Market share prediction model based on customer sentiment analysis using Twitter data for a Mobile Money Operator in Zimbabwe: A case for Ecocash

By

Trevor Kauyu

(C1213036T)

A Research Dissertation Submitted in Partial Fulfilment of the

Requirements for the Degree of

Master of Science in Data Analytics

Graduate Business School

Chinhoyi University of Technology

Zimbabwe

Supervisor(s)

C. Chiundidza

Month Year

CHINHOYI UNIVERSITY OF TECHNOLOGY

APPROVAL FORM

The undersigned certify that they have read and recommended to the Graduate Business School, Chinhoyi University of Technology, for acceptance a dissertation entitled, "*Market share prediction model based on customer sentiment analysis using Twitter data for a Mobile Money Operator in Zimbabwe: A case for Ecocash*", submitted by, in partial fulfilment of the requirements for the Master of Science Degree in *Big Data Analytics*.

Name of Supervisor: Mr C. Chiundidza

Signature: 

Date: 12/01/2022

DECLARATION

I, declare that this MSc study is my own effort and is a true reflection of research executed by me. This research in full or part thereof has not been submitted for examination for any degree at any other university/institution.

Name of Student: Trevor Kauyu

Signature: 

Date: 12-01-2022

DEDICATION

I dedicate this write-up to my family, friends, workmates, and colleagues for their continued support throughout the entire process.

ACKNOWLEDGEMENTS

I am extending my sincere thanks to my supervisor, his patience, guidance, and recommendations made me produce this work in its present form which is much better than my initial expectation. I would also like to thank the Master of Science in Data Analytics lecturers at the Chinhoyi University of Technology Graduate Business School for their teachings as all that I have learnt has played a part in coming up with this dissertation. I extend my gratitude to my fellow classmate and friend Mr T. Dadirai for the general support during the whole process.

ABSTRACT

In the field of customer analytics, sentiment analysis is becoming increasingly popular. Customers are spending an alarming amount of time on social media, and in most situations, customers use social media to vent their frustrations about a product or service. To acquire a better understanding of how customers feel about your organization, you must analyze the huge amounts of unstructured data collected every day. The main objective of this study aimed at developing a machine learning model to perform twitter sentiment analysis and market share trend forecasting for EcoCash Mobile Money Operator. The model was able to classify sentiments and forecast market share trends with accuracy. For the model, four years of Twitter data regarding EcoCash were retrieved. LSTM-CNN hybrid model was used for forecasting the market share trends for EcoCash Mobile Money Operator. From the research, the predicted trends in active mobile money subscriptions for Ecocash shows a steady rise in the number of active mobile money subscribers thus resulting in increased market share for the operator. The study recommends that there is need for further research in finding the best way for sentiment detection and classification of local languages since the dataset that was used contained several tweets in the local language (Shona and Ndebele).

Table of Contents

List Of Figures

List Of Tables

Abbreviations

AI	Artificial Intelligence
LSTM	Long Short-Term Memory
GRU	Gated Recurrent Unit
ML	Machine Learning
EDA	Exploratory Data Analysis
ML	Machine learning
CNN	Convolutional Neural Network
RNN	Recurrent Neural Network
MMO	Mobile Money Operator
AMMS	Active Mobile Money Subscribers

Definition Of Key Terms

Sentiment Analysis: is the process of extracting the feeling of someone through the mining of opinions from a piece of text and categorise the writer's attitude into positive, negative and neutral towards a particular topic

Machine Learning: a branch of artificial intelligence where machines learn from data to discover hidden patterns and solve complex problems with very minimal human intervention.

Customer Churn (Attrition)*:* is a customer's tendency to abandon a brand and cease to be a paying customer of a certain firm. It is also termed as customer attrition.

Customer Churn (Attrition) Rate: The percentage of customers that switch using a company's products or services to a rival during a particular period.

Mobile Money: mobile money is a service in which a mobile phone is used to access financial services (GSMA, 2011; Tobbin, 2011; Jenkins, 2008).

CHAPTER 1: Introduction

1.1 Background of the study

Mobile payment technology is extremely important in the financial sector for most countries, especially developing countries where many low-income families and microbusinesses lack ready access to financial services. Thus, mobile money is poised to become a very powerful tool for financial inclusion in Zimbabwe (Mandizha, 2013). However, mobile money operators within the sector have experienced fluctuations on the active mobile money subscriber base and market share due to customer churn.

Customers are the most valuable assets in every industry because they are the primary source of profit. Companies have realized that they must put up significant effort not merely to persuade clients, but also to maintain existing customers. Customer churn has become a big concern for organizations as customers defect to a competitor. Sales and marketing departments are constantly expending significant time and money to acquire new customers, and those customers do not remain around long enough for the company to help offset the cost of acquisition, resulting in greater cost of sales and lower profit margins.

MTN Uganda's churn rate for regular mobile subscribers in 2010 was 4.5 percent each month, compared to 0.2 percent for active mobile money customers (Leishman, 2011). The overall business profitability is increased if a business reduces the rate of customer churn (Ho-Young, 2010). MTN Uganda's churn reduction advantages account for 33% of revenue produced, supporting the claim that lower churn rates enhance mobile money profitability (Leishman, 2011). It's crucial to note, nonetheless, that the incentives of reducing churn will not be achieved if consumers who aren't interested in the service and has a poor customer experience, agent networks that aren't well-planned, and service offerings that aren't up to par are registered by a mobile money operator (Ho-Young, 2010).

EcoCash is the market leader and dominating mobile money platform in Zimbabwe, provided by Econet Wireless Zimbabwe Limited (Econet). According to Potraz Sector Performance Report for Q2 2020, Ecocash experienced a negative variance of -7.6% that is from 7,065,382 down to 6,530,000 mobile money subscribers between the first quarter of 2020 and second quarter of 2020. Telecash also experienced a negative variance of -34.0% that is from 52,564 down to 34,689 subscribers between the same periods mentioned. However, Telone was the

only money operator that experienced a positive variance of 60.8% that is from 555,255 up to 892,963 mobile money subscribers between 1st Quarter 2020 and 2nd Quarter 2020. Econet and Telecel both lost market share, with Econet losing 4.6 percent and Telecel losing 0.2 percent, respectively, due to a drop in active mobile money subscribers. while NetOne gained market share by 4.8%. The declining trend in active mobile money subscribers and market share for Ecocash between 1st Quarter 2020 and 2nd Quarter 2020 is crucial to be investigated so as to formulate best strategies and apply adequate policy correction since mobile money subscribers and market share are crucial to the mobile money operator. (TechZim, 2021) reported that EcoCash lost its subscriber due to several reasons mentioned in Table 1.1.

