/step - loss: 0.0032 - val_loss: 0.0013
Epoch 2/10
157/157 [==============================] - 0s 2ms/step - loss: 0.0027 - val_loss: 9.7368e-04
Epoch 3/10
157/157 [==============================] - 0s 2ms/step - loss: 0.0024 - val_loss: 7.8512e-04
Epoch 4/10
157/157 [==============================] - 0s 2ms/step - loss: 0.0026 - val_loss: 6.6413e-04
Epoch 5/10
157/157 [==============================] - 0s 2ms/step - loss: 0.0030 - val_loss: 5.8425e-04
Epoch 6/10
157/157 [==============================] - 0s 2ms/step - loss: 0.0033 - val_loss: 5.3116e-04
Epoch 7/10
157/157 [==============================] - 0s 2ms/step - loss: 0.0036 - val_loss: 4.9570e-04
Epoch 8/10
157/157 [==============================] - 0s 2ms/step - loss: 0.0038 - val_loss: 4.7175e-04
Epoch 9/10
157/157 [==============================] - 0s 2ms/step - loss: 0.0040 - val_loss: 4.5525e-04
Epoch 10/10
157/157 [==============================] - 0s 2ms/step - loss: 0.0041 - val_loss: 4.4353e-04
```

Figure 5. 1 Model loss. Source [own]

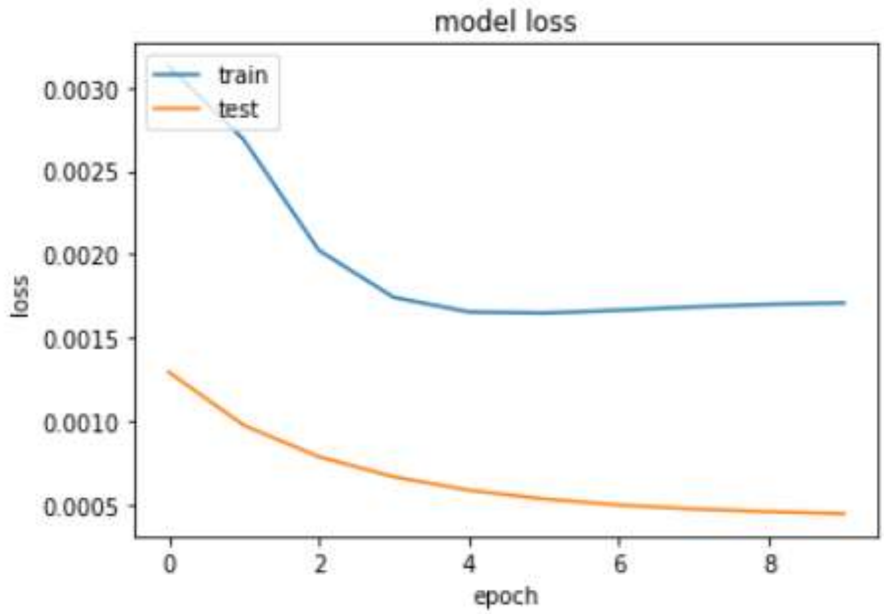

Figure 5. 2 Model loss graph. Source [own]

The simple RNN model scored 89.91% accuracy with MSE of 0.1009 . Figures 5.1 and 5.2 showed the training and validation loss for the utilized ten epochs.

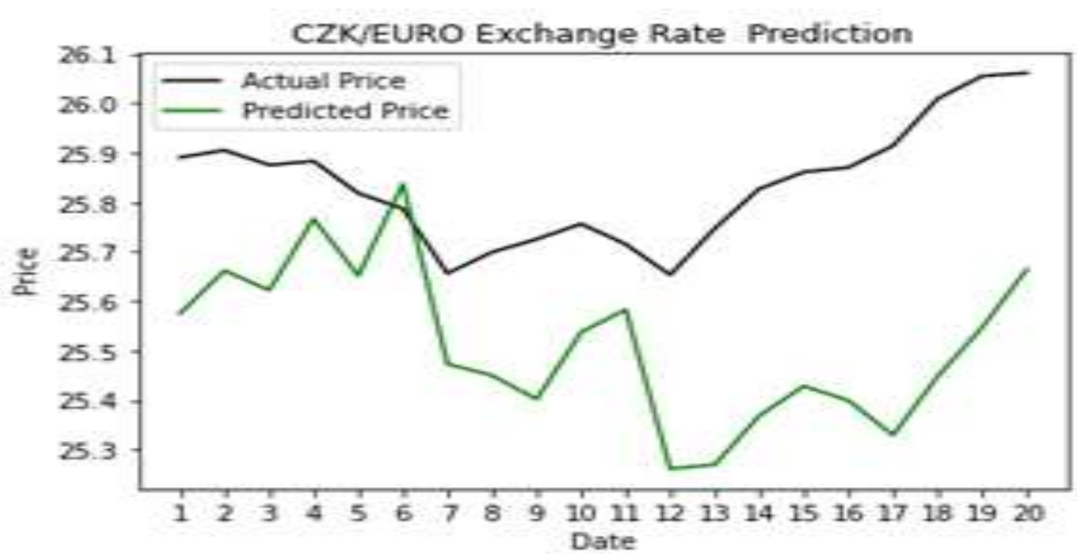

Figure 5. 3 Actual vs Predicted Forex Price. Source [own]

Figure 5.3 showed the actual and predicted forex price for the last 20 days to see how close our model predicts compared to the actual price value. Our simple RNN does not perform well in predicting the price; it signifies a few days' prices closer to the actual price.

