Currency Exchange Rate Forecasting: A Text-mining Approach

Zanele Khumalo

1043606

Supervisor:

Mr. Rendani Mbuvha

A research report submitted in partial fulfillment of the requirements
for the degree of Master of Science in the field of e-Science

in the

School of Computer Science and Applied Mathematics
University of the Witwatersrand, Johannesburg

29 May 2020

Declaration

I, Zanele Khumalo, declare that this research report is my own work. It is being submitted for the degree of Master of Science in the field of e-Science at the University of the Witwatersrand, Johannesburg. It has not been submitted for any degree or examination at any other university. I have acknowledged all sources used and have cited these in the reference section.

Khumalo

Zanele Khumalo

1043606

29 May 2020

Abstract

The currency market deals with all aspects of buying, selling and hedging currencies. Financial markets are a complicated system and are difficult to model due to structural instabilities and noise driven by different factors such as economic conditions, investor behaviour, and politics. Investors are required to be abreast of the latest economic news while also synthesizing historical market performance in order to come up with investment strategies that minimise risks and maximise profits. With the rapid growth of digitisation of news articles, analysis of such information can be difficult due to the volume and variety of contents; therefore, requiring automation. Text mining approaches can assist in automating the process of extracting useful information from multiple news article contents and possibly improve market predictions when combined with historical market prices.

In this thesis, we investigate whether information derived from news articles adds statistically significant predictive power in exchange rate market forecasting. This is done by comparing the performance of the ARIMA, SVR and LSTM in forecasting the closing prices of the USD/ZAR currency pair. Furthermore, we investigate the effect of Reddit news headlines and Reuters news article contents on the prediction of the USD/ZAR currency pair, by using Latent Dirichlet Allocation (LDA), to extract topics from raw text documents and using these as additional input features to LSTM and SVR models. The results show that the traditional ARIMA outperforms the SVR and LSTM models in forecasting the USD/ZAR closing prices, however, the additional news features can yield statistically significant improvements in performances of SVR models when forecasting the daily USD/ZAR closing prices. While marginally improving LSTM models. The performances of improved SVR and LSTM models show no statistically significant differences when compared to ARIMA models.

Acknowledgements

I would like to thank the "**DST-CSIR National e-Science Postgraduate Teaching and Training Platform (NEPTTP)**", for financial support. In addition, I wish to extend my special thanks to my supervisor Mr. Rendani Mbuvha for his invaluable contribution and in aiding this research project.

Contents

List of Figures

List of Tables

Acronyms

Adam Adaptive moment estimation.

ANN Artificial Neural Network.

ARIMA Autoregressive Integrated Moving Average.

CHAID Chi-squared Automatic Interaction Detection.

EUR Euro.

GDA gaussian discriminant analysis.

GRU Gated Recurrent Unit.

HMM Hidden Markov Model.

JPY Japannese Yen.

LDA Latent Dirichlet Allocation.

LR Logistic Regression.

LSTM Long Short-Term Memory.

MAE Mean Absolute Error.

NB Naive Bayes.

NLP Natural Language Processing.

RBF Radial Basis Function.

RF Random Forest.

RMSE Root Mean Square Error.

RMSProp Root mean square propagation.

RNN Recurrent Neural Network.

SGD Stochastic gradient descent.

SVM Support Vector Machine.

TSE Tehran Stock Exchange.

TSLDA Topic Sentiment Latent Dirichlet Allocation.

USD United States Dollar.

1 Introduction

1.1 Background

Exchange rates are a measure of one country's currency with reference to another and are determined in the foreign exchange market or the currency market. The currency market is a platform that deals with all aspects of buying, selling and hedging currencies [45]. Financial markets are a complicated system and are difficult to model due to structural instabilities and noise driven by various elements such as economic factors, investors behaviour, and politics [58]. Exchange rates prediction is one of the most laborious tasks in the financial services industry, and thus it is of crucial importance [45] [58]. Investors, importers and exporters make decisions to buy or sell a currency based on the value and its volatility. In an attempt to minimize risks and maximize returns, practitioners and academics have dedicated significant efforts to investigate the currency market to comprehend its influential components, performance techniques and features [45][58] [21].

There are two main approaches used to analyze the currency market, particularly, the fundamental and technical analysis [57][21]. Fundamental analysis is a technique used to assess exchange rates based on basic economic factors extracted from economic models [21] [58]. It takes into consideration the assumption that exchange rates have a tendency to revert to their real value over time [21] [58] [41]. Due to its robustness in deducing the basis forecasts, this method commonly used for long-term investment strategies [41][58]. In contrast, the technical analysis method pays particular attention to the features of market movements, through the analysis of historical data [41] [58]. This method is particularly reactive to market movements because its strategies are mostly built based on popular models, and for this reason, it is used in short-term trading [58].

With recent notable developments of computational intelligence theory and methodology, and increasing complexities in data generated, more techniques have emerged that focus on the analysis and prediction of exchange rates. From a machine learning and artificial intelligence perspective three main data sources are utilized, historical time series data as used in technical analysis, semantic components and sentiment data extracted from political and financial news articles as used in fundamental analysis [56]. In such scenarios, the frontier between technical and fundamental analysis is likely to be unclear. Thus, the resulting model cannot be grouped as technical or fundamental, but there is a need to assess which fundamental or technical features should be incorporated in order to find the most appropriate model [58].

The use of textual information in financial time series modelling has long been the tradition of the trading practice. In order to make beneficial decisions, investors are required to carefully read relevant financial and economic news, study market trends and political events that influence markets. With the rise in the volume of internet usage, there has been an increasing number of financial reports and news articles. Therefore it is of crucial importance to automate this process [41]. The idea of analyzing textual information for financial forecasting dates back to the 1980s, however automating this process has made little progress throughout the literature for various reasons [56]. This research report uses text mining techniques in financial time series prediction.

1.2 Study Aims and Objectives

This research aims to investigate whether news article headlines or contents can be used as one of the predictors for the exchange rate market. The hypothesis is that contents of news articles contain vital information for assessing the performance of the currency market in a short term. We evaluate and compare the predictive performance of ARIMA, SVR and LSTM on the historical USD/ZAR closing prices, and further investigate whether topics contained by news article headlines and contents can improve the predictive performance of the LSTM and SVR models.

1.2.1 Problem Statement

Analysts are required to read latest financial and economic news and study historical market performance in order to come up with investment strategies that minimise risks and maximises profits. The task its self can be difficult and time consuming for humans as there has been a rapid growth of digitization of news articles. Analysis of such information is beyond human capabilities and therefore requires automation. A vast number of tools have been developed for analysing historical market prices in the literature, however research on the use of textual data in this field has been of limited success. Text mining approaches can assist in automating the process of extracting useful information from multiple news article contents and possibly improve market predictions when combined with historical market prices.

Mo, Liu, and Yang [34] shows that there is strong correlation between news sentiment and market returns this validates that analyzing news articles can be beneficial for market prediction. In overall, we aim to elucidate the following research questions:

- What fundamental topics in the news articles influence future movements of the South African currency market?
- How financial news articles influence the predictive performance when predicting foreign exchange rate market movements?

1.3 Contributions Of The Study

Although financial time series forecasting has been a popular topic in academia, research on forecasting emerging markets such as the South African Rand (ZAR) against other currencies from the machine learning and NLP perspective has been of limited success. Consequently, it stands to reason that there exist abundant opportunities to better understand the drivers of the South African currency market. Specifically, this research provides insights on the effect of news article headlines and contents on the USD/ZAR pair. We also demonstrate differences in the performance of the traditional time series forecasting technique (i.e ARIMA) and machine learning techniques (i.e SVR and LSTM) for this task. We identify top topics from Reddit news headlines and Reuters news article contents using an unsupervised topic clustering technique LDA and our result shows that these can have positive predictive power in forecasting the USD/ZAR pair. Therefore, this research presents text mining as a new approach to improving predictive performance in exchange rate modelling.

1.4 Structure Of This Research Report

The rest of this report is organised as follows: Chapter 2 contains "literature review" on financial time series modeling, chapter 3 provides methodology and experimental settings followed in this research, chapter 4 provides experimental results and discussions, and chapter 5 concludes.

2 Literature Review

Predictability of Financial Markets

Predictability of financial markets is one of the most important and attractive topics for researchers and investors. Financial forecasting covers prediction of key indicators, such as volatility, price and volume in the currency exchange or stock markets. Ability to accurately predict financial markets can support investors in creating beneficial investment strategies [57]. With regards to predicting financial markets, various theories are available. The well known efficient market hypothesis (EMH) theory introduced by Fama [14] is the most prevailing theory in market prediction. The EMH states that financial markets mirror all available information and always trade at a fair value [50] [34]. The EMH theory is further broken down into three types: weak, semi-strong and strong. In weak EMH, market prices are immersed with historical information. The semi-strong EMH progresses by incorporating both history and publicly available current affairs in market prices. Strong EMH advances from semi-strong EMH by also incorporating private information such as internal news in market prices [50]. From the principles of EMH, markets are believed to react promptly to news releases and general political events and thus are completely unpredictable [14] [50]. The random walk theory also proclaims that despite the availability of information, predicting financial markets is completely impossible [32].

2.1 Financial Forecasting Techniques

This section gives a background on previous work on financial time series forecasting. There are a number of technical approaches that have been successful in literature. These can be separated into two types, namely, traditional time series and machine learning methods [57] [3].

2.1.1 Traditional Time Series Methods

Time series analysis methods can be grouped into two types, namely, univariate and multivariate models [52]. Univariate time series models analyze a series of a single variable recorded sequentially over time e.g, daily historical stock prices [52]. A popular example of a univariate time series method is the autoregressive integrated moving average (ARIMA) model. Traditional time series approaches are linear models and are often incapable of forecasting financial markets due to the complex nature of the market. However, ARIMA models have been demonstrated as a powerful tool for generating short term market forecasts in comparison to other favoured methods such as artificial neural networks (ANN) [3].

Academics have developed numerous modifications to ARIMA models to account for the complexity of the market and improve predictive accuracy [52]. Multivariate time series models are a natural expansion of univariate models. They are commonly used to investigate relationships between historical market prices and other financial indicators [52]. Multivariate time series techniques involve linear regression and GARCH [3] [55]. Most traditional approaches rely on strong assumptions regarding the structure of the time series that is, data are assumed to be generated from stationary stochastic process, which may not be realistic for market data [52]. However, GARCH models propose a process for analyzing dynamic stochastic variances. In this way, GARCH models are effective in modelling market risk [52]. The linear structure and strict assumptions that are required by traditional time series models restricts them from successfully modeling the underlying performance of the markets, which is their main drawback [58].

2.1.2 Machine Learning Methods

Support Vector Machine (SVM) and ANN are established machine learning models for modelling financial markets [53]. With their flexibility and ability to effectively model complex patterns between inputs and outputs, they both offer a great alternative to traditional time series approaches for financial market prediction [58]. Previous researchers have shown that the ANN performs better than traditional forecasting methods [19]; [24]. Hsu, Tse, and Wu [19] applied both ANN and ARIMA to forecast the stock index. This demonstrates competitive abilities of the ARIMA model and the ANN model in financial forecasting [19]. ANN models are built based on empirical risk minimization principles. Local minimum problems, difficulties in determining the learning rate, and overfitting issues limit the application of ANN-based models in financial forecasting [52]. However, SVM uses kernel functions and has a global optimum. Therefore, it achieves better generalization error compared to ANN [52]. A study regarding the use of SVM in financial forecasting is presented by Sheta, Ahmed, and Faris [53]. The study compares Regression, ANN and SVM to model the S&P 500 stock index. The SVM outperforms regression and ANN models. Similary, Liu et al. [28] investigated Logistic Regression (LR), gaussian discriminant analysis (GDA), naive bayes (NB) and SVM in forecasting the S&P 500 market index. The result shows superior performance on the SVM model with the Radial Basis Function (RBF) kernel. Kara, Boyacioglu, and Baykan [24] applied both ANN and SVM in prediction of stock market movements. Experimental results shows superior performance on ANN based model compared to the SVM model. Xin-Yao and Shan [55] compared performances the of ARIMA, LR and SVM for stock index forecasting, and the SVM demonstrated superior performance compared to other models.

Additional neural network based techniques such as "recurrent neural networks (RNN)" have also been prevalent in financial sequence modelling. The RNN has shown superior ability to learn complex temporal patterns compared to feed forward neural networks [13]. Particularly, the LSTM RNN has been gaining popularity in sequence learning due to its success in diverse applications being natural language processing, image processing and speech recognition [25]. [30] uses RNNs to forecast the daily USD/JPY currency pair. The result shows that RNN are effective in modelling non linear time series. Di Persio and Honchar [13] presents a study that RNN implements for financial time series forecasting. Where different types of RNN archetectures are compared i.e. the "basic multilayer RNN, Long Short-Term Memory (LSTM) and gated recurrent unit (GRU)". Their results show that the LSTMs approach performs better than other RNN based architectures. Which demonstrates the superiority of the LSTM in modelling sequential data

Tree based approaches have also been successfully applied to forecast a wide range of financial time series [26]. These approaches are generally applicable for classification and regression tasks. A decision tree is a type of learning model that predicts output values by applying different decision rules based on its input features [1]. Different examples of decision tree based approaches such as decision trees, random forests (RF), and Chi-squared automatic interaction detection (CHAID) have been applied in financial forecasting. Like other SVM and ANN, tree based methods provide reasonable alternatives to traditional techniques that are often limited by strong assumptions in modelling financial markets. Researchers use decision trees because they are simple and interpretable. However, similar to ANN, decision trees suffer from overfitting issues. RF provides a solution to improve the predictive performance of decision trees and avoid overfitting [26].