Period	Effect
21st April 2020	Reduction in daily, monthly, and transactional limits
4th May 2020	Suspension of Agents with transactions above ZW$100,000 and requirement for their re-registration,
4th June 2020,	Suspension of Agent-to-Agent transactions
26th June 2020	Directive to integrate to Zimswitch, in line with SI 80, by 30 September 2020,
27th June 2020	Suspension of some Ecocash User Categories and Functions,
25th August 2020	Revision of mobile money limits and permissible transactions
25th August 2020	Ban of use of multiple wallets by individuals effective 8 September 2020.

Table 1.1: What impacted EcoCash (TechZim, 2021)

Companies are worried about keeping or retaining existing clients since they are considered a profit and keeping them is less expensive than gaining a new one. Each business strives to keep its clients by increasing their loyalty. Customers are excellent market ambassadors (Adnan, et al., 2019) because they may be used to advertise the company's product or service. Possible churn indicators recognition, customer requirements fulfilment, loyalty repairing, and re-establishment are all measures intended to assist the business in reducing the cost of acquiring new customers (Mitkees, Ibrahim, & Elseddawy, 2017).

Being able to analyse and forecast customer churning behaviour in advance offers a business a competitive advantage in terms of retaining and growing its client base which will in turn impact business revenue positively. Through the utilization of social media as a source of information, Mobile Money Operators should conduct customer sentiment analysis to aid in informative decision making, effective response to customer changing needs, retain customers and counter competitors thus impacting positively on the operator's market share.

Sentiment analysis is a process of building opinion mining systems to collect people's opinions from social media comments and reviews from platforms like Twitter and other blog posts (Martin, 2018). For sentiment analysis, numerous machine learning approaches have been created and employed, and there has been significant progress from traditional machine learning approaches to deep learning approaches, also known as Artificial Neural Networks. Traditional Machine Learning techniques can obtain reasonable results especially on small data sets but also suffer from issues like needing human work in developing features, missing values can have a significant impact on classification results, they are unable to identify complicated data patterns, and have no good solution for considering work order. Traditional machine learning techniques also perform poorly with cross-lingual or cross-domain data (Sun, Luo & Chen, 2017). Deep Learning approaches have solved these issues; hence a Deep Learning methodology is used in this study.

Over the last several years, Deep learning approaches have greatly outperformed classical methods in various NLP tasks (Chen & Manning, 2014); (Bahdanau, Cho, & Bengio, 2014), and this trend, sentiment analysis has not been spared (Rojas-Barahona, 2016). (Ali, El-Hamid, & Youssif, 2019) carried out sentiment analysis using movie reviews dataset by utilizing four deep learning algorithms, namely CNN, LSTM, MLP and CNN_LSTM. The experiments have shown that CNN_LSTM outperformed other deep learning methods thus in this research, CNN_LSTM which is a hybrid model that combines CNN and LSTM algorithms will be used.

The research's main goal is to use machine learning techniques to create a customer sentiment analysis and market share prediction model for Ecocash Mobile Payment Operator to formulate strategies to reduce or eliminate customer attrition, which the company is presently experiencing.

1.2 Statement of the problem

Ecocash is one of the biggest and most successful mobile money operators in Zimbabwe and its market share is the largest within the mobile money industry in the country. Arguably, one of the reasons why Zimbabweans have embraced Ecocash is because of the goodwill it gained from the public since its inception and the push by the government to adopt a cashless based economy. However, that sentiment has gradually eroded over the years, increasingly becoming more and more negative over the last few years which has resulted in huge losses by the operator. Several strategies have been employed to keep present clients and also to acquire new customers for revenue generation and profit maximization. In spite of the efforts towards customer retention and loyalty offering, Ecocash continues to face customer churn as customers continue to defect to rivals within the mobile money sector which is negatively impacting the customer base for the mobile operator as evidenced by the negative market share variations faced by Ecocash and the positive variations experienced by Ecocash's rival OneMoney as reported in POTRAZ quarterly reports.

This research seeks to create a machine learning model for customer sentiment analysis and market share forecasting using twitter data and market share trend data for Ecocash Mobile Money Operator. The customer sentiment analysis and market share trend prediction model will enable the mobile money operator to track customer sentiment and emotions over time, determine which customer segment feels more strongly about its brand, keep track of how user behaviour varies in response to trends in consumer base. and find out key promoters and detractors to mitigate customer attrition which the operator is currently experiencing.

1.3 Research objectives

1.3.1 Main Research Objective

To develop a machine learning model for customer sentiment analysis and market share trend prediction using twitter data for Ecocash Mobile Money Operator in Zimbabwe.

1.3.2 Specific Research objectives

1. To construct a time series analysis of how the frequencies of customer complaints have changed over the last 5 years.
2. To present the most prevalent customer complaints categories.
3. To evaluate the impact of social media sentiments to the actual business market share.
4. To forecast future market share changes in relation to customer complaint changes.

1.4 Research questions

1.4.1 Main Research Question

How can machine learning be used for predicting market share trends based on customer sentiment analysis using twitter data for Ecocash Mobile Money Operator?

1.4.2 Specific Research Questions

The study is guided by the following sub-research questions:

1. How has the frequency of complaints changed over the last 5 years?
2. What are the most prevalent customer complaints categories?
3. What is the impact of social media sentiments on business market share?
4. What are the future market share changes in relation to customer complaint changes?

1.5 Research hypothesis

The study is carried out to prove the following hypothesis:

> H1: Online customer sentiments determine future market share trends for mobile money operators.

1.6 Significance of the study

The findings of this study are expected to have an impact on several stakeholders namely, the researcher, Chinhoyi University of Technology, mobile money operators.

1.6.1 The Researcher

The research will help the research with better understanding in solving real-world problems while applying theoretical aspects and practical concepts learned. This research will also add the researcher in the list of those contributing to the body of knowledge. This is because the researcher intends to publish a paper from this research. Carrying out the research exposes the researcher to customer attrition patterns and their influence on market share trends for mobile money providers, and the effect of machine learning in the mobile money sector. This increases the researcher's knowledge depth in the context of the investigation

1.6.2 Chinhoyi University of Technology

The research is conducted to meet the university's study requirements for the degree being pursued. As a result, the study is intended to improve the university's knowledge base by filling that important literature gap. Therefore, the research will add up to the existing literature, which shall be a source of reference material for other students and staff of CUT who might in future want to carry out research on Sentiment Analysis. Thus, the research gives a concrete base for further studies in relation to this area of study.

1.6.3 Mobile Money Operator

The research brings an understanding of the trends and impact of customer complaints with regards to service offering by the mobile money operator in relation to the market share trends for the company. It will also help the Mobile Money Operator in assessing its customer retention strategies relative to the customer churning trends thus aiding informative decision making. Hence by harnessing customer feelings from media platforms like twitter, the operator may then determine if their market share is going to rise or reduce in the future and make suitable tactical actions that reduces customer churn or attrition rate.