5.1.2. CZK/USD

The model loss and graphical representation of the predicted result versus actual values for CZK/USD are presented below.

```
Epoch 1/10
157/157 [==============================] - 2s 3ms/step - loss: 0.0138 - val_loss: 8.4113e-04
Epoch 2/10
157/157 [==============================] - 0s 2ms/step - loss: 0.0205 - val_loss: 4.3622e-04
Epoch 3/10
157/157 [==============================] - 0s 2ms/step - loss: 0.0158 - val_loss: 3.2701e-04
Epoch 4/10
157/157 [==============================] - 0s 2ms/step - loss: 0.0114 - val_loss: 2.6048e-04
Epoch 5/10
157/157 [==============================] - 0s 2ms/step - loss: 0.0084 - val_loss: 2.2203e-04
Epoch 6/10
157/157 [==============================] - 0s 2ms/step - loss: 0.0064 - val_loss: 2.0151e-04
Epoch 7/10
157/157 [==============================] - 0s 2ms/step - loss: 0.0051 - val_loss: 1.8983e-04
Epoch 8/10
157/157 [==============================] - 0s 2ms/step - loss: 0.0043 - val_loss: 1.8043e-04

Epoch 9/10
157/157 [==============================] - 0s 2ms/step - loss: 0.0037 - val_loss: 1.7002e-04
Epoch 10/10
157/157 [==============================] - 0s 2ms/step - loss: 0.0033 - val_loss: 1.5797e-04
```

Figure 5. 4 Model loss. Source [own]

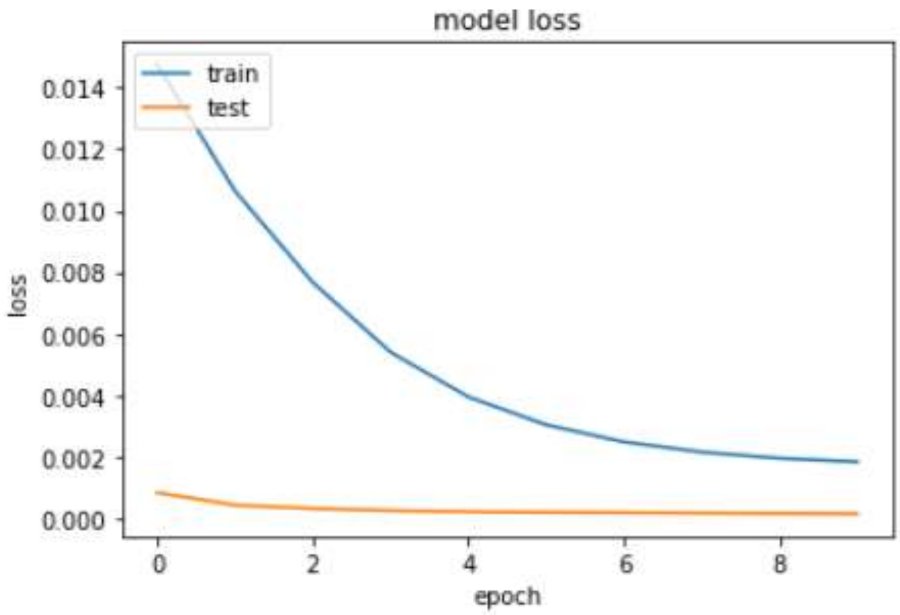

Figure 5. 5 Model loss graph. Source [own]

Figures 5.4 and 5.5 showed the training and validation loss for the utilized 10 epochs.The simple RNN model achieved 82.81% accuracy with MSE of 0.1719.