One of the studies with reference to the utilization of a tree based method for financial prediction is presented by Patel et al. [43]. The study compares a random forest model with ANN, SVM and NB in stock market prediction. The results show that the RF model outperforms ANN, SVM and NB. Aggarwal et al. [1] presented a CHAID model to estimate the volatility of the Indian stock market. CHAID is a method based on adjusted significance testing, therefore it has an ability to provide information about features interactions [1]. Similary, Imandoust and Bolandraftar [20] applied "RF, decision trees, and NB" to predict daily TSE currency prices. The decision tree performed significantly better than RF and NB models [20]

The application of HMM has also been prominent in financial time series forecasting. HMM are based Markov chains theory and are known to be substantial for time series data modelling [16]. They have been widely applied in different areas such as pattern and speech recognition, DNA sequencing, and image processing [17]. An experiment presented by [39] used HMM to predict the S&P 500 stock market index. In a similar fashion, [38] also applied the HMM to forecast daily stock prices of Apple, Google, and Facebook. Again, [27] used linear regression line as a feature extraction method and then applied HMM in of the foreign exchange rate overtime. The HMM model presented promising result in short term trading.

2.1.3 Text Mining for Financial Time Series Prediction

Market movements are strongly driven by the availability of new information, vast amounts of information are streaming on digital platforms. With different types of information including political news, financial news articles are considered to be the most consistent source for researchers and investors alike [50] [51]. A variety of methods are available to analyze financial news articles. There is often a need to pre-process textual data by extracting meaningful features before further analysis. These features are extracted using sentiment analysis or semantics modeling. Both these feature representation methods employ computational linguistics techniques such as natural language processing (NLP) [56].

A commonly used methods for textual feature extraction is the bag of words approach, which formulates a vector representation (vectorization) of a text field based on a set of words and the frequency of their appearance [56]. This approach, often requires the removal of stop words such as "is", "the", "a", etc. The bag of words method disregards word order and cannot capture similarity in different phrasing, which is the obvious drawback of this method [56]. These issues are solved by examining textual semantics in documents, traditionally n-gram based approaches are used to achieve this task [56] [5].

With recent developments in computational intelligence theory, more semantic similarity estimation or word embedding approaches have emerged. Deep learning has been one of the successful methods in changing the way text vectorization is done and finding better ways to represent text [56]. Semantic textual similarity is relevant for different tasks including machine translation and question answering [31]. Advancing from word embedding, topic models are used to capture semantics at a document level, allowing an analysis of large volumes of financial articles [5]. Topic models define documents as mixtures of various topics. Feature extraction is performed based on topic distributions within documents [40]. Peramunetilleke and Wong [45] were one of the first research papers to use news data to forecast exchange rates. Their research used news headlines as inputs and the model performed significantly better than random prediction [45].

Nguyen and Shirai [40] and Jin et al. [23] are two studies that apply topic modelling based approaches to analyze textual information for financial forecasting. Jin et al. [23] used "Latent Dirichlet Allocation (LDA)" which was initially proposed by Blei, Ng, and Jordan [6] for feature extraction. LDA was used to classify news articles into different topics and obtain each article’s topic distribution which were further used for sentiment analysis. Nguyen and Shirai [40] proposed a novel approach to topic modeling Topic Sentiment Latent Dirichlet Allocation (TSLDA). TSLDA is an extended version of LDA that captures topics and their sentiments simultaneously in documents [40].

Sentiment analysis has became a popular forecasting tool in finance, a lot of work has been done in this field for stock market prediction. However, the use of sentiment analysis of news articles in foreign exchange rate prediction has been limited in literature. Nassirtoussi et al. [36] presents a novel approach to foreign exchange prediction that incorporates financial news sentiment analysis. Their work focuses on tackling problems associated with high dimensionality of the data and that of semantic similarity estimation in textual data. This method performs significantly better reaching 83% accuracy [36]. Jin et al. [23] performs sentiment analysis based on topic distributions per document as produced by LDA model. This was done through identification of top topics (i.e increase, decrease) by combining news articles with currency trends in the training data and later using linear regression for sentiment prediction [23]. The result successfully demonstrates a predictive relationship between the foreign exchange market and the financial news data. Other various techniques have been used for sentiment analysis such as Naive Bayes, Maximum Entropy and SVM [42].

Forecasting exchange rates has been a popular topic for the world's most traded currencies such as the United States Dollar (USD), Euro (EUR) and the Japannese Yen (JPY). Research on forecasting emerging markets such as the South African Rand (ZAR) against other currencies from the machine learning perspective has been limited. There are also limitations on literature that uses machine Learning and NLP approaches for this task. More work needs to be done in the area to find techniques that can effectively identify correlations between news articles and financial markets. Thus, this research focuses on analyzing the South African currency market in this regard.

3 Methodology

3.1 Research design

Due to the complex nature of the currency markets, forecasting accuracy using text data is usually low in existing literature, despite the application of state of the art machine learning approaches. The ARIMA, SVR and the LSTM have been demonstrated as efficient tools for sequential data modeling [3] [13] [28], therefore, we explore their performances in forecasting the USD/ZAR closing prices. Furthermore, features derived by text mining of news headlines and contents are incorporated in the SVR and the LSTM models. To analyse the effectiveness of textual features, we follow two types of approaches. Firstly, we build forecasting models using historical prices only and then build the same models while incorporating textual features.

3.2 Methods

3.2.1 ARIMA Models

The mathematical formulation of the ARIMA model can be presented as follows [3]:

$$Y_t = \phi_0 + \sum_{i=1}^{p} \phi_i Y_{t-i} + \varepsilon_t + \sum_{j=1}^{q} \theta_j \varepsilon_{t-j} \tag{3.1}$$

Where,

- Y_t is the actual price at t,
- ε_t is the error at t,
- ϕ_i, θ_j are coefficients,
- p and q are integers (often called AR and MA orders respectively)

3.2.2 Support Vector Regression

SVR is a method based on SVM suitable for continuous outputs [47]. The mathematical formulation can be presented as follows [8]:

Given data with n dimensional input features x_i and one output vector y_i in the form of $S = \{(x_i, y_i)\}_{i=1}^{N}$. The aim is to find a function that effectively maps the original dataset into a higher-dimensional space, that is, solving equation 3.2:

$$h(x) = \sum_{i} \alpha_i y_i (x_i^T x_i) + b \tag{3.2}$$

Where, x_i and y_i are support vectors with α_i and b coefficients. The coefficients can be approximated by minimizing the following standardized function:

$$R(C) = C\frac{1}{N}\sum_{i}^{N} \mathbf{L}_{\epsilon}(y_i, h(x_i)) + \frac{1}{2}\|\mathbf{w}\|^2 \tag{3.3}$$

$$\mathbf{L}_{\epsilon}(y, h(x)) = \begin{cases} |y - h(x)| - \epsilon, & \text{if } y - h(x) \geq \epsilon, \\ 0\,, & \text{otherwise.} \end{cases} \tag{3.4}$$

Where:

- ϵ threshold out of sample error,
- $L_{\epsilon}(y, h(x))$ is a *linear loss function* [7]; [47],
- $\frac{1}{2}\|w\|^2$ in equation 3.3 is used as a measure of flatness, and
- C is a regularization parameter for determining the trade-off between the training error and model flatness.

We can introduce slack variables ζ and ζ^* and consequently we
Minimize:

$$min\left[\|\mathbf{w}\|^2 + C^* \sum_{i}^{N}(\zeta_i, \zeta_i^*)\right] \tag{3.5}$$

Subject to:

$$\begin{aligned} y_i - \mathbf{w} \cdot \mathbf{x}_i - b &\leq \epsilon + \zeta_i \\ (\mathbf{w} \cdot \mathbf{x}_i + b) - y_i &\leq \epsilon + \zeta_i^* \end{aligned} \tag{3.6}$$

With $\zeta, \zeta^* \geq 0$. substituting equation 3.2 results in the following objective function.

$$f(x, \alpha, \alpha^*) = \sum_{i=1}^{N} (\alpha_i - \alpha_i^*) K(x, x_i) + b \tag{3.7}$$

Where $K(x, x_i)$ is a"Kernel function, α_i and α_i^* are *"Lagrange multipliers"* that satisfy the following:

- $\alpha_i \times \alpha_i^* = 0$
- $\alpha_i, \alpha_i^* \geq 0 \; i = 1, ..., N$

These are Lagrange multipliers are obtained by maximizing:

$$W(\alpha, \alpha^*) = \sum_{i=1}^{N} y_i(\alpha_i - \alpha_i^*) - \epsilon \sum_{i=1}^{N} y_i(\alpha_i + \alpha_i^*) - \sum_{i,j=1}^{N} y_i(\alpha_i - \alpha_j^*) K(x_i, x_j) \tag{3.8}$$

With the following constrains

- $\sum_{i=1}^{N} \alpha_i = \sum_{i=1}^{N} \alpha_i^*$
- $0 \leq \alpha_i \leq C$ for $i = 1, ..., N$
- $0 \leq \alpha_i^* \leq C$ for $i = 1, ..., N$

Finally, the process of training the SVM is similar to optimizing the Lagrange multipliers with equation 3.8 inequality constraints [47]. Different types of kernel functions can be used in this process, namely, the linear, polynomial, radial basis function (RBF), and sigmoid kernel. With the following mathematical formulations.

Linear,

$$K(x_i, x_j) = {x_i}^T x_j \tag{3.9}$$

Polynomial,

$$K(x_i, x_j) = (1 + x_i \cdot x_j)^d, \tag{3.10}$$

RBF,

$$K(x_i, x_j) = exp(-\gamma \|x_i - x_j\|^2), \tag{3.11}$$

Sigmoid,

$$K(x_i, x_j) = \left[v(x_i, x_j) + \alpha\right], \tag{3.12}$$

Where,

- d in equation 3.9 is the polynomial degree
- γ in equation 3.11 the kernel function parameter.

3.2.3 Long Short Term Memory

LSTM's are a particular type of an RNN introduced by Hochreiter and Schmidhuber [18] as a solution to RNN's short-term memory problem. The LSTM architecture is comprised of a set of recurrently connected memory blocks. Each of these blocks contain one or more self-connected memory cells and three multiplicative units that provide signals of write read and reset operations for the input, output and forget gates [25]. The multiplicative gates permit memory cells to reserve and ingress information over long time periods, that way overcoming the vanishing gradient problem. The LSTM uses the cell state C_t as a conveyor belt that simplifies information flow across the network unchanged. The network can append or discard information from the cell state, and cell gates control this. Initially the LSTM has to decide what information shall remain or be omitted in the cell state, the forget sigmoid layer known as the forget gate layer enables this by taking the input X_t, and the previous hidden state h_{t-1} for each value in the cell state C_{t-1}. At time t the output of the cell state C_t is computed as:

$$f_t = \sigma(W_f \cdot [h_{t-1}, X_t] + b_f) \tag{3.13}$$

An additional step is to choose the type of information that is transferred in the cell state for the output h_t. This is done by using explicit gating mechanisms, where the sigmoid layer called the input gate layer decides which values will be updated and the tanh layer establishes a vector of new possibles values, $\hat{C}_t$, that can be appended to the state. The formulation for cell state updates is defined as:

$$\begin{aligned} i_t &= \sigma(W_i \cdot [h_{t-1}, X_t] + b_i) \\ \hat{C}_t &= tanh(W_c \cdot [h_{t-1}, X_t] + b_c) \end{aligned} \tag{3.14}$$

$$C_t = f_t * C_{t-1} + i_t * \hat{C}_t \tag{3.15}$$

Equation 3.15 is applied to update the past cell state C_{t-1} into the new cell state C_t. Where σ is the sigmoid activation function defined in equation 3.16, *tanh* is the hyperbolic tangent function given by equation 3.17, and the current hidden state h_t is obtained by multiplying an element-wise *tanh* of the current cell state i.e o_t with the output gate as shown by equation 3.18. Vectors W, b and $o \in 1, ...,$ are binary gates that control whether each C_t is updated, whether it is reset to zero, and whether its state is revealed in the hidden vector respectively.

$$\sigma(x) = \frac{1}{1 + e^{-x}} \tag{3.16}$$

$$tanh(x) = \frac{e^x - e^{-x}}{e^x - e^{-x}} \tag{3.17}$$

$$\begin{aligned} o_t &= \sigma(W_o[h_{t-1}, X_t] + b_o) \\ h_t &= o_t * tanh(C_t) \end{aligned} \tag{3.18}$$

3.2.4 Latent Dirichlet Allocation

LDA is an unsupervised probabilistic model concerned with understanding topics across documents. It was initially introduced by Blei, Ng, and Jordan [6] and it has been demonstrated as an efficient tool for topic modeling throughout literature. In LDA each document is assumed to be a mixture of different topics. Given a corpus D consisting of M documents. Each document d consists of N_d words for $d \in 1, ..., M$, LDA models the corpus by the following generative process [6]:

1. Choose a topic mixture or multinomial distribution ϕ_t for topic t ($t \in 1, ..., N$) from a "Dirichlet distribution" with parameter β

2. Choose a "multinomial distribution" θ_d for each document d in the corpus from a Dirichlet distribution with parameter α

3. For each word W_n where $n \in 1, ..., N_d$ in document d:

   (a) Select a topic z_n from θ_d

   (b) Select a word W_n from ϕ_{zn}

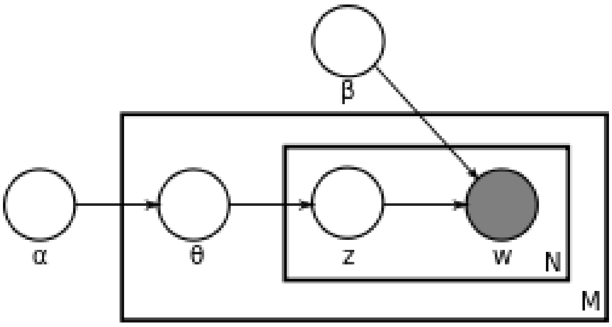


FIGURE 3.1: Plate Notation [6]

Figure 3.3.3.2 and the above generative process explain the relationships between observed variables (words) and unobserved variables ϕ and θ [22]. The probability of observed variables and the Dirichlet prior parameters α and β can be computed as follows:

$$P(D|\alpha, \beta) = \prod_{d=1}^{M} \int P(\theta_d|)(\prod_{z_{dn}}^{N_d} P(z_{dn}|\theta_d)P(W_{dn}|z_{dn}, \beta))d\theta_d \tag{3.19}$$

Finally, LDA treats documents as probability distributions over latent topics, where each topic is treated as a probability distribution over words. The corpus is produced by a generative process that is defined by the joint probability distribution over observed and unobserved variables as in equation 3.19.