1.7 Delimitations of the study

(Saunders, 2019) defines delimitations of study as the boundaries that have been set for the study. This research will confine itself to studying the relationship between customer sentiments and market share responsiveness as depicted by existing market share trends data. The study shall focus on analysing the effect of sentiment analysis on market share using machine learning as well as forecasting future market share trends. The study is going to use twitter data collected for the past 4 years ranging from the year 2017 to 2021 as well as quarterly market share trend data for the past 4 years for Ecocash Zimbabwe. Therefore, the product of this research is for EcoCash only.

1.8 Limitations of study

Some of the comments were made in non-English terminology, usually slang and short words, because our data source came from the worldwide domain with no restrictions on language, grammar, or spellings. As a result, during the data cleaning stage, the researcher eliminates most of these words by restricting the length of words to no more than two characters.

1.9 Dissertation structure

The study will be carried out in Five Chapters as discussed in the following sub-paragraphs:

1.9.1 Chapter 1: Introduction

Chapter One introduces the study and gives a background to the existence of the research problem. It will also give an account of the research objectives, research problems as well as the hypothesis of the study. It will also cover the significance of the study as well as pronounce the delimitations of the study.

1.9.2 Chapter 2: Literature Review

This Chapter will focus on Literature Review which explores previous studies for empirical evidence as well as define the theoretical framework of the study. A discussion of the theories around which the study evolves will be provided. This chapter will also provide the conceptual framework of the study.

1.9.3 Chapter 3: Research Methodology

Chapter Three will provide a detailed outline of the methodology used to carry out the study in terms of the philosophy adopted and the corresponding research design. An account of the study population as well as the sampling techniques used will be provided together with the research instruments used to collect data about the study. Chapter Three will also give an account of the data analysis techniques used in the study as well as the validity and reliability of the research instruments used in the study.

1.9.4 Chapter 4: Results and Discussion

This Chapter will give a detailed presentation of the results of the study as collected in Chapter Three. Machine learning techniques will be used to analyse the data and visualisations will be used to present the data so that insights can be drawn from the data and facilitate generalisation of the research findings.

1.9.5 Chapter 5: Summary, Conclusions, and Implications

The last Chapter of the study will provide a summary of the study as well as give a detailed account of the conclusions and the implications of the study. This will be done in line with the objectives of the study.

1.10 Chapter Summary

This Chapter discussed the introduction and background to the problem under investigation which traced the customer church and sentiment analysis from a global perspective down to the domestic mobile money sector. The Chapter also gave the objectives and research questions which the study seeks to achieve and provide answers for respectively. Additionally, the Chapter provided a discussion of significance of the study as well as the delimitations of the study. The Chapter rounded off with an outline of the structure of the dissertation which gave a brief description of what each Chapter of the study covers. The coming Chapter discusses the literature review of the study which gives the discussion of the context of the problem as well as the empirical evidence available on the subject under study.

CHAPTER 2: Literature review

2.1 Introduction

Chapter one focused on the introduction and background of the study which gave the outline of the research problem as well as an account of the significance of the study. This section of the study discusses the literature related to the study in which the theory guiding the conduct of the research is discussed as well as the exploration of previous studies. It also considers the theoretical framework of the study which considers the theories around which the study evolves as well as a discussion of the concept of machine learning with a particular emphasis on the sentiment analysis algorithms found in machine learning. The review will focus on published journal articles, books, and reliable online content. This Chapter gives a discussion of the empirical evidence with regards to the related works and finally, an outline of the conceptual framework which provides the researcher's understanding of the relationship among the research variables.

2.2 What is Customer Churn?

Customer churn is defined by (Laudon & Laudon, 2012) as the number of customers that discontinue buying or using a company's products or services. According to these analysts, churn rate is an important measure of a company's customer base's growth or decrease. Customer churn was recently found to be one of the biggest problems, especially for big organizations such as Econet in the telecom field (Ahmad et al., 2019). According to (Reichheld & Sasser, 1990) a rise in the defection rate leads to a decrease in cash flow for the company, even if the company can replace lost consumers by obtaining new ones. According to a survey by the consulting firm McKinsey, lowering churn may boost a typical US cellular carrier's profitability by as much as 9.9%. According to (Lemmens & Croux, 2006), churn is a marketing phrase that describes current customers who decide to migrate their business to another provider in brief bursts. Customer churn, also known as customer defection or consumer attrition, is roughly defined as the rate at which a corporation loses customers or revenue through customer defection.

2.3 Concept of Sentiment Analysis

(Bhardwaj, Narayan, Vanraj, Kumar, & Dutta, 2015) defined sentiment analysis as an approach used to extract intelligent information from people's feelings and opinions on the internet social media platforms, such as twitter and news platforms. Another research (Gohil et al., 2018) defined a sentiment as a typical statistic for determining whether a person has a favourable or unfavourable view from online social media platforms messages. As a result of the recent increase in social media use, it is now possible to collect data from social media sites such as Facebook, Twitter, blogs, and other user forums and use the data as a source for sentiment analysis (Feldman, 2013). Sentiment analysis tasks with promising results can be achieved when Deep Neural Networks models are applied (Ehsan et al. (2021); Dang et al. (2020)). Xu (2019) noticed that distributed word representation has been applied in most cases for sentiment analysis, however, it ignores sentiment of the words as it focuses only on the semantic of the word.

Some scholars argue that sentiment analysis and opinion mining are not the same thing. Opinion mining extracts and analyses people's opinions on a larger scale, whereas sentiment analysis is solely concerned with the polarity of emotions portrayed in the text (Hassan & Medhat, 2014).

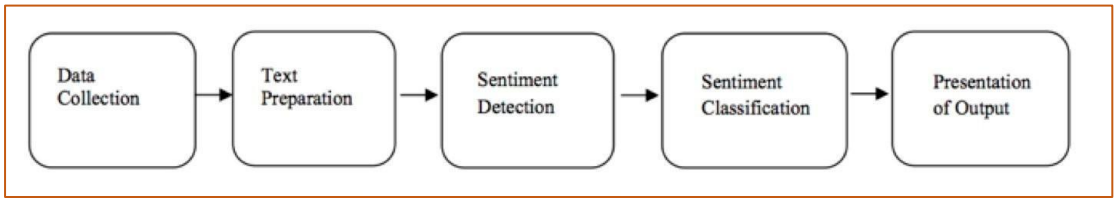


Figure 1: Sentiment Analysis Process (Martin, 2018)

Sentiment analysis is useful in practically every situation in which your potential and current consumers identify themselves. In most cases, these remarks were not meant for direct consumption by any organization but were just a mechanism for consumers to share their delight or dissatisfaction amongst themselves.