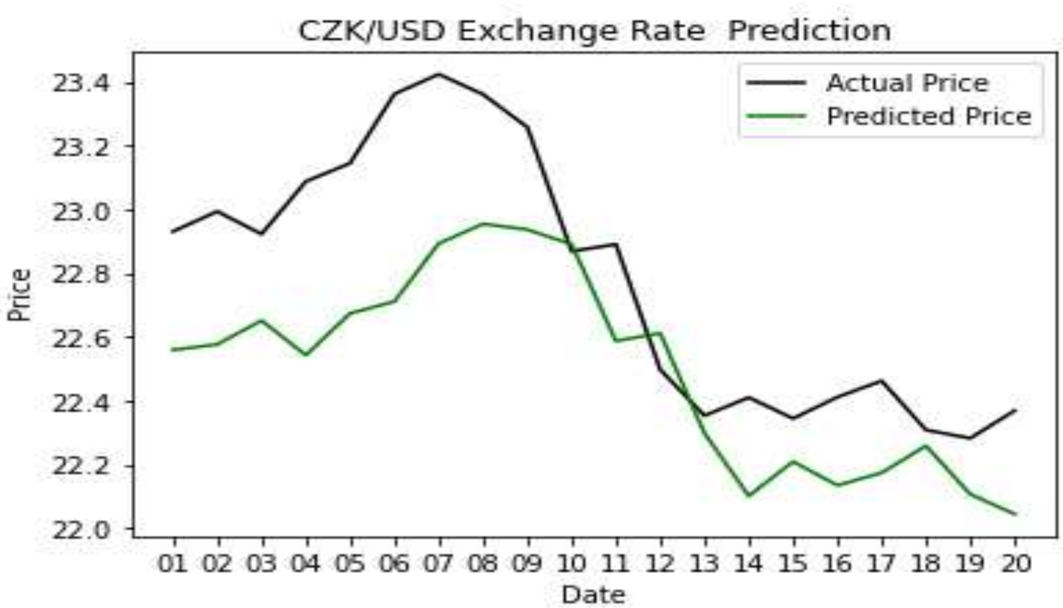

Figure 5. 6 Actual vs Predicted Forex Price. Source [own

Figure 5.6 shows the actual and predicted forex price for the last 20 days to see how close our model predicts compared to the actual price value.The simple RNN model does not perform well in predicting the price; it predicts a few day's prices closer to the actual price. The model has a significantly closer result for both forex pairs, there is an accuracy different of around 5% for the simple RNN model for CZK/Euro and CZK/USD.

5.2. LSTM Prediction

The LSTM model forecasting results including model loss evaluation and 20 days predicted price for both currency pairs is presented in 5.2.1 and 5.2.2 subsections below.

5.2.1. CZK/Euro

The model loss and graphical representation of the predicted result versus actual values for CZK/Euro are presented below.

```
Epoch 1/10
157/157 [==============================] - 4s 6ms/step - loss: 0.4439 - val_loss: 0.0161
Epoch 2/10
157/157 [==============================] - 0s 3ms/step - loss: 0.2991 - val_loss: 0.0022
Epoch 3/10
157/157 [==============================] - 1s 3ms/step - loss: 0.2224 - val_loss: 0.0024
Epoch 4/10
157/157 [==============================] - 0s 3ms/step - loss: 0.1770 - val_loss: 0.0064
Epoch 5/10
157/157 [==============================] - 0s 3ms/step - loss: 0.1501 - val_loss: 0.0096
Epoch 6/10
157/157 [==============================] - 0s 3ms/step - loss: 0.1339 - val_loss: 0.0111
Epoch 7/10
157/157 [==============================] - 0s 3ms/step - loss: 0.1230 - val_loss: 0.0098
Epoch 8/10
157/157 [==============================] - 0s 3ms/step - loss: 0.1142 - val_loss: 0.0065
Epoch 9/10
157/157 [==============================] - 0s 3ms/step - loss: 0.1049 - val_loss: 0.0036
Epoch 10/10
157/157 [==============================] - 0s 3ms/step - loss: 0.0939 - val_loss: 0.0022
```

Figure 5. 7 Model loss. Source [own]

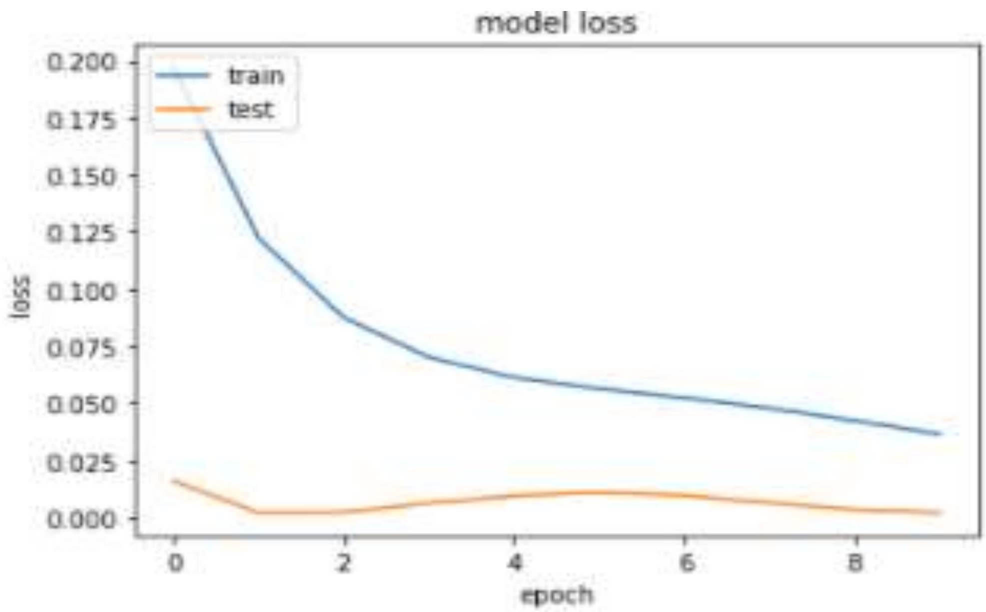

Figure 5. 8 Model loss graph. Source [own]

The LSTM based prediction achieved 97.63% accuracy with 2.36 % MSE. Figures 5.7 and 5.8 depicted the training and validation loss using 10 epochs. The LSTM model performed the prediction with less error value and higher accuracy.