3.3 Experimental Design And Analysis

This research focuses on the prediction of the daily USD/ZAR currency pair using both news articles and historical prices. In order to address the research questions described in section 1.2, we perform the following experiments:

1. Performance comparison of ARIMA, SVR and LSTM in forecasting the closing prices of the USD/ZAR pair for two separate periods.

2. Building LDA-based topic models as follows:

   - We build a separate topic model for each news data sets (Reddit headlines and Reuters news contents) and obtained their topic distributions.
   - Aligning topic distributions with historical currency data

3. Performance comparison of SVR and LSTM in forecasting the currency closing prices using historical currency prices and topic distributions as input features.

The detailed experimental process is provided in sections 3.3.2 and 3.3.3.

3.3.1 Data Description

The study presented in this research report focuses on the analysis of the daily USD/ZAR currency pair using both news articles and historical prices. We collected two types of data sets for the development of our models, namely, historical exchange rates and general news article headlines and contents.

3.3.1.1 News Article datasets

To obtain relevant news articles for our experiments we collected data from a public source published by Kaggle [54] and the Reuters news website. The kaggle news headline data set was sourced from the Reddit World News Channel for the period of approximately 8 years (08-August-2008 to 01-July-2016), with 25 news headlines for each day [54]. The second news data set was collected using News API [37] from Reuters.com. Reuters news are published on a daily basis and gives detailed discussions on the day's economic events, politics, entertainment and technology. Reuters news were collected daily for the period of 30-July-2018 to 08-August-2019, with 10 articles for each day. These data sets were collected separately due to data availability, Reuters news provides both news headlines and contents, and for this reason, we decided to explore the effect of both headlines and contents separately. We use the 8 year period of data to analyse the effect of news headlines and we denote that experimental process as *experiment 1*. The 1 year period of data is used to explore the effect of news contents and is denoted by *experiment 2*.

We split both data sets into training and testing set in order to evaluate the model's ability to generalise on unseen data. The training data points for experiment 1 cover the time period from 08-August-2008 to up to 31-July-2015 having a total of 1821 observations, while observations from 01-Jun-2015 up to 01-July-2016 are used for testing which has total of 240 observations. For experiment 2, the training set is comprises of observations from the 30-July-2018 to the 31-May-2019 with a total of 231 observations, and the testing set is comprises of observations from 01-June-2019 to 08-August-2019 with 49 observations.

3.3.1.2 Historical Foreign Exchange Rate Data

We collected historical USD/ZAR prices from a global financial portal (Investing.com). The historical price data set is comprised of closing, opening, high, low price, and percentage change attributes and was collected daily for two separate periods. The first period is from 08-August-2008 to 01-July-2016, which is approximately eight years of data, with 2061 observations. The second period is from 30-July-2018 to 08-August-2019, which is approximately one year of daily data, with 291 observations. Both data sets are exclusive of weekends and as they are not considered as valid trading days. The original closing prices for experiment 1 and 2 are shown in figures 3.2 and 3.3 respectively.

Dataset	Period	# obs.	# news articles
Experiment 1	08-Aug-2008 to 01-Jul-2016	2061	51525
Experiment 2	30-Jul-2018 to 08-Aug-2019	291	7275

TABLE 3.1: Data description

3.3.2 Historical price-based approach

In this part we only use historical prices to build our forecasting models. The purpose of this experiment is to explore patterns within the USD/ZAR historical prices, and use these models as a baseline for evaluation of the effectiveness of news article features. Sections 3.3.2.1, 3.3.2.3 and 3.3.2.4 describe processes we followed when implementing the ARIMA, SVR and LSTM respectively.

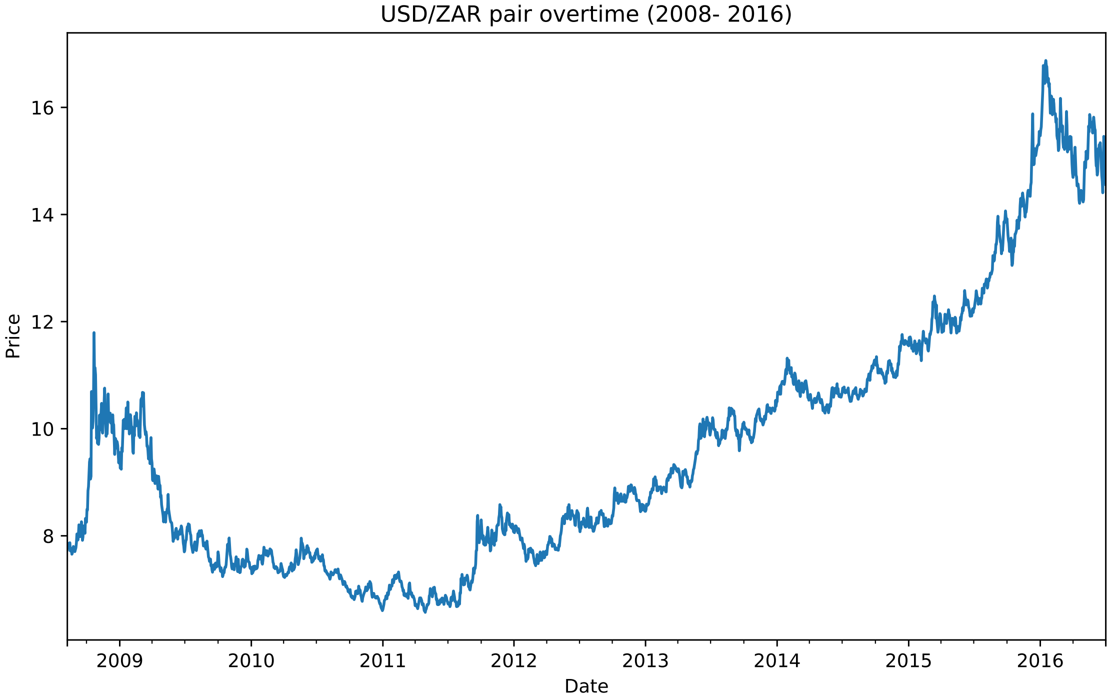


FIGURE 3.2: Original daily USD/ZAR closing prices for experiment 1

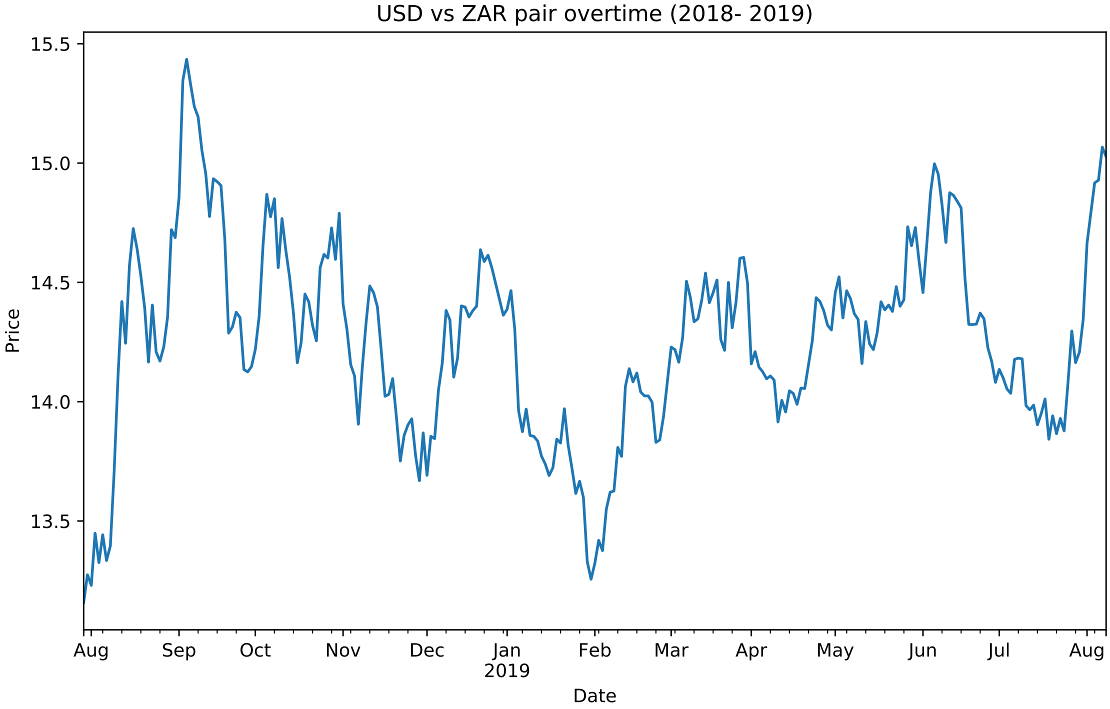


FIGURE 3.3: Original daily USD/ZAR closing prices for experiment 2

3.3.2.1 ARIMA Implementation

ARIMA models are developed by utilizing the Box-Jenkins technique through the following repetitive steps:

1. *Model identification:* in this step we use autocorrelation analysis and partial autocorrelation analysis to investigate any trends, stationarity, and seasonality patterns on the time series data. The ARIMA model expects stationary time series as an input. Therefore, a time series that is not stationary, requires data transformations in order to induce stationarity.

2. *Model parameter estimation:* Once necessary data transformations are done, the subsequent step is to estimate model coefficients in order to determine the best suitable model for the dataset. To achieve this, we used the *statsmodels* module to build train the ARIMA model while exploring different parameter combinations of p, q and q. Reasonable models were determined by comparing different outputs of RMSE and AIC values and smaller values indicate a better model.

3. *Model diagnostics:* In this step, we evaluate the reasonable model for adequacy based on the stationarity premise and decide whether it can be used to generate forecasts.

We followed the above steps to develop and train ARIMA models using the USD/ZAR closing prices for experiment 1 and experiment 2. Figure 3.4 shows ACF and PACF plots for experiment 1 dataset. The ACF plot is slowly dying down and the ACF shows that the is a time dependent structure at lags 1 and 3, these both show that the series for *experiment 1* is not stationary. Similarly, figure 3.5 shows the ACF and PACF plots for *experiment 1*. These plots suggests that both dataset require transformations to induce stationarity at, which means, the parameter one level of differencing is required in order to build appropriate ARIMA models for both experiment 1 and 2. The next step is to determine parameters p and q respectively. These were determined by exploring a range of different values of p, d and q using gridsearch. These final models were then used to forecast their corresponding testing sets and their performances were measured as per section 3.3.4

3.3.2.2 Data Preprocessing

The SVR and LSTM are both supervised learning techniques that map inputs to outputs based on given input and output pairs. Currency data do not posses this structure, therefor data transformation is required before applying these models. The following data transformations are performed on the datasets prior to fitting models and making forecasts.

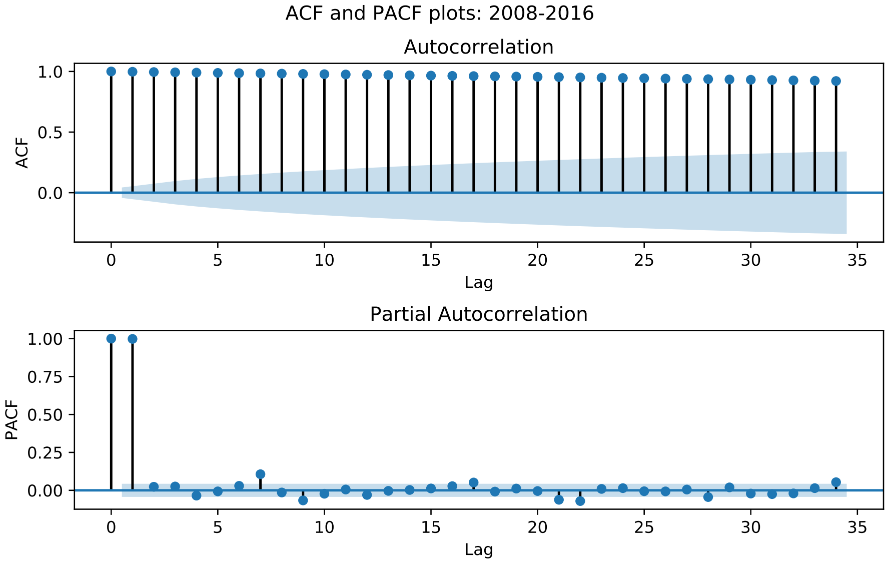


FIGURE 3.4: ACF and PACF for experiment 1

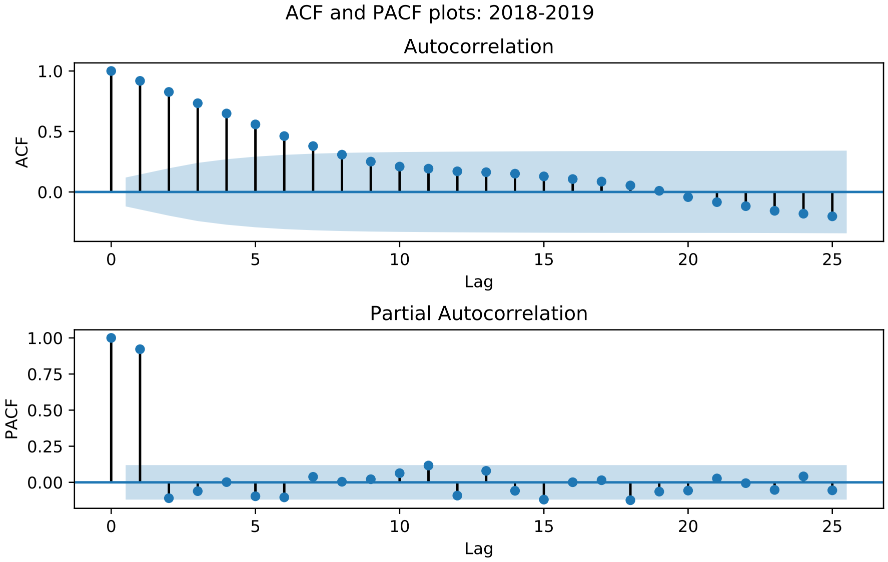


FIGURE 3.5: ACF and PACF for experiment 2

1. Transforming the time series into a supervised machine learning problem, specifically, each data set was transformed into its inputs and output labels by using observations of previous time events as inputs to forecast the current time event. That is, forecasting the closing price at time (t) using values of closing, opening, high, and low price at times $(t-1), (t-2), ..., (t-n)$ where n is the number of lags or previous time steps.