Therefore, implementing automated sentiment analysis is a critical step when you are looking to extract vital customer feedback from the various social media channels. In Mobile Money

industry, a customer services officer or a digital marketer cannot monitor and understand all that is communicated through social media platforms like twitter as there is large volume of data. Because information is disseminated too quickly and over too many channels, the process should be automated so that it is possible to analyze not only each individual remark but also the collective thoughts stated in real time.

2.4 Sentiment Analysis in Zimbabwe

In Zimbabwe's mobile money payment industry, there is relatively little literature on consumer sentiment analysis. When the researcher attempted to collect data on the quantity and types of sentiment analysis performed in Zimbabwe, the researcher discovered that such data is hardly available. However, International Journal of Science and Research online, one sentiment analysis research in Zimbabwe which mainly focused on predicting social unrest in Zimbabwe using data from twitter (Mbunge, Vheremu, & Kajiva, 2017). People use social media platforms like WhatsApp, Facebook, and Twitter to communicate their social, economic, and political ideas, thoughts, attitudes, and emotions, according to the study. These social media platforms are frequently used by social groups to plot and disseminate information that leads to riots and huge protests against the government. As a result, the researchers developed a tool to predict the possibility of having social unrest in Zimbabwe using Twitter data (Kajiva, 2017). In Zimbabwe's mobile money payment sector, sentiment analysis is a relatively new concept.

2.5 Mobile Money Subscriber and Market Share Trends in Zimbabwe

2.5.1 Active Mobile Money Subscribers

OPERATOR	1st Quarter 2020	2nd Quarter 2020	Variance (%)
ECOCASH	7,065,382	6,530,000	-7.6%
TELECASH	52,564	34,689	-34.0%
ONE MONEY	555,255	892,963	60.8%
TOTAL	**7,673,201**	**7,457,652**	-2.8%

Table 2.1: Market distribution for Mobile Money operators (POTRAZ, 2020)

The overall number of active mobile money subscribers fell by 2.8 percent to 7,457,662 in the first quarter of 2020, down from 7,673,201 in the previous quarter as shown in table 2.1 above. Ecocash experienced a negative variance of -7.6% that is from 7,065,382 down to 6,530,000 mobile money subscribers between the first quarter of 2020 and second quarter of 2020.

Telecash also experienced a negative variance of -34.0% that is from 52,564 down to 34,689 subscribers between the same periods mentioned. However, Telone was the only money operator that experienced a positive variance of 60.8% that is from 555,255 up to 892,963 mobile money subscribers between 1st Quarter 2020 and 2nd Quarter 2020. In accordance with the drop in active mobile money subscriptions, Econet and Telecel lost 4.6 percent and 0.2 percent market share, respectively, while NetOne gained 4.8 percent. This clearly shows that the customer churn faced by Econet was a benefit to NetOne, which means EcoCash customers became One-money customers. The declining trend in active mobile money subscribers for Ecocash between 1st Quarter 2020 and 2nd Quarter 2020 is crucial to be investigated to formulate best strategies and apply adequate policy correction since mobile money subscribers are crucial to the mobile money operator.

2.5.2 Active Mobile Money Subscriptions Market Share

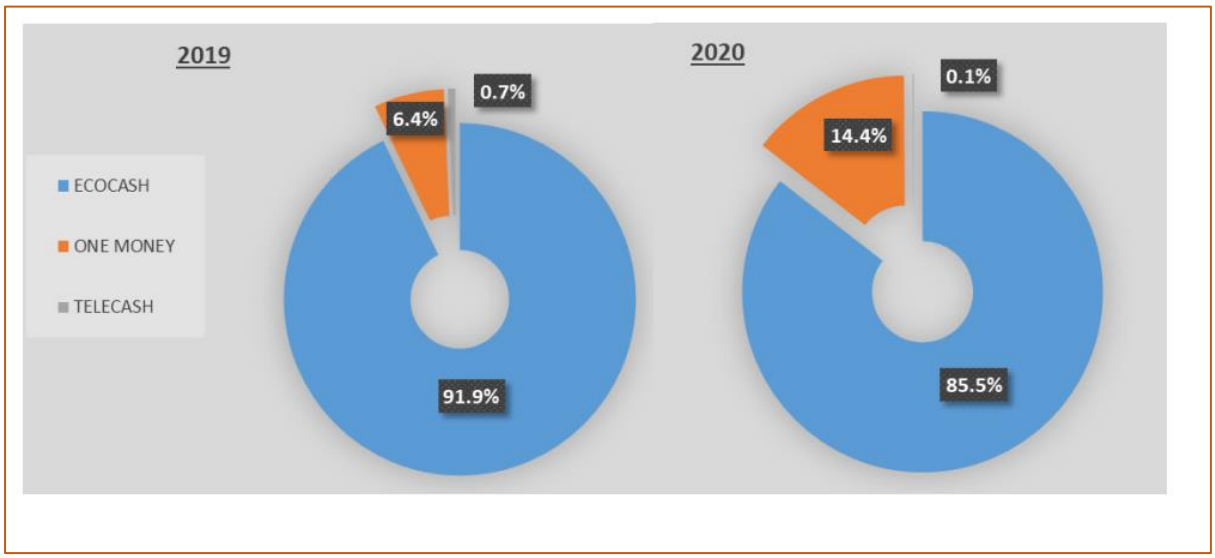


Figure 2: Active Mobile Money Subscriptions Market Share (POTRAZ, 2020)

Ecocash and Telecash, as seen above, lost market share by 6.4 percent and 0.6 percent, respectively, due to a drop in active mobile money subscriptions. OneMoney, on the other hand, increased its market share by 8%. This drop in market share by Econet needs to be investigated to ensure a positive gain in mobile money market share.

2.6 Sentiment Analysis Approaches

Several approaches can be used to perform sentiment analysis tasks. According to D'Andrea et al. (2015), sentiment analysis is mainly based on Machine learning, Lexicon and hybrid approaches. Reagan et al. (2017) found out that dictionary-based sentiment analysis methods are more robust for classifying longer texts. However, Shayaa et al. (2018) observed that sentiment analysis and opinion mining face several challenges ranging from classifying documents with opinionated material, making query classification and presenting sentiment data in a comprehensible manner.