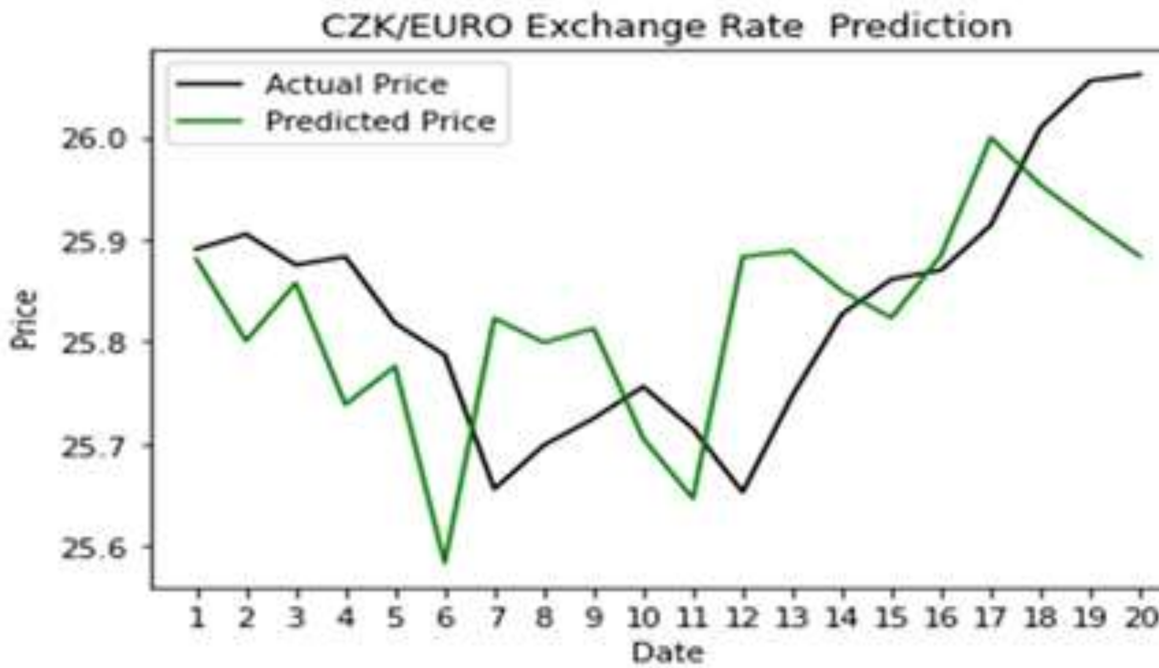

Figure 5. 9 Actual vs Predicted Forex Price. Source [own]

Figure 5.9 depicted the actual and predicted forex price for the last 20 days to see how close our model predicts compared to the actual price.LSTM based model performed well in predicting the price for CZK/Euro forex pair.The predicted prices are closer to the actual price in the market .

5.2.2. CZK/USD

The model loss and graphical representation of the predicted result versus actual values for CZK/USD are presented below.

```
Epoch 1/10
157/157 [==============================] - 4s 7ms/step - loss: 0.4364 - val_loss: 0.0341
Epoch 2/10
157/157 [==============================] - 0s 3ms/step - loss: 0.2181 - val_loss: 0.0033
Epoch 3/10
157/157 [==============================] - 0s 3ms/step - loss: 0.1314 - val_loss: 0.0010
Epoch 4/10
157/157 [==============================] - 0s 3ms/step - loss: 0.1040 - val_loss: 0.0010
Epoch 5/10
157/157 [==============================] - 0s 3ms/step - loss: 0.0929 - val_loss: 9.6795e-04
Epoch 6/10
157/157 [==============================] - 0s 3ms/step - loss: 0.0856 - val_loss: 8.5924e-04
Epoch 7/10
157/157 [==============================] - 0s 3ms/step - loss: 0.0789 - val_loss: 7.5721e-04
Epoch 8/10
157/157 [==============================] - 0s 3ms/step - loss: 0.0724 - val_loss: 6.6646e-04
Epoch 9/10
157/157 [==============================] - 0s 3ms/step - loss: 0.0659 - val_loss: 5.8600e-04
Epoch 10/10
157/157 [==============================] - 0s 3ms/step - loss: 0.0594 - val_loss: 5.1454e-04
```

Figure 5. 10 Model loss. Source [own]

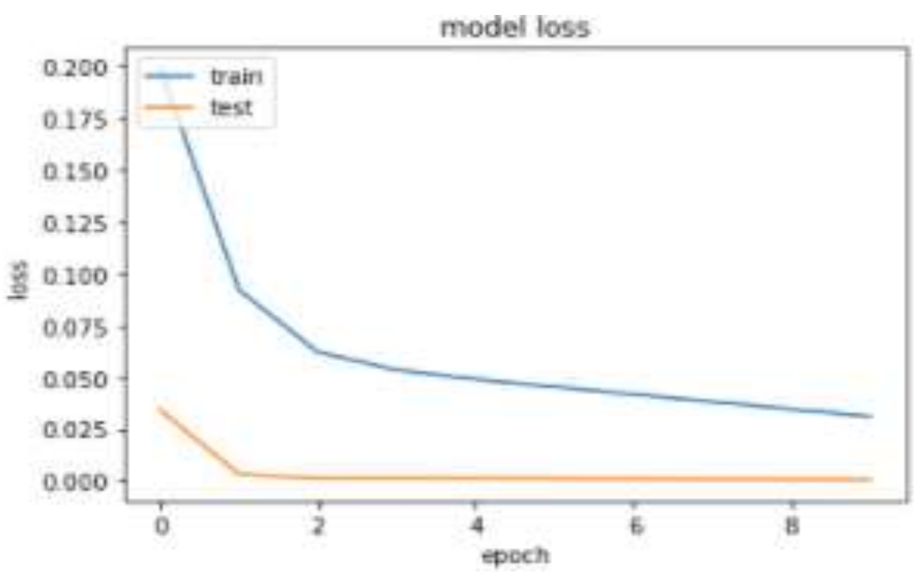

Figure 5. 11 Model loss graph. Source [own]

The LSTM based prediction achived 93.68% accuracy with 6.32% MSE.The following diagram shows the training and validation loss using 10 epochs. The LSTM model performed well in the prediction with less error value and higher accuracy.

Figure 5. 12 Actual vs Predicted Forex Price. Source [own]

Figure 5.12 depicted the actual and predicted forex price for the last 20 days to see how close our model predicts compared to the actual price value. The LSTM based model performs well in predicting the exchange rate prices for CZK/USD pair with higher accuracy and MSE of 0.0632 . It indicates that predicted forex prices are closer to the actual price.

5.3. Result Comparison

Table 5. 1 Result Comparison. Source [own]

	Currency Pair	Simple RNN	LSTM
MSE	CZK/Euro	0.1009	0.0236
	CZK/Dollar	0.1719	0.0632
Accuracy	CZK/Euro	89.91%	97.64%
	CZK/Dollar	82.81%	93.68%

The proposed models are compared using MSE, and accuracy. In both MSE and accuracy, the LSTM model shows the best performance than simple RNN for both currency pairs.When the result are compared, LSTM is with a minimum MSE of 2.36 % and 6.32% for CZK/Euro and CZK/USD. LSTM achieved accuracy of 97.64% and 93.68% for CZK/Euro and CZK/USD respectively. From the results achieved for both curreny paris LSTM has the least error than simple RNN. Generally, The LSTM forcasting model outperforms the simple RNN model by with high accuracy and minimum MSE for predicting the forex price for CZK/Euro and CZK/USD.