2. Feature Scalling: all features in the datasets were converted to their $z - scores$, where $z = \frac{x-\mu}{\sigma}$, μ is the mean and σ is the standard deviation of each feature. Feature scaling is one of the most critical pre-processing steps in machine learning to ensure that all

features are of the same scale before fitting models.

3.3.2.3 SVR Implementation

After data preprocessing, SVR models were developed using the *Scikit-learn* package in Python [44]. When fitting a supervised learning model we use the training set to build the model and then evaluate its performance using the testing set. One of the most commonly used procedure for model evaluation in machine learning is cross validation. This method is applicable for independent observations, however, when it comes to sequential observations it has been demonstrated likely to overfit [48]. Therefore, we apply the modified cross-validation technique introduced by Chu and Marron [12]. To do this, we further split the training set into multiple segments, with each segment having the training and the validation set. We employ the $TimeSeriesSplit()$ function to split the training set into 5 segments. This function allows time based data splits for dependent observations. We then used the $gridsearchCV()$ function to estimate and tune parameters the models by measuring the R^2 value from training and evaluation sets. R^2 is generally used to compare the degree of variation between actual and fitted values for regression problems and is, therefore, a suitable measure to determine the optimal model. Particularly, possible combinations of values for C, ϵ, $gamma$ and the kernel function were explored as per table 3.2. Finally, we evaluated the performances of the best models by forecasting their respective testing data sets. Transformations were inverted on forecasts to return them into their original scales before calculating model performance measures as per section 3.3.4.

Parameter	*Range*
C	(0, 10000]
ϵ	(0.0001, 0.1]
γ	(0.05, 0.5]
Kernel	linear, RBF, Sigmoid

TABLE 3.2: Parameter ranges for SVR models

3.3.2.4 LSTM Implementation

The LSTM network models were implemented by using the Keras deep learning package in python [11]. The LSTM network requires data to be presented in a three dimensional array format, where the first feature denotes the batch size, the second feature denotes the time-steps and the third feature denotes the number of units in one input sequence. Once the dataset is reshaped, we fit an LSTM network using a keras sequential model. The training stage of building the LSTM network involves the tuning of hyperparameters before selecting a reasonable model. Some of the hyperparameters that can be tuned are the number of hidden layers, number of neurons per layer, learning rate, drop out rate, batch size, number of epochs, and optimizers. We evaluated the reasonable model for adequacy by plotting the validation loss versus the training loss at each epoch.

We construct the models that have one LSTM input layer each with 50 neurons. We added two hidden layers with 50 neurons each and RELU activation functions, separated by a single drop out layer with a dropout rates ranging from 0.1 to 0.5. The drop out layer prevents overfitting by temporarily removing a ratio of neurons from the network, and its adjacent incoming and outgoing connections during model training. Finally, a dense output layer with a single neuron and RMSProp optimizer was added. We search the space of parameters as follows. The LSTM networks are trained with 16-80 neurons on the LSTM layer, 50 to 1000 epochs, and a batch size range between 32 to 250.

3.3.3 Text-mining based approach

This section describes the steps taken when obtaining news features from raw text documents and using them in forecasting closing prices. The experiment is comprised of three major phases: the data acquisition, data pre-processing and the machine learning phase. These phases are outlined in Figure: 3.6 and are explained in detail in sections 3.3.3.2 and 3.3.3.3.

3.3.3.1 Pre-processing of unstructured text data

Real-world data often require cleaning due to inconsistencies and corruption that occur during data collection. For topic modelling, news data are expected to be clean and consistent. Hence, in this step, Reddit news headlines and Reuters news article contents were pre-processed separately. The entire pre-processing task was implemented using *nltk* [29] and *gensim* [49] modules in python. These modules provide various operators for dealing with textual data. Before pre-processing, news articles were aggregated for each day. Basic

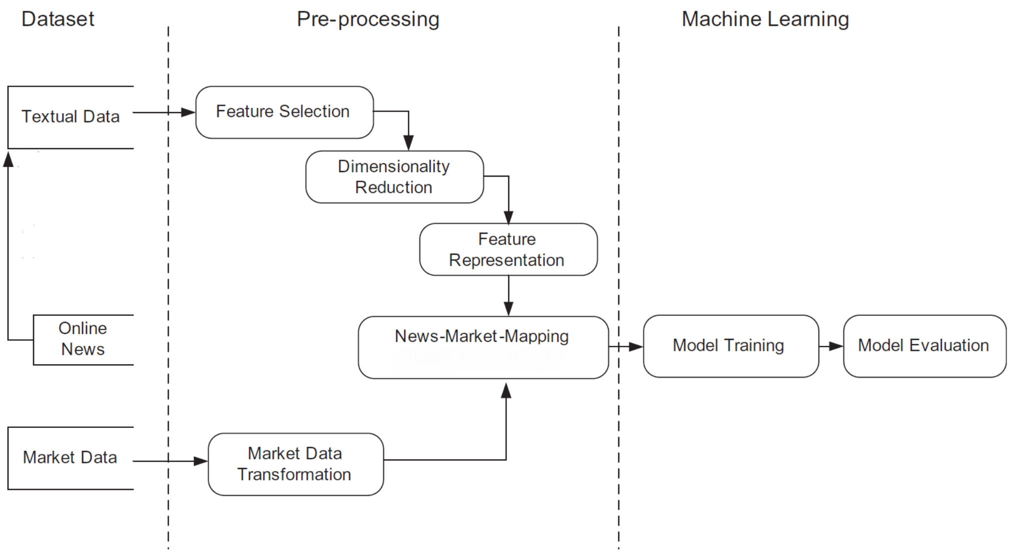


FIGURE 3.6: Research design framework for the text mining approach [36]

pre-processing tasks were performed such as text normalization, stopwords removal, tokenization, and lemmatization. Text processing aims to eliminate the words, symbols or characters that provide insignificant meaning for textual feature representation. Tokenization is a method of separating words or phrases into independent entries in the sentence. Lemmatization is a process of converting words to their root forms. In contrast with stemming, lemmatization does not merely remove endings of words. Rather it uses lexical knowledge bases to get the correct base forms of words.

This helps reduce the total number of unique words in the dictionary and thereby reducing the dimensionality of the document-word matrix that is created later. LDA requires its inputs to be presented as a document-word matrix. The document-word matrix represents the frequency of terms that occur in a collection of documents. After the pre-processing stage, textual data was transferred to LDA topic probability vectors that are later used as features when forecasting the USD/ZAR closing prices.

3.3.3.2 LDA-Based Textual Feature Representation

In this stage, we apply NLP technique for feature selection, extraction and representation. We used an unsupervised technique to perform text classification amongst documents. Specifically, LDA is used to classify news articles into clusters of topics and obtain each article's topic distribution. In the case of LDA for the document-level classification task, there are two main processes, namely, training and testing. We used the training and testing splits we defined earlier in 3.3.1.1 to ensure consistency in our experiments. We build a separate LDA model each of the news data sets. The document-word matrices of each of the news articles were used as LDA inputs. LDA models were built using the *gensim* [49] module in python. To optimize the number topics K, we run models with different values of K: (2, 18]. The optimal K is chosen for each data set on the training sets. LDA clusters sets of words based on their importance automatically from unlabelled documents in an unsupervised fashion; for this reason, we can not guarantee well intepretable output topic clusters. Therefore, topic coherence scores were used to determine the optimal number of topics K. Topic coherence measures are used to score a single topic by measuring the degree of semantic similarity between the top words in the topic [10], therefore a high score is desired. Finally, the trained models with the chosen number of topics (K) are evaluated on the test data. The obtained topic distributions are then aligned with historical currency prices. The final data sets are in time series format with $Date, Topic_1, Topic_2, ..., Topic_k, Open, Close, High,$ and Low as features.

3.3.3.3 Model Building

After aligning the topic distributions with historical currency data. The data sets are split into training and testing sets as initially described in 3.3.1.1. Prior to fitting the SVR and LSTM models, the data sets are pre-processing as described in section 3.3.2.2. We then fit the SVR and LSTM on the training data by following the experimental process described in sections 3.3.2.3 and 3.3.2.4 respectively for both data sets. Finally, the models are evaluated on testing datasets by calculating measures presented in 3.3.4.

3.3.4 Performance Measurements

We use commonly used statistical accuracy measures, namely, RMSE, R^2, and MAE [2]. These performance measures are formulated as follows [2]:

$$RMSE = \sqrt{\frac{1}{N}\sum_{t=1}^{N}\left(\hat{Y}_t - Y_t\right)^2} \tag{3.20}$$

$$MAE = \frac{1}{N}\sum_{t=1}^{N}\left|Y_t - \hat{Y}_t\right| \tag{3.21}$$

$$R^2 = 1 - \frac{\sum_{t=1}^{N}\left(Y_t - \hat{Y}_t\right)^2}{\sum_{t=1}^{N}\left(Y_t - \frac{1}{N}\sum_t Y_t\right)} \tag{3.22}$$

Where Y_t and $\hat{Y}_t$ are the original and the predicted prices at time t respectively.

3.4 Statistical Hypothesis Tests

In order to draw further inferences on the performances of our models, we perform statistical significance tests on the testing RMSEs of the models. This is done by adopting the hypothesis testing technique used by Mbuvha et al. [33], in testing the testing for differences between models using the Friedman test [15] and the Nemenyi test. The Friedman test is nonparametric test for a randomized block experimental design, it is used to compare the significant differences between multiple models with regards to their distributions and it does not assume samples are drawn from a normal distribution [15]. This is therefor an appropriate test for our performance measurement samples as they are drawn from different algorithms with unknown distributions. Specifically, the Friedman test investigates the null hypothesis (H_0) that the performances of all models come from identical but unspecified population distributions, against the alternative hypothesis (H_1) that at least one model distribution differs. The test statistic of the Friedman test is calculated as:

$$F_R = \frac{12}{rc(c+1)}\sum_{j=1}^{c} R_j^2 - 3r(c+1) \tag{3.23}$$

Where R_j is the sum of the ranks for the RMSE of model j ($j = 1, ..., c$), r is the number of blocks (number of random samples), and c is the number of groups or model types. When comparing more than 5 models, the test statistic F_R can be estimated by using a chi-squared distribution with $(c-1)$ degrees of freedom. Therefore, the null hypothesis is rejected at a selected α level of significance if the calculated value F_R is greater than the upper-tail critical value for the chi-square distribution with $(c-1)$ degrees of freedom.

When significance differences are observed between models, i.e. when the null hypothesis of the test is rejected. There are various post-hoc tests that are applicable to investigate which specific models differ from others [15]. The new set of hypotheses then becomes $H_0 : \theta_k = \theta_j$, for $k \neq j$ vs $H_1 : \theta_k \neq \theta_j$, where θ_k is the median RMSE of model k. With this test, the null hypothesis is rejected for pairwise comparisons at an α level of significance if:

$$|\bar{R}_k - \bar{R}_j| \geq r_n; k; 1 - \alpha \tag{3.24}$$

When $n \rightarrow \infty$ the inequality 3.25 becomes:

$$|\bar{R}_k - \bar{R}_j| \geq q_k; n - k; 1 - \alpha \sqrt{\frac{nk(k-1)}{12}} \tag{3.25}$$

Where $\geq q_k; n - k; 1 - \alpha$ is the $1 - \alpha$ quantile of the studentized range distribution with K and $(n - K)$ "degrees of freedom" [46].

4 Results And Discussion

In this chapter the results of the study are presented and discussed with reference to the research objectives defined in 1.2. Results are divided according to the data sets detailed in section 3.3.1.1. Specifically, experiment 1 focuses on investigating the effect of Reddit news headlines on the predictive performance when forecasting the USD/ZAR closing prices for the period between 08-August-2008 to 01-July-2016. Similary, experiment 2 focuses on investigation the effect of Reuters news contents when forecasting the USD/ZAR closing prices for the period between 30-July-2018 to 08-August-2019. Each experiment is separated into three parts, firstly, we develop the models using historical prices only, secondly, we apply LDA on news data to obtain topic distributions that are later used in combination with historical currency prices to build forecasting models. Finally, we perform statistical significance tests in order to draw further inferences from our forecasting models.

4.1 Historical Price Based Approach

4.1.1 ARIMA Parameters

We explored different parameter combinations for ARIMA models manually and the Box-Jenkins methodology suggests that the optimal models are $ARIMA(0,1,0)$ and $ARIMA(1,1,1)$ for experiment 1 and 2 respectively. These were chosen with regards to their training $RMSE$ values. Figures 4.4b shows residual diagnostic plots for the chosen an $ARIMA(0,1,0)$ for experiment 1. The two figures provide evidence that residuals of both models are likely to follow a normal distribution. Using kernel density estimation plots, q-q plots, and correlograms, we can see that the residuals reflect randomness with no time dependency. Therefore the suggested models are appropriate for making forecasts.