2.7 Text Classification methods

A set of researchers (Gargiulo et al., 2019) analysed a Deep learning architecture that is devoted to text classification and proposed a methodology called Hierarchical Label Set Expansion (HLSE). The methodology was presented to regularize data labels and it was found as a useful methodology. However, the methodology brought more complex problems as a result of the higher label set of each sample. Furthermore, employing deep neural networks for hierarchical text categorization is difficult due to the vast quantity of training data needed and the method's inability to select acceptable document levels in a hierarchical situation (Meng et al., 2019).

Deep neural networks, both with and without word embeddings, have recently exhibited considerable advantages over typical machine learning–based approaches when applied to various sentence- and document-level classification issues. (Kim, 2014) demonstrated that utilizing basic static word embeddings and hyper-parameter tuning, on a variety of tasks, including sentiment classification, question type classification, and subjectivity classification, CNNs surpass traditional machine learning–based approaches. For text classification, (Zhang, Wang, & Liu, 2018) proposed character-level CNNs. For addressing a text classification problem, (Lai, Pan, Liu, & Yan, 2015) presented recurrent CNNs, whereas Johnson and Zhang (2015) proposed semi-supervised CNNs. Tang et al. (2015) improved on a sentiment classification task by using a document classification technique based on recurrent neural networks (RNNs). (Palangi, et al., 2015) suggested sentence embedding for an information retrieval challenge in which an LSTM network was used. For a relation classification challenge, (Zhou, et al., 2016) suggested attention-based, bidirectional LSTM networks. On Twitter stance detection data, (Augenstein, Rocktaschel, Vlachos, & Bontcheva, 2016) used a weakly supervised conditional LSTM encoding method to stance identification for unseen targets and demonstrated enhanced results. RNNs successfully model text sequences by capturing long-range relationships between words. When compared to CNN and SVM-based methods, LSTM

algorithms based on RNNs successfully capture the sequences in the sentences. However, all these models lacked the power of combining CNN and LSTM based algorithms to come up with a hybrid model known as CNN-LTM model.

2.7.1 Long Short-Term Memory (LTSM)

LTSM is an advanced model of RNN which comprises of a cell gate that moves information to the whole sequence, the forget gate which screens relevant information to retain, input gate, which moves relevant information to add to the current timestamp and an output gate that looks at the value of the output at current timestamp as alluded by (Del Pra, 2020).

In LSTMs there is a cell state S_t that conveys a flow of information from one module to the next. The transmission from S_t to S_{t+1} is regulated by components called gates. The data that is removed from the cell state is regulated by a forget gate layer. This layer uses Y_{t-1} and the input into the network x_t concatenated together into a matrix. This is multiplied by associated weights (by the dot product) and a bias term is added. This value is then inserted into a sigmoid activation function (σ) to attain a value between [0, 1] and multiplied with S_{t-1}. A value of zero means that no information should be transmitted within the cell state and a value of one means that all information should be transmitted (Olah, 2015).

(13) $f_t = (W_f \cdot [Y_{t-1}, x_t] + b_f)$

The second type of gate is an input gate (i_t) together with a tanh layer (G_t) which transforms values to the interval [-1, 1]. This section of the module determines what information that should be stored within the cell state.

(14) $\tanh(x) = \frac{2}{1+e^{-2x}} - 1$

(15) $i_t = \sigma(W_i \cdot [Y_{t-1}, x_t] + b_i)$

(16) $G_t = \tanh(W_c \cdot [Y_{t-1}, x_t] + b_c)$

The input gates decide which values that should be updated, and the tanh layer determines candidate values that could be added to the cell state.

The new cell state at S_t is then calculated using equations (13, 15, 16).

(17)$S_t = f_t * S_{t-1} + i_t * G_t$

2.7.1.1LTSM Architecture

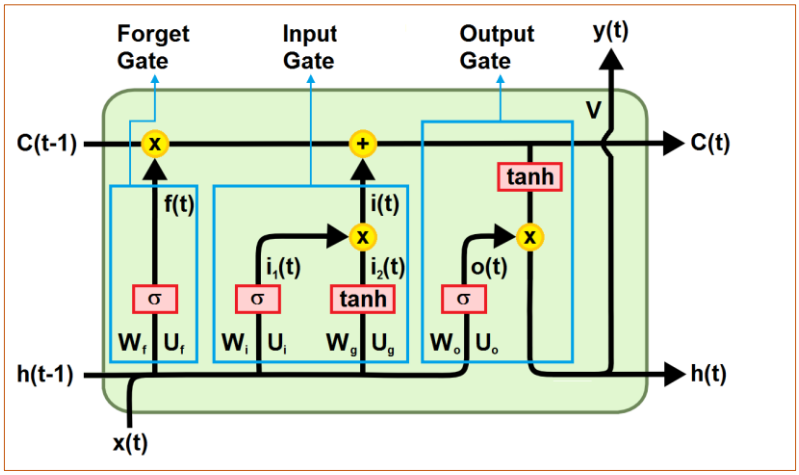


Figure 3: LTSM Architecture (Y. Zhu & Xiong, 2015)

Zhang et al. (2019) proposed LSTM RNN to comprehensively make use of the fault propagation information and it was found that the LSTM RNN can reliably predict the remaining useful life of a bearing by identifying its deterioration phases. Another research by Zhang et al. (2018) has made use of an approach that was based on the LSTM network. The method was designed to find hidden patterns in time series and was proposed with the goal of tracking system degradation and, as a result, predicting remaining useful life (RUL). LSTM neural network was also used in forecasting the concentration of air pollutants (X. Li et al., 2017). Usage of this technology by such recent researchers shows its relevance in the field and it is necessary for us to have a look at it in our research.

In another recent research by Zhou et al. (2019), in order to anticipate solar power generation in a time series way, two LSTM neural networks were deployed. Furthermore, for the two LSTM neural networks, we used the attention mechanism to adaptively focus on input features that are more important in forecasting.

2.8 Convolutional Neural Networks (CNN)

The CNN is a deep neural network that employs layers with convolution filters to process a batch of data (Kim, 2014). (Collobert, et al., 2011) designed the CNN architecture, however

Kim's (2014) research slightly altered it. In order to build local characteristics surrounding each word of the sentence, (Collobert, et al., 2011) considered the entire input sentences that would be transmitted to the lookup table layer. Convolutional layers integrate these features into a global feature vector, which can subsequently be given to fully connected layers.

Yao et al. (2019) proposed the use of graph convolutional networks for text classification. Their experiments demonstrated that Text Graph Convolutional Network (Text GCN) is better than usual methods for text classification without external word embeddings of knowledge. Meng et al. (2019) proposed a weakly-supervised neural network for classification of hierarchical text. This method was suggested since no large amount of training data is required.