6. Conclusion

Foreign currency exchange rate prediction can assist individuals, economist, and business owners in the process of decision-making by evaluating the benefits and risks attached to the market. Fundamental and technical approaches are widely used methodologies to perform forex forecasting. qaRecently deep neural networks have been applied to forex forcasting and they show remarkable results. RNN based models have been applied for time series problems. RNN based models are favourable for processing and predicting sequential data. In this study, a forecasting model that can predict future forex prices using RNN is developed. To train and test the proposed model, the forex dataset was collected from investing.com financial platform. Simple RNN and LSTM models are designed and implemented to evaluate which RNN based model performs better for the prediction. To further investigate the performance the models are also compared on two forex pairs. The overall result indicates that the proposed LSTM model outperforms the simple RNN model with an accuracy of 97.64% and a mean squared error of 2.36% for the CZK/Euro forex pair. The LSTM model performed the forecast efficiently for CZK/USD pair as well with an accuracy of 93.68% and a mean squared error of 6.32%. Based on the results gained LSTM model provides promising results for solving problems with dynamic and non-linear relationship data like forex price. After performing forex forecasting for Czech koruna, based on the achieved results, this research work supports the idea of exchange rates being difficult to forecast without considering the fundamental economic factors that affect the forex market.

The proposed model meets the current criteria. However, there are a few issues that need additional work. First, the proposed model performed the prediction by taking a relatively close value of the current day's price, to gain better accuracy and to make the model suitable for the real-world scenario the model can be further updated by applying fundamental analysis to add other fundamental economic factors with the forex price dataset. Another recommended work is to predict the price movement by stating if the price will be appreciated, depreciated, or will be the same, rather than anticipating the exact exchange rate price for the next day. As a result, this research work presented a contribution that could be strengthened and implemented with additional effort.

7. Reference

[1] P. Wang, "Foreign Exchange Markets and Foreign Exchange Rates," 2020.

[2] E. Kayacan, B. Ulutas, and O. Kaynak, "Grey system theory-based models in time series prediction," *Expert Systems with Applications*, vol. 37, no. 2, Mar. 2010, doi: 10.1016/j.eswa.2009.07.064.