Similarly, figure 4.2 shows residual diagnostic plots for the chosen an $ARIMA(1,1,1)$ obtained for experiment 2.

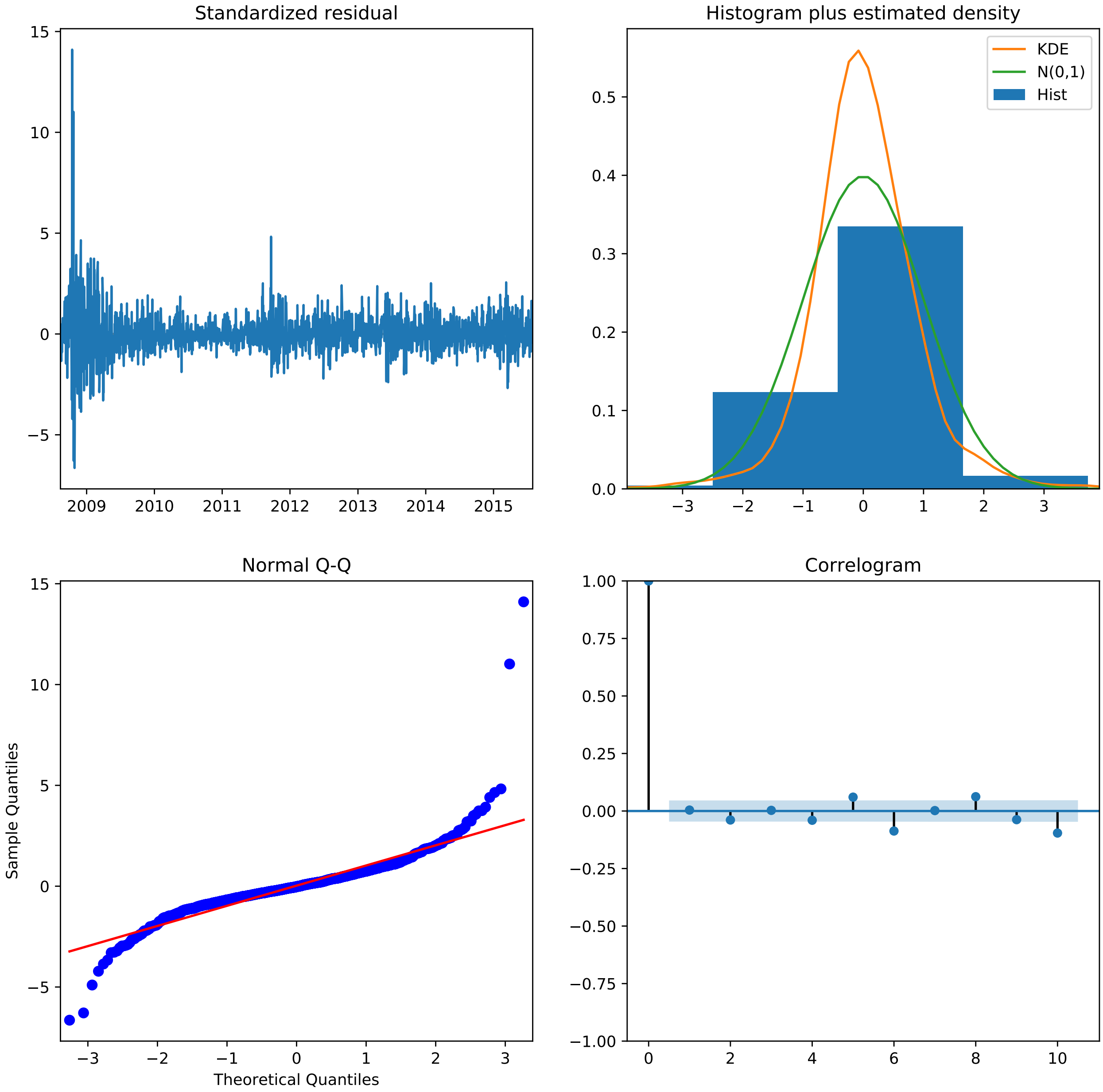


FIGURE 4.1: $ARIMA(0,1,0)$ diagnostic plots for experiment 1

4.1.2 SVR Parameters

Table 4.1 shows the optimal parameters for experiment 1 and experiment 2 models respectively. These parameters were determined using gridsearch and they demonstrate that the non linear RBF kernel is a better fit for both datasets.

Dataset	Kernel	C	ϵ	γ
Experiment 1	RBF	1000	0.1	0.0001
Experiment 2	RBF	1000	0.5	0.0001

TABLE 4.1: Ideal parameters for SVR models

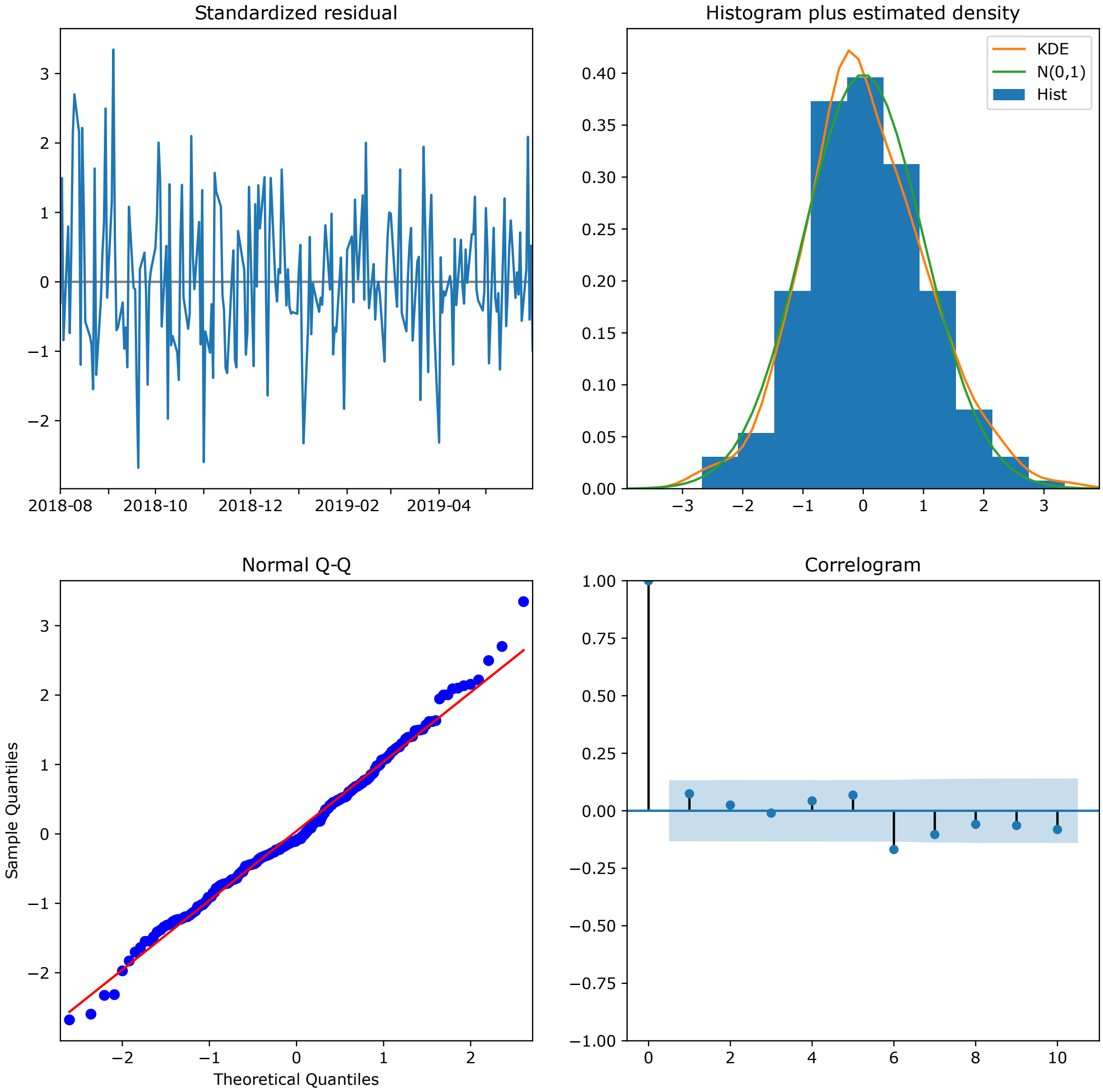


FIGURE 4.2: $ARIMA(1,1,1)$ diagnostic plots for experiment 2

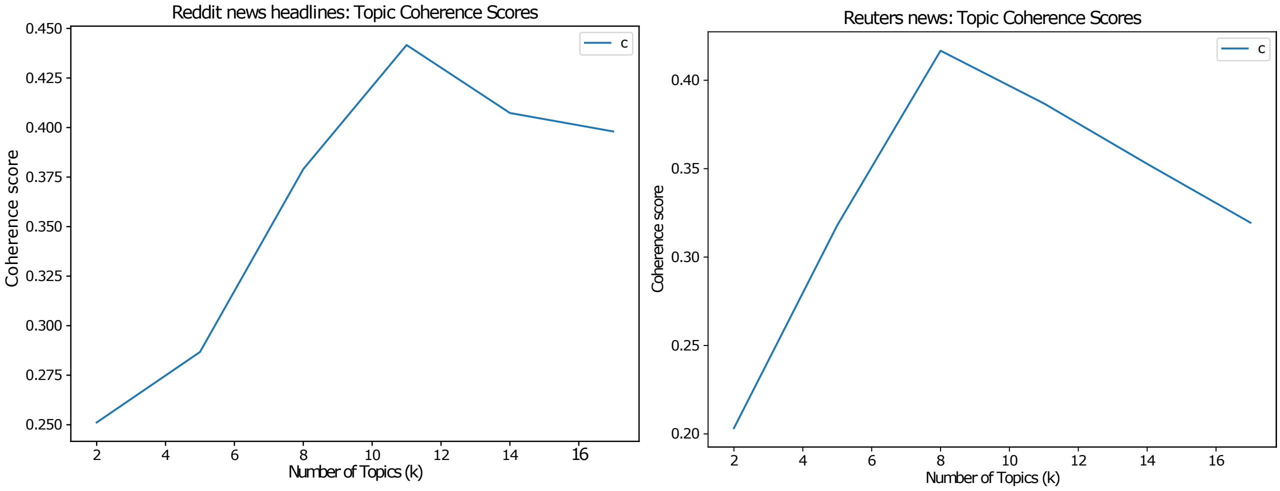
4.1.3 LSTM Parameters

Parameters of LSTM models were determined using the gridsearch function and trail and error while monitoring the R^2 measure to ensure that models were not overfitting. After running initial experiments, the Root mean square propagation (RMSProp) optimizer showed better results compared to the classic Stochastic gradient descent (SGD), and Adaptive moment estimation (Adam) [9]. Therefore, RMSProp with the learning rate of 0.001 is used in the LSTM training process for both experiments 1 and 2. The learning rate is a decreasing function that controls the number of weights updated during model training until convergence. Model convergence is vastly reliant on the number of epochs and the learning rate. For the price only approach on the experiment 1 data set, the batch size and number of epochs used are 72 and 80 respectively. Similarly, for experiment 2, the batch size and the number of epochs are both equal to 100 respectively. For the text mining based approach, we used the same optimizer (RMSProp) and learning rates while varying the LSTM architectures, dropout rates, batch sizes and the number of epochs.

4.2 LDA-Results

The model described in section 3.3.3.2 was applied to experiment 1 and 2 data sets. Figures 4.3a and 4.3b present topic coherence score comparisons for different values of k for the LDA topic models for Reddit news headlines (experiment 1 data set) and Reuters news contents (experiment 2 data set) respectively. From figure 4.3a we can see that 11 topics yield the highest topic coherence score of 0.44, therefore $k = 11$ for experiment 1. Similarly, figure 4.3b shows that an optimal number of topics for Reuters news contents is $k = 8$, with the coherence score of 0.41. Figures A.3 and A.4 depict the top 10 keywords for each topic and their weights for Reddit and Reuters news respectively. Broader topics can be inferred from keywords of topics outputs, e.g. topic 2 in figure A.4 can be identified as financial or business news.

The top right plot in figure A.3 that the top terms that contribute topic 1 are: "database", "memo", "suspicious", "obey", etc. The weight of each term is represented by the size of the bar, it reflect the importance of each keyword towards the topic. From both figures we notice the ranges of weights differ between the Reddit and Reuters news, this may be due to the relative vocabulary sizes of news headlines compared to contents. Additionally, figure A.3 shows that Reddit news are dominated by international war, politics and crime news based on the topics 2, 6, 7, 8, and 9. Figure A.4 suggests that Reuters news contents are dominated by different types of financial or business news since topics 1, 2 and 4 are clearly overlapping and can be inferred as financial news. The topics obtained from Reuters news contents highlight the known drawback of LDA described by Blei, Ng, and Jordan [6], which is the failure to capture correlations among different topics.



(A) Experiment 1: Reddit news

(B) Experiment 2: Reuters news

FIGURE 4.3: Topic coherence scores for Reddit news headlines 4.3a, and Reuters news contents 4.3b

4.3 Model Performance comparisons

Finally, we trained each model 30 times with randomly initialised weights while measuring the RMSE on the testing sets. Table 4.4 presents the average testing error (i.e R^2, $RMSE$ and MAE) rates from all models for experiment 1 and 2. The price only based models are denoted by $ARIMA_p$, SVR_p, and $LSTM_p$, while news based models are denoted by SVR_{news} and $LSTM_{news}$. LSTM models are highly sensitive to the initialization of weights compared to the ARIMA and SVR, therefore their performance values vary through different experimental runs. Table 4.2 shows that the traditional ARIMA model has better performance compared to the to non-linear models (SVR and LSTM), this is characterized by the lowest error $RMSE$ and MAE. The result suggests that lower error rates can be achieved with the news based model (i.e. SVR_{news} and $LSTM_{news}$) compared to the corresponding price only based models (i.e. SVR_{news} and $LSTM_{news}$), however, the significance of these improvements can not be determined by merely analyzing this result.