Yao et al. (2019), proposed an approach that merge rule-based features and knowledge-guided deep learning models. Recognizing trigger phrases, predicting classes, and training a convolutional neural network utilizing word embeddings and Unified Medical Language System (UMLS) entity embeddings are all part of the proposed method. Therefore, the CNN model can be used to discover useful hidden features, while CUI embeddings can be used to create clinical text representations (Yao et al., 2019). This highlights the value of including domain knowledge into CNN models. However, Chen et al. (2019), proposed deep short text classification with knowledge powered attention (STCKA). With the goal of collecting weight of concepts from the aspects, the proposed method makes use of Concept towards Short Text (CST) attention and Concept towards Concept Set (CCS) attention.

Below is Convolutional Neural Network for sentence classification:

2.8.1 Convolutional Neural Network for Sentence Classification

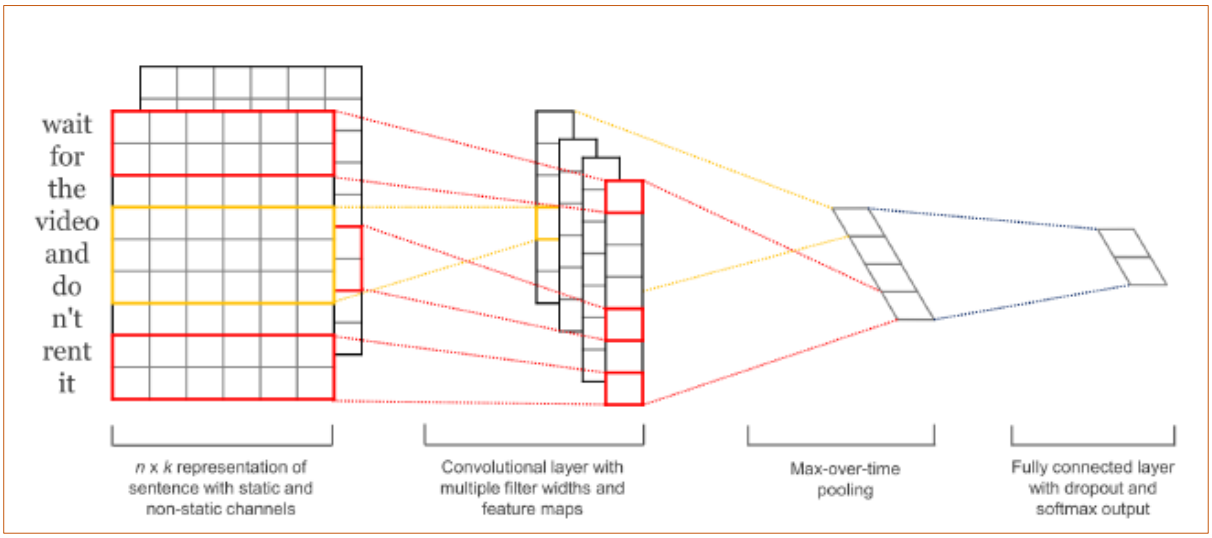


Figure 4: Convolutional Neural Network for Sentence Classification (Sosa, 2017)

The researcher discovered that there is limited literature in the use of CNN in sentiment analysis. In a research paper titled "Twitter Sentiment Analysis using combined LSTM-CNN Models" CNN achieved an average of 66.7% (Sosa, 2017) which implies that CNN is not the best approach for sentiment analysis. However, if you combine it with other models like LSTM, it can achieve a high accuracy level.

2.8.2 Sentiment analysis models accuracy comparison

Neural Network Model	Avg. Accuracy
Feed-Forward (Word Embeddings) [1]	58.4%
Feed-Forward (Feature Vectors) [1]	66.8%
CNN	66.7%
LSTM	72.5%
CNN-LSTM	69.7%
LSTM-CNN	75.2%

Table 2.2: Sentiment analysis models accuracy comparison (Sosa, 2017)

2.8.3 LSTM-CNN Model

The initial LSTM layer in our CNN-LSTM model will receive word embeddings for each token in the tweet as inputs. The assumption is that the LSTM layer's output tokens will store information from both the start and preceding tokens; in other words, the LSTM layer will

generate a new encoding for the original input. The LSTM layer's output is then passed into a convolution layer, which should extract local features. Finally, the output of the convolution layer will be pooled to a smaller dimension and output as a positive or negative label.

A hybrid system which combines LSTM network and CNN was suggested by Tan et al. (2018) for the detection of coronary artery disease (CAD) Electrocardiogram (ECG) signals and the system had high diagnostic accuracy of 99.85%. Therefore, LSTM can be improved if used with CNN. Pei et al. (2020) found that LSTM-CNN is a promising algorithm as they tested it in predicting real-time crash risk on arterials. Similarly, Zhu et al. (2020) concurred that LSTM-CNN methods are promising to give better results, however, when it comes to investigating spatial-temporal relational information, LSTM-CNN approaches have likewise been proven to be limiting. Therefore, end-to-end bidirectional LSTM-CNN (BiLSTM-CNN) was proposed as a spatial-temporal model.

It was stated by Xia et al.(2020) that LSTM is more suitable for processing temporal sequences, adding global average pooling (GAP) layer improves performance as it reduces model parameters by replacing fully connected layer after convolution Speeding up the convergence and results can be achieved through batch normalisation. On average, this technique was found to have accuracy of 94.75%. LSTM-CNN was tested for recognizing dynamic gesture based on surface electromyography (sEMG) signal and it was found to have accuracy of 98.14% (Wu et al., 2019).

Several more researchers (Liu et al. (2019); Song et al. (2019); Zhu et al. (2019); Li et al. (2018); Vo et al. (2017)) have successfully used LSTM-CNN in their different classification researches and it was found to be so helpful and accurate. It can be concluded from literature that, combining LSTM with CNN improves classification accuracy in diverse research domains. Moreover, modifications and additions to this method were also found to be helpful at improving its performance.

2.8.4 LSTM-CNN Model architecture

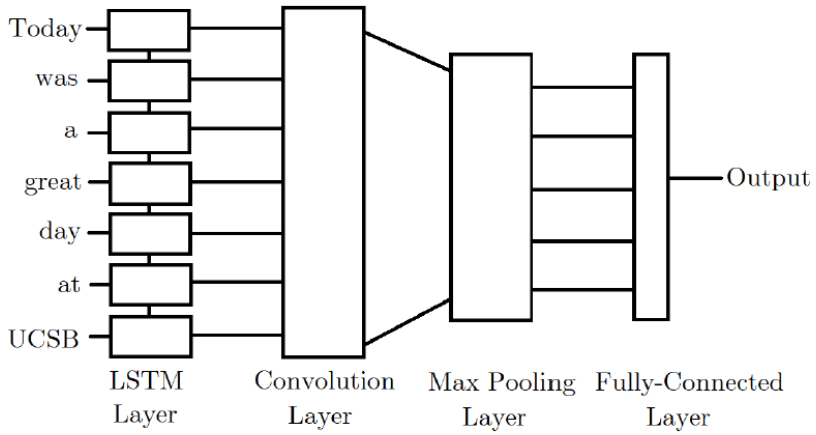


Figure 5: LSTM-CNN Model Architecture (Sosa, 2017)

2.9 Customer Churn related Work

The study reviewed a set of related works done by previous researchers so as to build an understanding of the way in which other researchers understood the problem and how they solved it. The reviewed works are as discussed in the following sub-paragraphs.