[3] T. Fischer and C. Krauss, "Deep learning with long short-term memory networks for financial market predictions," *European Journal of Operational Research*, vol. 270, no. 2, Oct. 2018, doi: 10.1016/j.ejor.2017.11.054.

[4] A. J. Dautel, W. K. Härdle, S. Lessmann, and H.-V. Seow, "Forex exchange rate forecasting using deep recurrent neural networks," *Digital Finance*, vol. 2, no. 1–2, pp. 69–96, Sep. 2020, doi: 10.1007/s42521-020-00019-x.

[5] A. Emam and H. Min, "The artificial neural network for forecasting foreign exchange rates," *International Journal of Services and Operations Management*, vol. 5, no. 6, pp. 740–757, 2009, doi: 10.1504/IJSOM.2009.026772.

[6] D. C. Yıldırım, I. H. Toroslu, and U. Fiore, "Forecasting directional movement of Forex data using LSTM with technical and macroeconomic indicators," *Financial Innovation*, vol. 7, no. 1, Dec. 2021, doi: 10.1186/s40854-020-00220-2.

[7] L. Qi, M. Khushi, and J. Poon, "Event-Driven LSTM For Forex Price Prediction."

[8] L. Ni, Y. Li, X. Wang, J. Zhang, J. Yu, and C. Qi, "Forecasting of Forex Time Series Data Based on Deep Learning," in *Procedia Computer Science*, 2019, vol. 147, pp. 647–652, doi: 10.1016/j.procs.2019.01.189.

[9] J. Yao and C. Lim Tan, "A case study on using neural networks to perform technical forecasting of forex."

[10] M. O. Özorhan, "FORECASTING DIRECTION OF EXCHANGE RATE FLUCTUATIONS WITH TWO DIMENSIONAL PATTERNS AND CURRENCY STRENGTH A THESIS SUBMITTED TO THE GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES OF MIDDLE EAST TECHNICAL UNIVERSITY," 2017.

[11] CompareRemit, "8 Key Factors that Affect Foreign Exchange Rates," Apr. 2020. [Online]. Available: *https://www.compareremit.com/money-transfer-guide/key-factors-affecting-currency-exchange-rates/*.[Accessed: 29-jul-2020].

[12] T. Zhou, W. Professor, and X. Liu, "FOREX TREND FORECASTING BASED ON LONG SHORT TERM MEMORY AND ITS VARIATIONS WITH HYBRID ACTIVATION FUNCTIONS."

[13] F. Ye, L. Zhang, D. Zhang, H. Fujita, and Z. Gong, "A novel forecasting method based on multi-order fuzzy time series and technical analysis," *Information Sciences*, vol. 367–368, pp. 41–57, Nov. 2016, doi: 10.1016/j.ins.2016.05.038.

[14] H. T. Sykora, D. Bachrathy, and G. Stepan, "Stochastic semi-discretization for linear stochastic delay differential equations," *International Journal for Numerical Methods in Engineering*, vol. 119, no. 9, Aug. 2019, doi: 10.1002/nme.6076.

[15] Admiral Market, "Using a Forex Volatility Indicator," Nov. 2018. [Online]. Available: *https://admiralmarkets.com/education/articles/forex-indicators/using-a-forex-volatility-indicator/*.[Accessed: 10-jan-2021].

[16] A. N. Refenes, A. Zapranis, and G. Francis, “Stock Performance Modeling Using Neural Networks: A Comparative Study With Regression Models,” 1994.
[17] H. Palangi, R. Ward, and L. Deng, “Distributed Compressive Sensing: A Deep Learning Approach,” Aug. 2015, doi: 10.1109/TSP.2016.2557301.
[18] Y. S. R. and I. L. T. Hope, “Learing TensorFlow. .”
[19] Y. Zhuang, M. Almeida, M. Morabito, and W. Ding, “Crime Hot Spot Forecasting: A Recurrent Model with Spatial and Temporal Information,” Aug. 2017, doi: 10.1109/ICBK.2017.3.
[20] Ö. Yildirim, “A novel wavelet sequence based on deep bidirectional LSTM network model for ECG signal classification,” *Computers in Biology and Medicine*, vol. 96, May 2018, doi: 10.1016/j.compbiomed.2018.03.016.
[21] J. Y. and Y. R. W. Bao, “A deep learning framework for financial time series using stacked autoencoders and longshort term memory,” vol. 64.
[22] E. Khosla, D. Ramesh, R. P. Sharma, and S. Nyakotey, “RNNs-RT: Flood based Prediction of Human and Animal deaths in Bihar using Recurrent Neural Networks and Regression Techniques,” in *Procedia Computer Science*, 2018, vol. 132, pp. 486–497, doi: 10.1016/j.procs.2018.05.001.
[23] B. Belesti Yenegeta Bizuneh, “SCHOOL OF RESEARCH AND GRADUATE STUDIES FACULTY OF COMPUTING DETECTION AND GRADING OF TRACHOMA USING DEEP CONVOLUTIONAL NEURAL NETWORK BAHIR DAR, ETHIOPIA,” 2019.