Similarly, table 4.3 presents model performance comparisons for experiment 2. This results suggests that the news based SVR_{news} outperforms the rest of the models with regards to the RMSE and MAE, however, the performance of the SVR_{news} is only marginally better than performances of the price based $ARIMA_p$ and SVR_p. Therefore, two classical statistical significance tests are conducted in order to examine the significance of performance differences among models.

TABLE 4.2: Avg. error rates for experiment 1

Model	$RMSE$	MAE
$ARIMA_p$	0.188	0.140
SVR_p	0.198	0.142
$LSTM_p$	0.361	0.316
SVR_{news}	0.197	0.142
$LSTM_{news}$	0.221	0.160

TABLE 4.3: Avg. error rates for experiment 2

Model	$RMSE$	MAE
$ARIMA_p$	0.125	0.098
SVR_p	0.127	0.099
$LSTM_p$	0.136	0.110
SVR_{news}	0.123	0.092
$LSTM_{news}$	0.138	0.153

TABLE 4.4: Model performance summary

4.4 Statistical Hypothesis Tests On Performance

Presented in table 4.5 are the summarized results for the Friedman test for performance differences between models for both experiment 1 and 2 calculated from $RMSE$ measurements. Each model type is considered as a treatment effect and the number of random experimental runs are assumed to be the blocking factor. At an $\alpha = 0.05$ level of significance, the null hypothesis is rejected for both experiment 1 and 2 with p-values of 7.813e-25 and 5.341e-25 respectively. Therefore, in both cases we do not have enough evidence to conclude that performances of all models come from a symmetric population distribution.

Data set	Test Statistic F_R	p-value
Experiment1	119.22	7.813e-25
Experiment2	120.0	5.341e-25

TABLE 4.5: Friedman test results for both Experiment 1 and 2

Since the Friedman test suggests that there are significant differences between model performances, we conduct the Nemenyi test for pairwise comparisons. Table 4.6 shows the results of the Nemenyi test at an $\alpha = 0.05$ level of significance for the experiment 1. The test reveals no statistically significant differences between performances of the ARIMA and the news based SVR (SVR_{news}), SVR_p-SVR_{news}, SVR_p-$LSTM_{news}$, and finally the $LSTM_p$ and $LSTM_{news}$. The test however suggests that there are significant differences between performances of $ARIMA_p$-SVR_p, $ARIMA_p$- $LSTM_p$, $ARIMA_p$-$LSTM_{news}$, and SVR_p-$LSTM_p$ model pairs. These results suggest that the $ARIMA_p$ significantly outperforms the rest of the models for this dataset. To assess the effectiveness incorporating news features, we compare performances of price based methods (SVR_p and $LSTM_p$) with news based approaches the SVR_{news} and $LSTM_{news}$. The use of Reddit news topic distributions for SVR and LSTM significantly improves the performance of these models compared to using historical prices only. Moreover, performances of the SVR_{news} and the $ARIMA_p$ are similar for experiment 1.

Model Pair	p-value
$ARIMA_p$ - SVR_p	0.001
$ARIMA_p$ - $LSTM_p$	0.001
$ARIMA_p$ - SVR_{news}	**0.102**
$ARIMA_p$ - $LSTM_{news}$	0.001
SVR_p - $LSTM_p$	0.001
SVR_p - SVR_{news}	**0.102**
SVR_p - $LSTM_{news}$	**0.068**
$LSTM_p$ - $LSTM_{news}$	**0.210**

TABLE 4.6: Nemenyi test results for Experiment

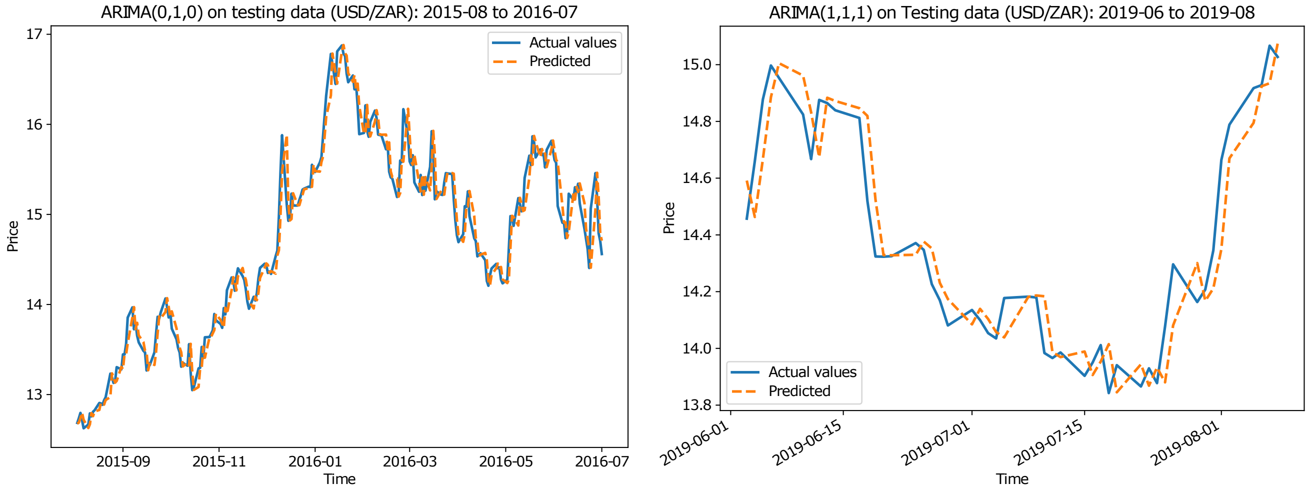
Similarly, table 4.7 shows the results of the Nemenyi test at an $\alpha = 0.05$ level of significance for the experiment 2 dataset. The result suggests that there are no significant differences between performances of the $ARIMA_p$-$LSTM_p$, $ARIMA_p$-SVR_{news}, SVR_p-$LSTM_{news}$, and $LSTM_p$-$LSTM_{news}$ model pairs. However, the result suggests that there are significant differences between the performances of $ARIMA_p$-SVR_p, $ARIMA_p$-$LSTM_{news}$, SVR_p-$LSTM_p$, and SVR_p-SVR_{news} model pairs. This is in agreement with table 4.3 which suggests that performances of the ARIMA, SVR based models and the $LSTM_p$ are comparative. This result provides sufficient evidence that the use of Reuters news topic distributions does not necessarily yield better predictive performances.

Model Pair	p-value
$ARIMA_p$ - SVR_p	0.001
$ARIMA_p$ - $LSTM_p$	**0.102**
$ARIMA_p$ - SVR_{news}	**0.102**
$ARIMA_p$ - $LSTM_{news}$	0.001
SVR_p - $LSTM_p$	0.001
SVR_p - SVR_{news}	0.001
SVR_p - $LSTM_{news}$	**0.102**
$LSTM_p$ - $LSTM_{news}$	**0.102**

TABLE 4.7: Nemenyi test results for Experiment 2

4.5 Discussion

The findings for experiment 1 and 2 demonstrate that news features and the currency market have a rudimentary relationship. News based SVR and LSTM models performed better than historical price based SVR and LSTM approaches respectively, these results are inline with Mudinas, Zhang, and Levene [35]'s result. However, it should be noted that Mudinas, Zhang, and Levene [35]'s results are based on news sentiment analysis of different datasets. The news based SVR and LSTM models did not result in improved predictions compared to the benchmark ARIMA on average. Additionally, these results corroborate findings by Aye et al. [4] that show that there are no significant improvements in the performance of non linear models in comparison to linear models when forecasting the USD/ZAR currency pair in a short term. For experiment 2, the usage of news features significantly improves the performance of the SVR, as table 4.7 shows that differences in performances of the SVR_p and are SVR_{news} significant. Furthermore, the improved SVR_{news} model outperforms the rest of the models, which demonstrates superiority of using news contents in the models.



(A) $ARIMA(0,1,0)$ (2008-2016)

(B) $ARIMA(1,1,1)$ (2018-2019)

FIGURE 4.4: Actual and predicted values from SVR models for using for experiment 1 (4.4a), and experiment 2 (4.4b)

From this, we can infer that Reuters news contents have a significant impact on the predictive power of the SVR when forecasting the USD/ZAR currency pair compared to Reddit news headlines. Additionally, both Reddit and Reuters news are not specific to South Africa and the United States, therefore, the extracted topics are merely reflecting global economic trends. Furthermore, although the performance of the news based SVR model is improved, the differences between performances of the ARIMA model and the improved SVR model are statistically insignificant according to table 4.7. Finally, results also demonstrate that there are no significance differences between performances of the LSTM based models and the traditional ARIMA, this confirms [17]'s findings. Figure 4.4a shows the graph of predicted closing price against actual USD/ZAR closing price for the testing set of experiment 1. The plot demonstrates the correlation of accuracy of the $ARIMA(0,1,0)$ model on on the testing set. Similarly, figure 4.4b shows result of the $ARIMA(1,1,1)$ based on experiment 2 data set. Plots for other models are provided in section A.3 for further model comparisons.

5 Conclusions

This research attempts to address the short-term currency forecasting problem using news articles and historical currency prices. We investigate the effect of Reddit news headlines and Reuters news article contents on the prediction of the daily USD/ZAR currency pair, by using an unsupervised topic modelling approach, i.e. LDA, is used to extract topics from raw text documents and using these as additional features to supervised learning models. We compare predictive performances the ARIMA, SVR and LSTM models. These models are equally applicable to financial forecasting problems and are well suited for forecasting exchange rates. The result shows that additional news features can yield statistically significant improvements in performances of SVR models when forecasting the daily USD/ZAR closing prices. While news features can only marginally improve LSTM models. We have also shown evidence of performance differences models using classical non-parametric Friedman tests, followed by Nemenyi post-hoc tests.

The result of the traditional univariate ARIMA model presents interesting findings for the data sets presented in this research as it outperforms most multivariate non-linear models, regardless of the use of additional news input features on average. The result obtained using Reuters news contents shows that using a larger vocabulary provides significant improvements to the SVR compared to the using shorter texts as provide by Reddit news headlines. Our findings also highlight that different types of news topics such as European monetary policy topics, international war and crime, social medial application topics and have positive predictive power in the SVR and LSTM models. The ARIMA results are in favour of the EMH [14], that states that all information is reflected in market prices. However, the insignificance of differences in performance between the ARIMA, SVR and LSTM models demonstrate the potential of news topics as appropriate features for currency market predictions.

5.1 Limitations Of The Study

The sample size of experiment 2 data is small (291 observations) and affects the power of the models used; therefore the conclusions made may be precise.

5.2 Suggestions for Further Research

For future work, we would like to adopt this research methodology on other currencies. Secondly, it would be interesting to formulate the research problem classification problem by forecasting the direction of closing prices instead of estimating the value.

A Appendix

A.1 Data Exploration

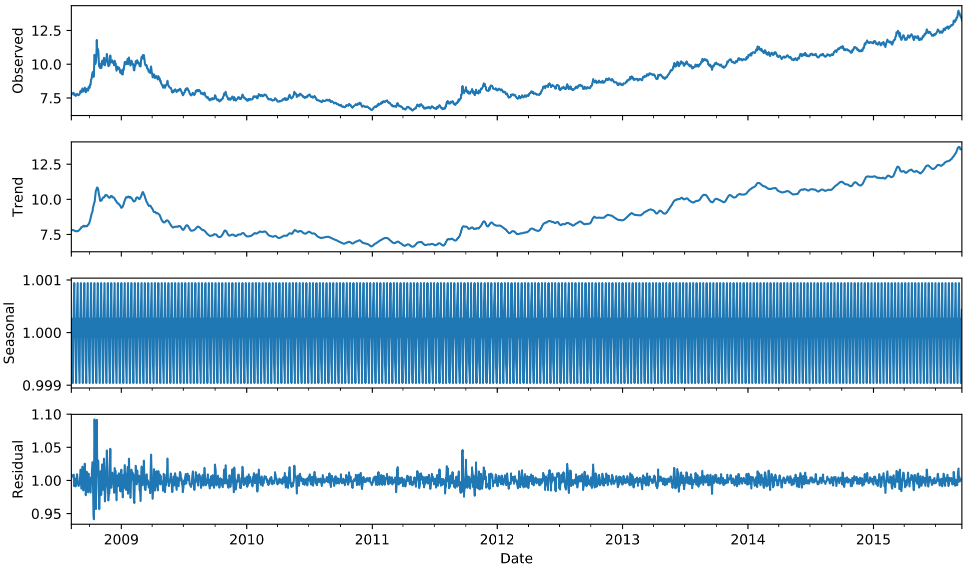


FIGURE A.1: Additive seasonal decomposition for experiment 1 (2008-2016)

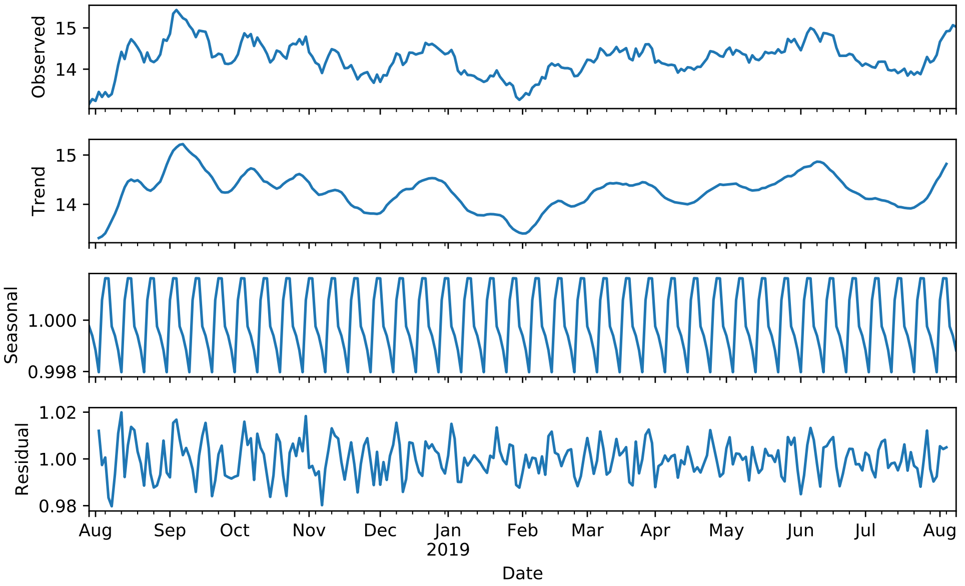


FIGURE A.2: Additive seasonal decomposition for experiment 2 (2018-2019)