Ahmad et al. (2019) recently carried impressive research that is inviolably related to this research. The research focused at coming up with a model that customers that are more likely going to be part of customer churn in the field of telecom. The model was based on social networks data analysis and the dataset used to train, test and evaluate had customers information spanning over 9 months. The model experimented with four algorithms including XGBoost, Gradient Boosted Machine Tree, Decision tree, Extreme Gradient Boosting and Random Forest. XGBoost showed the best results as the performance of the model was measured using Area Under the Curve (AUC). Our research intends to look at Zimbabwean telecom companies and will also experiment the model with related algorithms. In different research, it was reported that including textual data in a customer churn prediction model enhances its performance, Convolutional Neural Networks are the best in text data mining for customer churn prediction and unstructured textual data alone is not sufficient enough for a model that can compete with models that make use of traditionally structured data (Caigny et al., 2019).

Another recent research looked at Cross-Company Churn Prediction (CCCP) which focused on one company making use of another company's data to predict its own customer churn

successfully (Amin, Shah, et al., 2019). It was however specified that the most effective method for data transformation in telecoms is not yet clear and the authors have used different data transformation methods including z-score, box-cox, rank and log. Classifiers used include K-Nearest Neighbor (KNN), Naïve Bayes (NB), Deep Learning Neural net (DP), Single rule induction (SRI), Gradient boosted tree (GBT). This research inculcates the use of several Mobile Money operators' data in our research to improve the performance of our model. In different research, it was found that lower distance test set (LDT) samples perform better as compared to upper distance test set (UDT) samples (Amin, Al-Obeidat, et al., 2019). Comparison was based on precision, f-measures, accuracy, and recall.

2.10 Customer Churn prevention methods

This study studied how the active use of mobile money services can prevent (in discouraging) customer turnover by examining the mediation influence of active usage of mobile money services on the relationship between satisfaction and trust on customer continuation intention. According to findings, customer happiness has a significant beneficial impact on both active mobile money usage and customer retention intentions. Previous research has shown that satisfaction is an important determinant of continued behaviour (Kim, Hong, Min, & Lee, 2011; Kuo et al., 2009; Liu et al., 2011). As a result, satisfaction should be the primary tool used by telecommunication service providers to encourage active usage and loyalty in order to reduce churn. Both active mobile money usage and customers' long-term intentions towards service providers are positively correlated with trust. This backs up Liu et al (2011) claim that establishing user trust is critical for the growth of mobile technology-enabled services.

Finally, given that several works around sentiment analysis has been done, primarily using various Machine Learning techniques, this research will focus more on investigating the possibility of using a deep learning hybrid algorithm CNN_LSTM as it harnesses the power of both LSTM and CNN algorithms with a proven high accuracy level as evidenced by previous researches.

2.11Chapter Summary

The study's literature review was reported in this chapter. Explanation of how sentiment analysis was established and how it works, existing sentiment analysis strategies, application of machine learning in sentiment analysis area, combination of sentiment analysis and machine learning modelling, detailed theory behind RNN, CNN, and LSTM models, and finally why we are using the root of a hybrid algorithm LSTM-CNN on our model development are among the literature covered in the chapter. The Chapter also gave a review of related works which provided an understanding of the problem and application of the concepts as applied by other researchers. The next Chapter presents the research methodology adopted by the study in solving the problem.

CHAPTER 3: Research Methodology

3.0 Introduction

The preceding Chapter gave an outline of the study's literature review, which incorporated sentiment analysis ideas. The research methodology, which is a description of the methodologies utilized in data collection, analysis, and interpretation of the study's findings in line with the research objectives, is the subject of the third chapter. This chapter covers the study's philosophy and research design, as well as research methodology and materials and the operationalization process. The chapter also covers data collecting and analysis, as well as the study's reliability and validity, as well as ethical aspects.

3.1 Research Philosophy

The study adopted the positivist philosophy since the study requires a more scientific approach and this philosophy also assumes that knowledge gathered via measurements is reliable and gives an argument that research should collect and evaluate data objectively (Collins, 2010). More so, in applying this philosophy, the researcher maintained an impartial attitude by being distant, unbiased, and unaffected by the subject of the study.

3.2 Research Paradigm

The research adopted the quantitative research paradigm which focuses on statistical methods to quantitatively collect and analyse data about a given research phenomenon (Creswell, 2014). The chosen paradigm is in line with the positivism research philosophy chosen which is quantitative in nature. This study is so deeply rooted in machine learning and statistics, quantitative research can effectively translate data into easily quantifiable charts and graphs.

3.3 Research Design

The study adopted the six staged Cross Industry Standard process for Data Mining (CRISP-DM) model for data science. The stages of the model are Problem Understanding, Data Understanding, Data Preparation, Modelling, Evaluation and Deployment and these stages simplify the research while fostering better understanding of the process. The CRISP-DM model as depicted as shown in Figure 6.