[24] J. Schmidhuber, “Deep Learning in neural networks: An overview,” *Neural Networks*, vol. 61. Elsevier Ltd, pp. 85–117, Jan. 01, 2015, doi: 10.1016/j.neunet.2014.09.003.
[25] “Artificial Neural Network Based Numerical Solution of Ordinary Differential Equations,” 2012.
[26] H. Palangi, R. Ward, and L. Deng, “Distributed Compressive Sensing: A Deep Learning Approach,” *IEEE Transactions on Signal Processing*, vol. 64, no. 17, Sep. 2016, doi: 10.1109/TSP.2016.2557301.
[27] S. Hochreiter, “Long Short-Term Memory,” vol. 1780.
[28] C. Olah, “Understanding LSTM Networks,” 2015. [Online]. Available: https://colah.github.io/posts/2015-08-Understanding-LSTMs/. [Accessed: 2-Dec-2020].
[29] K. Greff, R. K. Srivastava, J. Koutnik, B. R. Steunebrink, and J. Schmidhuber, “LSTM: A Search Space Odyssey,” *IEEE Transactions on Neural Networks and Learning Systems*, vol. 28, no. 10, Oct. 2017, doi: 10.1109/TNNLS.2016.2582924.
[30] K. A. Hamad and M. Kaya, “A Detailed Analysis of Optical Character Recognition Technology,” 2016.
[31] T. Xayasouk, H. Lee, and G. Lee, “Air Pollution Prediction Using Long Short-Term Memory (LSTM) and Deep Autoencoder (DAE) Models,” doi: 10.3390/su12062570.
[32] Y. L. Yong, Y. Lee, X. Gu, P. P. Angelov, D. C. L. Ngo, and E. Shafipour, “Foreign currency exchange rate prediction using neuro-fuzzy systems,” in *Procedia Computer Science*, 2018, vol. 144, pp. 232–238, doi: 10.1016/j.procs.2018.10.523.
[33] K. Chandar, “Forecasting of foreign currency exchange rate using neural network Fusion techniques View project,” 2015. [Online]. Available: https://www.researchgate.net/publication/282296144. [Accessed: 16-Nov-2020].

[34] A. J. Dautel, W. K. Härdle, S. Lessmann, and H.-V. Seow, “Forex exchange rate forecasting using deep recurrent neural networks,” *Digital Finance*, vol. 2, no. 1–2, pp. 69–96, Sep. 2020, doi: 10.1007/s42521-020-00019-x.
[35] M. Rehman, G. M. Khan, and S. A. Mahmud, “Foreign Currency Exchange Rates Prediction Using CGP and Recurrent Neural Network,” *IERI Procedia*, vol. 10, pp. 239–244, 2014, doi: 10.1016/j.ieri.2014.09.083.
[36] A. Adetunji, P. A. Adio, T. Akintomide, and A. Bidemi, “Artificial Neural Network Model for Forecasting Foreign Exchange Rate,” 2011.
[37] L. Breiman, “Random Forests,” 2001.

8. Appendix

Appendix 1: Sample CSV dataset used for training

	A	B	C	D	E	F
1	Date	Price	Open	High	Low	Change %
2	4-Jan-99	35.1205	35.087	35.5355	34.488	-0.0007
3	5-Jan-99	34.905	35.081	35.45	34.7	-0.0061
4	6-Jan-99	34.784	34.881	35.373	34.072	-0.0035
5	7-Jan-99	34.921	34.719	35.22	34.566	0.0039
6	8-Jan-99	35.0385	34.898	35.2095	34.744	0.0034
7	11-Jan-99	35.2415	35.004	35.3205	34.607	0.0058
8	12-Jan-99	35.453	35.21	35.53	34.904	0.006
9	13-Jan-99	35.829	35.41	36.206	35.252	0.0106
10	14-Jan-99	35.805	35.751	36.085	35.066	-0.0007
11	15-Jan-99	35.5975	35.728	36.1385	35.379	-0.0058
12	18-Jan-99	35.536	35.603	35.837	35.131	-0.0017
13	19-Jan-99	35.6985	35.514	35.8235	35.366	0.0046
14	20-Jan-99	35.972	35.669	36.161	35.262	0.0077
15	21-Jan-99	36.164	36.039	36.251	35.569	0.0053
16	22-Jan-99	36.2555	36.137	36.3715	35.945	0.0025
17	25-Jan-99	36.284	36.23	36.337	35.927	0.0008
18	26-Jan-99	36.623	36.279	36.732	36.127	0.0093
19	27-Jan-99	36.54	36.592	36.649	36.151	-0.0023
20	28-Jan-99	36.595	36.499	36.776	36.181	0.0015
21	29-Jan-99	36.782	36.506	37.021	36.346	0.0051
22	1-Feb-99	37.135	36.837	37.362	36.256	0.0096

Appendix 2: LSTM model sample code

```
import pandas as pd
from sklearn.model_selection import train_test_split
from sklearn.metrics  import mean_squared_error
from keras.layers import LSTM
from keras.models import Sequential
from keras.layers import Dense
import matplotlib.pyplot as plt
from pandas import datetime
from keras.layers.core import Dense, Activation, Dropout