A.2 LDA Topic Distributions

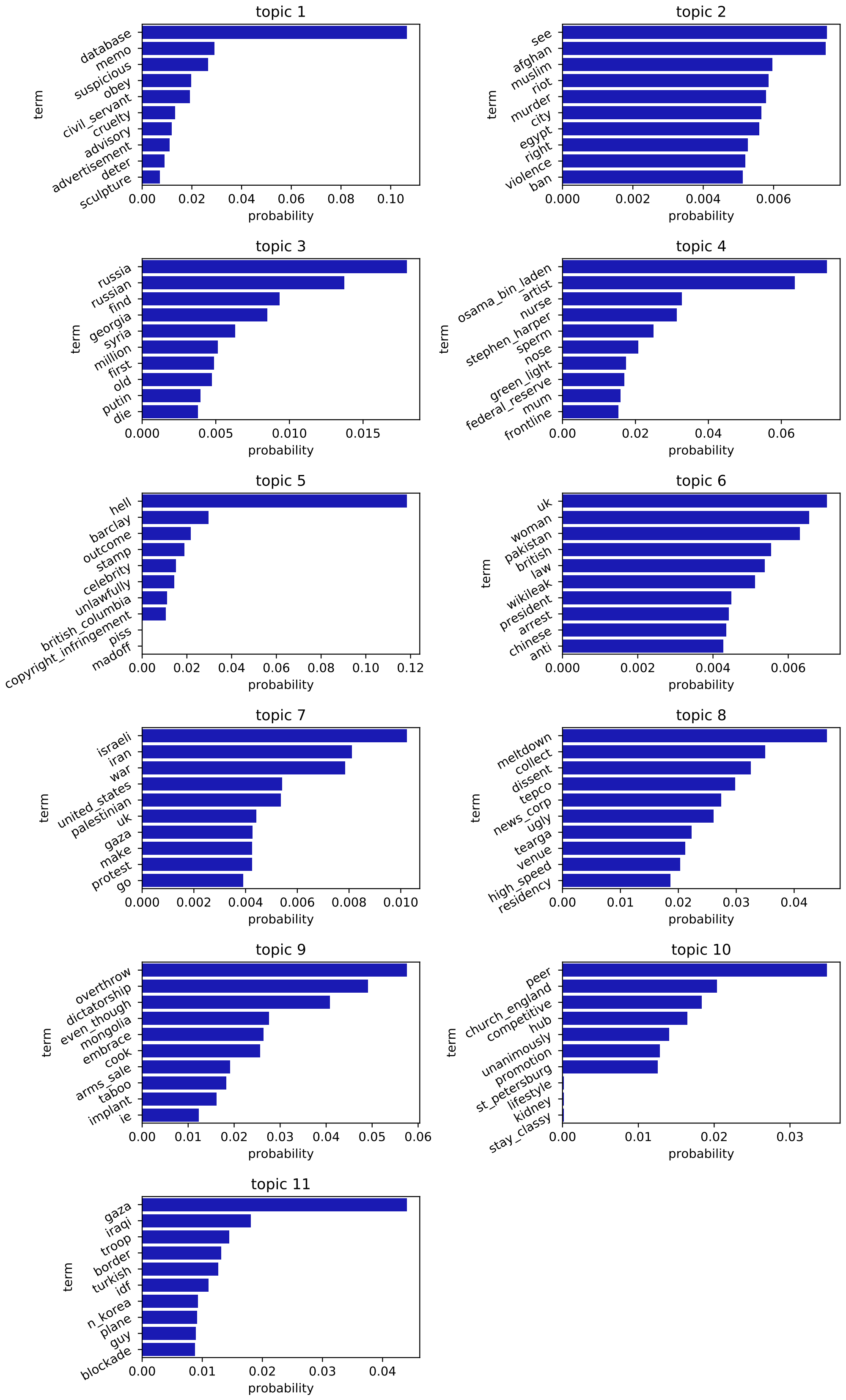


FIGURE A.3: Reddit news top 10 keyword distributions per topic

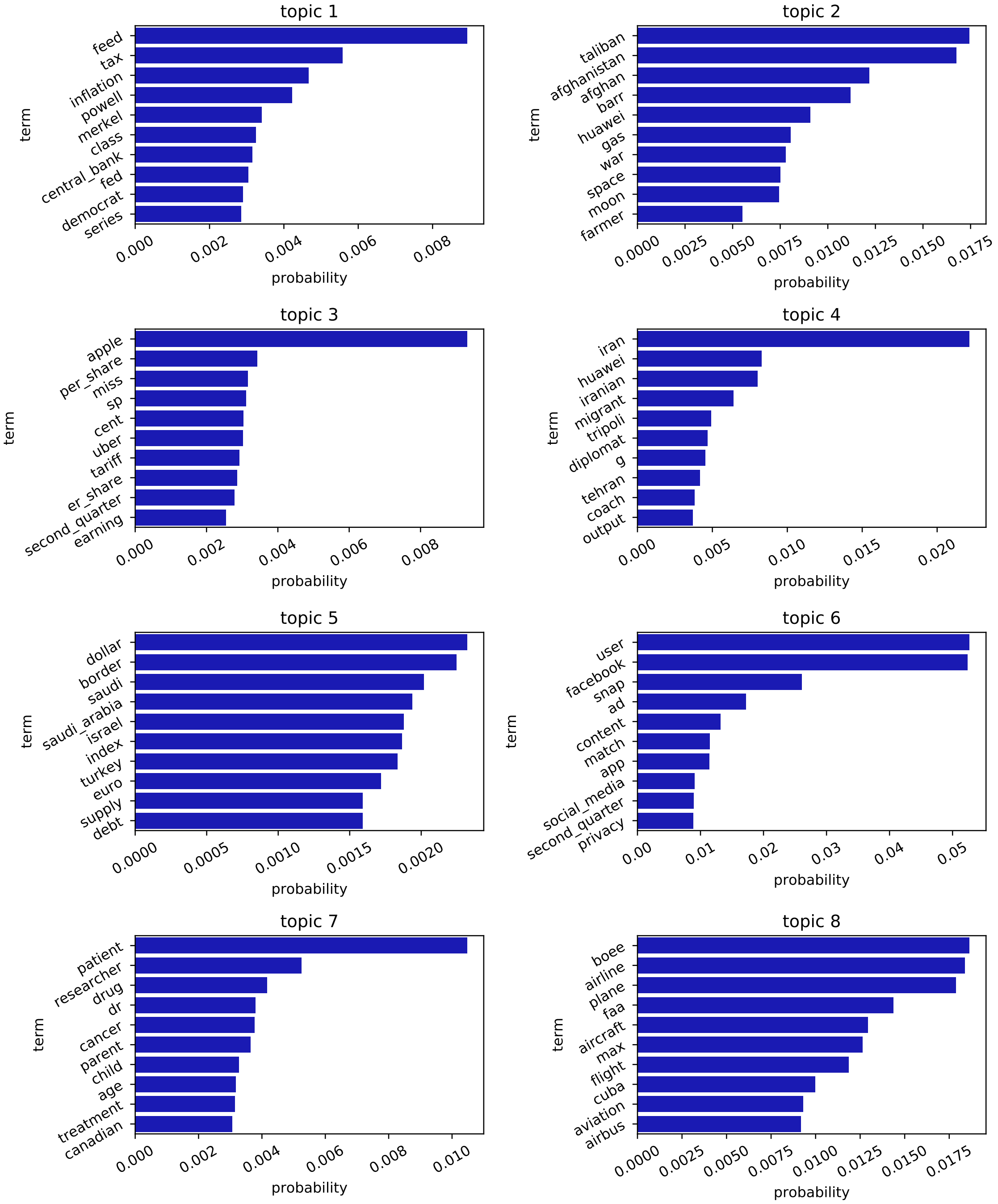
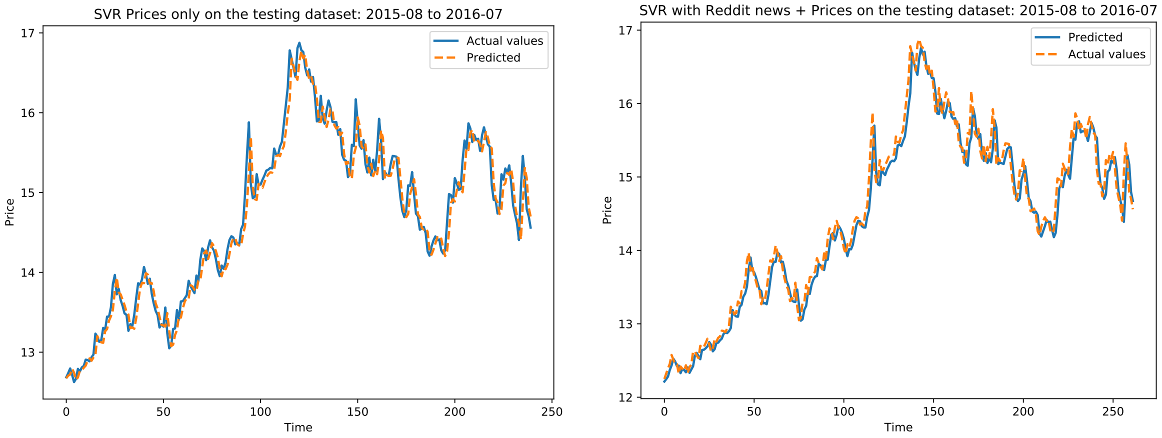


FIGURE A.4: Reuters news top 10 keyword distributions per topic

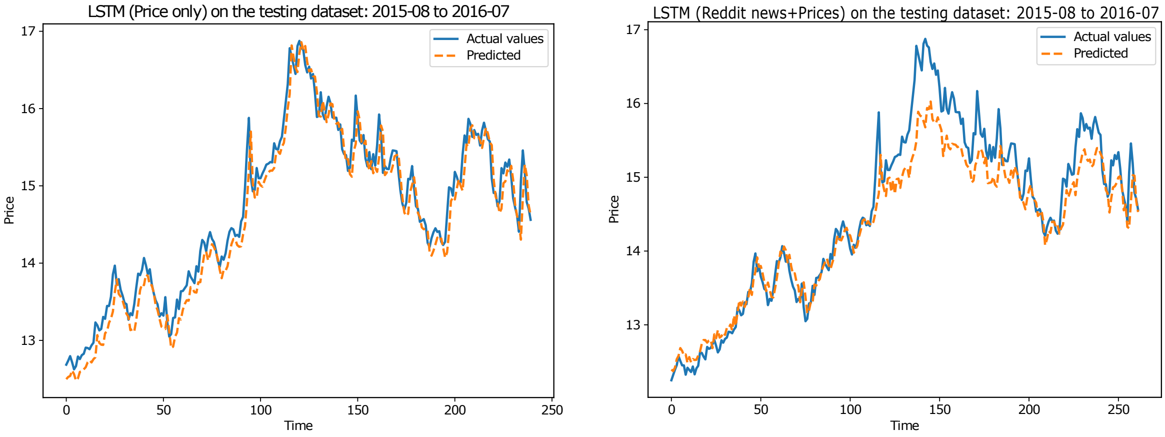
A.3 Plots



(A) SVR_p (2008-2016)

(B) SVR_{news} (2008-2016)

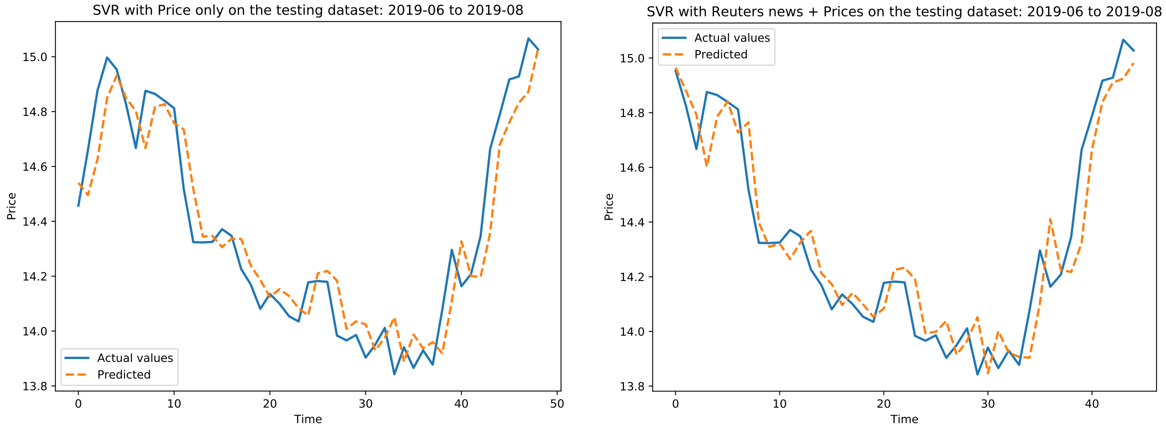
FIGURE A.5: Actual and predicted values from SVR models for experiment 1 using prices only (A.7a), and with Reddit news topic distributions (A.7b)