regressor = keras.Sequential()
regressor.add(LSTM(4, activation='relu', return_sequences=(True),input_shape=(1,11)))
regressor.add(LSTM(4,activation='sigmoid',return_sequences=(False)))
regressor.add(Dense(1) )
regressor.compile(optimizer='adam',loss="mean_squared_error")
dataset=pd.read_csv(r"C:\Users\Duni\Desktop\dollar.csv",parse_dates=['Date'], index_col='Date')

plt.title('Dataset')
dataset.plot()
plt.show()

print(dataset.isnull() )
dataset.dropna(inplace=True)
print(dataset)
#feature scaling (normalization)
from sklearn.preprocessing import MinMaxScaler
sc=MinMaxScaler()
traning_data=dataset.iloc[:,0:12].values
#getting the input and the outputs
x= traning_data[:, 1:12]
x=sc.fit_transform(x)

Xx = sc.inverse_transform(x)
plt.title('Traning Dataset')
plt.plot(Xx)
plt.ylabel('Price')
plt.xlabel('day')
plt.show()
y= traning_data[:, 0:1]
y=sc.fit_transform(y)
```

Appendix 3: Sample Code for Feature Selection

```
from sklearn.datasets import make_regression
from sklearn.ensemble import RandomForestRegressor
from matplotlib import pyplot
import pandas as pd
from sklearn.datasets import make_regression
from sklearn.ensemble import RandomForestRegressor
from matplotlib import pyplot
from sklearn.model_selection import train_test_split
from sklearn.feature_selection import SelectFromModel
from sklearn.ensemble import RandomForestClassifier
from sklearn.model_selection import train_test_split
dataframe = pd.read_csv(r"C:\Users\Duni\Desktop\pt\feature.csv", header=0)
X=dataframe [['Open','High','Low','Change %','MA_5','MA_10','CMA','EMA_0.1','EMA_0.3','12-Day EMA'
,'26-Day EMA','MACD','Signal Line','Upward Movement','Upward Movement','Downward Movement','Avg. 1
y=dataframe ['Price']  # Labels
y=y.astype('int')
# Split dataset into training set and test set
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.3)
sel = SelectFromModel(RandomForestClassifier(n_estimators = 100))
sel.fit(X_train, y_train)
sel.get_support()
selected_feat= X_train.columns[(sel.get_support())]
len(selected_feat)
```

Appendix 4: Sample Code for calculating technical indicators

```
df_26DayEMA=df_T['Price'].ewm(span=26).mean()
df_final=pd.read_csv(r"C:\Users\Duni\Desktop\euro.csv")
df_final['MA_5'] = df_forex1
df_final['MA_10'] = df_forex2
df_final['CMA'] = df_cumulative
df_final['EMA_0.1'] = df_EMA
df_final['EMA_0.3'] = df_EMA1
df_final['12-Day EMA'] = df_12DayEMA
df_final['26-Day EMA']=df_26DayEMA
df_MACD = df_final['12-Day EMA'] - df_final['26-Day EMA']
df_final['MACD']=df_MACD
df_SignalLine = df_final['MACD'].ewm(span=9).mean()
df_final['Signal Line'] = df_SignalLine
## calculating Relative Strength Index   momentum indicator
diff = df_final['Price'].diff()
up, down = diff.copy(), diff.copy()
up[up < 0] = 0
down[down > 0] = 0
df_final['Upward Movement'] = up
df_final['Downward Movement'] = abs(down)
df_14days_Up = df_final['Upward Movement'].rolling(14).mean() # using 14 day
df_14days_down = df_final['Downward Movement'].rolling(14).mean() # using 14
df_final['Avg. 14-Day Up Closes']=df_14days_Up
df_final['Avg. 14-Day Down Closes']=df_14days_down
df_RelativeStrength = df_final['Avg. 14-Day Up Closes'] / df_final['Avg. 14-L
df_final['Relative Strength'] = df_RelativeStrength
df_RSI = 100 - (100/(1+df_final['Relative Strength']   ))
```

Appendix 5: Sample 17 days Actual vs Predicted forex price for CZK/Euro Using LSTM

Actual - NumPy object

	0
0	26.061
1	26.055
2	26.009
3	25.914
4	25.87
5	25.861
6	25.827
7	25.747
8	25.653
9	25.715
10	25.756
11	25.725
12	25.699
13	25.656
14	25.787
15	25.818
16	25.883
17	25.875

final - NumPy object array

	0
0	26.3249
1	26.3935
2	26.4902
3	26.6055
4	26.4327
5	26.3323
6	26.3951
7	26.4947
8	26.505
9	26.0629
10	26.1462
11	26.3453
12	26.3192
13	26.3548
14	25.8778
15	26.1979
16	26.1105
17	26.3226