(A) $LSTM_p$ (2008-2016)

(B) $LSTM_{news}$ (2008-2016)

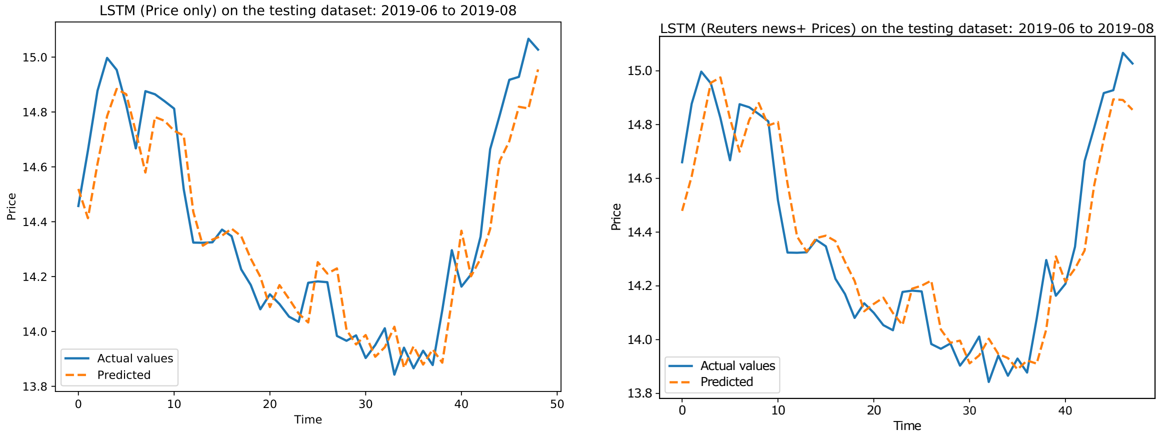
FIGURE A.6: Actual and predicted values from LSTM models for experiment 1 using prices only (A.6a), and with Reddit news topic distributions (A.6b)



(A) SVR_p (2018-2019)

(B) SVR_{news} (2018-2019)

FIGURE A.7: Actual and predicted values from SVR models for experiment 2 using prices only (A.7a), and with Reuters news topic distributions (A.7b)



(A) $LSTM_p$ (2018-2019)

(B) $LSTM_{news}$ (2018-2019)

FIGURE A.8: Actual and predicted values from LSTM models for experiment 2 using prices only (A.8a), and with Reuters news topic distributions (A.8b)

Bibliography

[1] Udit Aggarwal et al. “Indian Stock Market Analysis Using CHAID Regression Tree”. In: *Data Engineering and Intelligent Computing*. Springer, 2018, pp. 533–552.

[2] Razana Alwee et al. “Hybrid support vector regression and autoregressive integrated moving average models improved by particle swarm optimization for property crime rates forecasting with economic indicators”. In: *The Scientific World Journal* 2013 (2013).

[3] Adebiyi A Ariyo, Adewumi O Adewumi, and Charles K Ayo. “Stock price prediction using the ARIMA model”. In: *Computer Modelling and Simulation (UKSim), 2014 UKSim-AMSS 16th International Conference on*. IEEE. 2014, pp. 106–112.

[4] Goodness Chioma Aye et al. “The out-of-sample forecasting performance of non-linear models of real exchange rate behaviour: The case of the South African Rand”. In: (2013).

[5] Yoshua Bengio et al. “A neural probabilistic language model”. In: *Journal of machine learning research* 3.Feb (2003), pp. 1137–1155.

[6] David M Blei, Andrew Y Ng, and Michael I Jordan. “Latent dirichlet allocation”. In: *Journal of machine Learning research* 3.Jan (2003), pp. 993–1022.

[7] Colin Campbell. “An introduction to kernel methods”. In: *Studies in Fuzziness and Soft Computing* 66 (2001), pp. 155–192.

[8] Lijuan Cao and Francis EH Tay. “Financial forecasting using support vector machines”. In: *Neural Computing & Applications* 10.2 (2001), pp. 184–192.

[9] Wen-Chung Chang and Van-Toan Pham. “An efficient neural network with performance-based switching of candidate optimizers for point cloud matching”. In: *Proceedings of the 6th International Conference on Control, Mechatronics and Automation*. ACM. 2018, pp. 159–164.

[10] Zhiyuan Chen and Bing Liu. "Topic modeling using topics from many domains, lifelong learning and big data". In: *International Conference on Machine Learning*. 2014, pp. 703–711.

[11] François Chollet. *Keras*. `https://github.com/fchollet/keras`. 2015.

[12] C. K. Chu and James Stephen Marron. "Comparison of two bandwidth selectors with dependent errors". In: *The Annals of Statistics* 19.4 (1991), pp. 1906–1918.

[13] Luca Di Persio and Oleksandr Honchar. "Recurrent neural networks approach to the financial forecast of Google assets". In: *International journal of Mathematics and Computers in simulation* 11 (2017).

[14] Eugene F Fama. "Random walks in stock market prices". In: *Financial analysts journal* 21.1 (1965), 55–59.

[15] Milton Friedman. "A comparison of alternative tests of significance for the problem of m rankings". In: *The Annals of Mathematical Statistics* 11.1 (1940), pp. 86–92.

[16] Aditya Gupta and Bhuwan Dhingra. "Stock market prediction using hidden markov models". In: *Engineering and Systems (SCES), 2012 Students Conference on*. IEEE. 2012, pp. 1–4.

[17] Magnus Hansson. "On stock return prediction with LSTM networks". In: (2017).

[18] Sepp Hochreiter and Jürgen Schmidhuber. "Long short-term memory". In: *Neural computation* 9.8 (1997), pp. 1735–1780.

[19] Yu-Yun Hsu, Sze-Man Tse, and Berlin Wu. "A NEW APPROACH OF BIVARIATE FUZZY TIME SERIES ANALYSIS TO THE FORECASTING OF A STOCK INDEX". In: *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems* 11.06 (2003), pp. 671–690. DOI: `10.1142/S0218488503002478`.

[20] Sadegh Bafandeh Imandoust and Mohammad Bolandraftar. "Forecasting the direction of stock market index movement using three data mining techniques: the case of Tehran Stock Exchange". In: *International Journal of Engineering Research and Applications* 4.6 (2014), pp. 106–117.

[21] Zahid Iqbal et al. "Efficient Machine Learning Techniques for Stock Price Prediction". In: *Int. Journal of Engineering Research and Applications* 3.6 (2013), pp. 855–867.

[22] Hamed Jelodar et al. "Latent Dirichlet Allocation (LDA) and Topic modeling: models, applications, a survey". In: *Multimedia Tools and Applications* 78.11 (2019), pp. 15169–15211.

[23] Fang Jin et al. "Forex-foreteller: Currency trend modeling using news articles". In: *Proceedings of the 19th ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM. 2013, pp. 1470–1473.

[24] Yakup Kara, Melek Acar Boyacioglu, and Ömer Kaan Baykan. "Predicting direction of stock price index movement using artificial neural networks and support vector machines: The sample of the Istanbul Stock Exchange". In: *Expert systems with Applications* 38.5 (2011), pp. 5311–5319.

[25] Andrej Karpathy, Justin Johnson, and Li Fei-Fei. "Visualizing and understanding recurrent networks". In: *arXiv preprint arXiv:1506.02078* (2015).

[26] Manish Kumar and M Thenmozhi. "Stock Index Return Forecasting and Trading Strategy Using Hybrid ARIMA-Neural Network Model". In: *International Journal of Financial Management* 2.1 (2012), pp. 284–308.

[27] Yunli Lee, Leslie Ching Ow Tiong, and David Chek Ling Ngo. "Hidden markov models for forex trends prediction". In: *2014 International Conference on Information Science & Applications (ICISA)*. IEEE. 2014, pp. 1–4.

[28] Chongda Liu et al. "Forecasting S&P 500 Stock Index Using Statistical Learning Models". In: *Open Journal of Statistics* 6.06 (2016), p. 1067.

[29] Edward Loper and Steven Bird. "NLTK: The Natural Language Toolkit". In: *In Proceedings of the ACL Workshop on Effective Tools and Methodologies for Teaching Natural Language Processing and Computational Linguistics. Philadelphia: Association for Computational Linguistics*. 2002.

[30] Nijolė Maknickienė, Aleksandras Vytautas Rutkauskas, and Algirdas Maknickas. "Investigation of financial market prediction by recurrent neural network". In: *Innovative Technologies for Science, Business and Education* 2.11 (2011), pp. 3–8.

[31] Nikos Malandrakis, Elias Iosif, and Alexandros Potamianos. "DeepPurple: Estimating sentence semantic similarity using n-gram regression models and web snippets". In: *Proceedings of the First Joint Conference on Lexical and Computational Semantics-Volume 1:*

Proceedings of the main conference and the shared task, and Volume 2: Proceedings of the Sixth International Workshop on Semantic Evaluation. Association for Computational Linguistics. 2012, pp. 565–570.

[32] Burton G Malkiel. *A Random Walk Down Wall Street*. 1973.

[33] Rendani Mbuvha et al. "Bayesian neural networks for one-hour ahead wind power forecasting". In: *2017 IEEE 6th International Conference on Renewable Energy Research and Applications (ICRERA)*. IEEE. 2017, pp. 591–596.

[34] Sheung Yin Kevin Mo, Anqi Liu, and Steve Y Yang. "News sentiment to market impact and its feedback effect". In: *Environment Systems and Decisions* 36.2 (2016), pp. 158–166.

[35] Andrius Mudinas, Dell Zhang, and Mark Levene. "Market trend prediction using sentiment analysis: lessons learned and paths forward". In: *arXiv preprint arXiv:1903.05440* (2019).

[36] Arman Khadjeh Nassirtoussi et al. "Text mining of news-headlines for FOREX market prediction: A Multi-layer Dimension Reduction Algorithm with semantics and sentiment". In: *Expert Systems with Applications* 42.1 (2015), pp. 306–324.

[37] News API. *News API, August 2019*. from, `https://newsapi.org/`. 2019.

[38] Nguyet Nguyen. "An Analysis and Implementation of the Hidden Markov Model to Technology Stock Prediction". In: *Risks* 5.4 (2017), p. 62.

[39] Nguyet Nguyen and Dung Nguyen. "Hidden Markov model for stock selection". In: *Risks* 3.4 (2015), pp. 455–473.

[40] Thien Hai Nguyen and Kiyoaki Shirai. "Topic modeling based sentiment analysis on social media for stock market prediction". In: *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*. Vol. 1. 2015, pp. 1354–1364.

[41] Thomas Oberlechner. "Importance of technical and fundamental analysis in the European foreign exchange market". In: *International Journal of Finance & Economics* 6.1 (2001), pp. 81–93.

[42] Fulya Ozcan. "Exchange Rate Prediction from Twitter's Trending Topics". In: (2016).

[43] Jigar Patel et al. “Predicting stock and stock price index movement using trend deterministic data preparation and machine learning techniques”. In: *Expert Systems with Applications* 42.1 (2015), pp. 259–268.

[44] F. Pedregosa et al. “Scikit-learn: Machine Learning in Python”. In: *Journal of Machine Learning Research* 12 (2011), pp. 2825–2830.

[45] Desh Peramunetilleke and Raymond K Wong. “Currency exchange rate forecasting from news headlines”. In: *Australian Computer Science Communications* 24.2 (2002), pp. 131–139.

[46] Dulce G Pereira, Anabela Afonso, and Fátima Melo Medeiros. “Overview of Friedman’s test and post-hoc analysis”. In: *Communications in Statistics-Simulation and Computation* 44.10 (2015), pp. 2636–2653.

[47] Sujin Pyo et al. “Predictability of machine learning techniques to forecast the trends of market index prices: Hypothesis testing for the Korean stock markets”. In: *PloS one* 12.11 (2017), e0188107.

[48] Jeff Racine. “Consistent cross-validatory model-selection for dependent data: hv-block cross-validation”. In: *Journal of econometrics* 99.1 (2000), pp. 39–61.

[49] Radim Řehůřek and Petr Sojka. “Software Framework for Topic Modelling with Large Corpora”. English. In: *Proceedings of the LREC 2010 Workshop on New Challenges for NLP Frameworks*. Valletta, Malta: ELRA, May 2010, pp. 45–50.

[50] Robert P Schumaker and Hsinchun Chen. “Textual analysis of stock market prediction using breaking financial news: The AZFin text system”. In: *ACM Transactions on Information Systems (TOIS)* 27.2 (2009), p. 12.

[51] Robert P Schumaker et al. “Evaluating sentiment in financial news articles”. In: *Decision Support Systems* 53.3 (2012), pp. 458–464.

[52] Xu Selene Yue. “Stock price forecasting using information from Yahoo finance and Google trend”. In: *UC Brekley* (2014).

[53] Alaa F Sheta, Sara Elsir M Ahmed, and Hossam Faris. “A comparison between regression, artificial neural networks and support vector machines for predicting stock market index”. In: *Soft Computing* 7.8 (2015).

[54] Sun, J. *Daily News for Stock Market Prediction, Version 1, August 2019*. from, `https://www.kaggle.com/aaron7sun/stocknews`. 2016, August.

[55] Q Xin-Yao and Gao Shan. "Financial Series Prediction: Comparison Between Precision of Time Series Models and Machine Learning Methods". In: *arXiv preprint arXiv:1706.00948* (2017).

[56] Frank Z Xing, Erik Cambria, and Roy E Welsch. "Natural language based financial forecasting: a survey". In: *Artificial Intelligence Review* 50.1 (2018), pp. 49–73.

[57] J Zhang and S Li. "Financial Time Series Analysis Model for Stock Index Forecasting". In: *International Journal of Simulation–Systems, Science & Technology* 17.16 (2016).

[58] Xiaolian Zheng and Ben M Chen. *Stock Market Modeling and Forecasting*. Vol. 442. Springer, 2013.