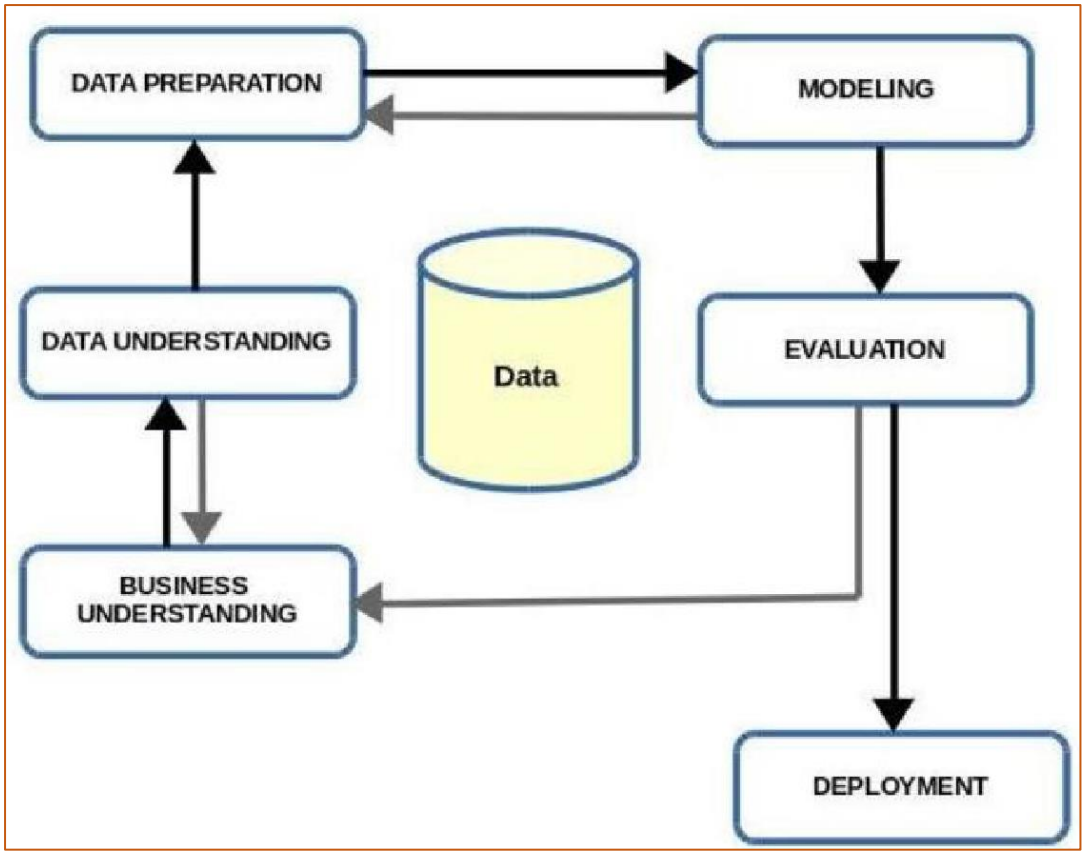


Figure 6: CRISP-DM Model (Source: Kopicka (2021))

3.3.1 Business understanding

The Business Understanding phase is of paramount importance as it is done to bring out the business problem and provide a business focus for the project. The study therefore sought to solve the problem of increased customer churning or customer attrition rate which was assumed to be impacting negatively on the market share of Ecocash Mobile Money Operator thus impacting business revenue. The objective of this research was to produce a machine learning model which analyses customer sentiments and market share trends as well as forecasting future market share trends based on the historical data.

3.3.2 Data Understanding

Data understanding comprises gathering information relevant to the problem at hand and ensuring that it is fit to address the problem under investigation. In other words, this phase involves data gathering, data description, data exploration and verification of data quality. The results of data understanding are used to investigate the data interpretation. A detailed

exploratory data analysis (EDA) was performed to identify data quality problems if any and get more detailed insights on the data gathered.

3.3.2.1 Data Gathering

3.3.2.1.1 Active Mobile Money Subscriber Data

The data for active mobile money subscribers for Ecocash, Telecash, OneMoney was collected from POTRAZ Quarterly reports for the period 2016 to 2021 which are found on POTRAZ official website. Below is a sample for the report section with the Active Mobile Money Subscribers data.

OPERATOR	4th Quarter 2019	1st Quarter 2020	Variance (%)
ECOCASH	6,812,368	7,065,382	3.7%
TELECASH	53,311	52,564	-1.4%
ONE MONEY	468,960	555,255	18.4%
TOTAL	**7,334,639**	**7,673,201**	**4.6%**

Table 3.1: Active Mobile Money Subscriptions Sample Data (POTRAZ, 2020)

3.3.2.1.2 Tweeter Data

The tweeter data was collected from tweeter for the period January 2016 to October 2021 using twint command. Below is a screenshot of a twint command for extracting Ecocash Mobile Money Operator data and saving data into a CSV file:

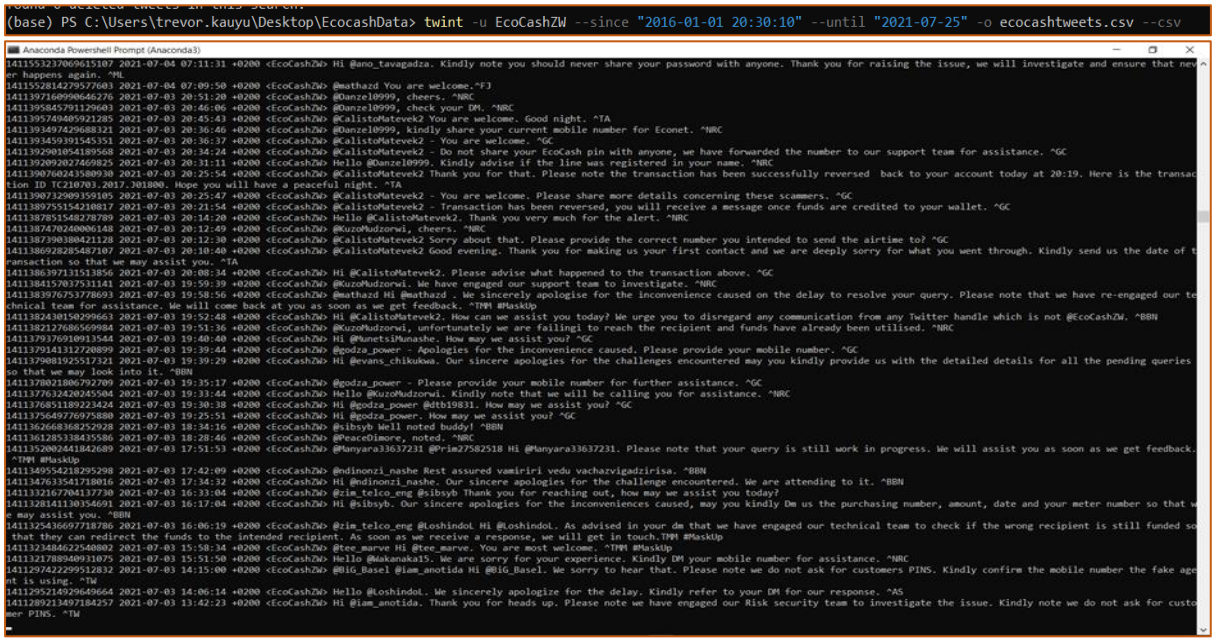


Figure 7: Scraping twitter data

3.3.2.3 Exploratory Data Analysis (EDA)

The dataset was then loaded into the notebook workspace to perform strong EDA using the snippet code below:

```
[109] data1 = pd.read_csv('/content/drive/My Drive/ecocashdata2.csv', sep="\t")
      data2 = pd.read_csv('/content/drive/My Drive/ecocash_twitter_JAN2017_to_JUN2021.csv', index_col=0)

      data=pd.concat([data1,data2])

      /usr/local/lib/python3.7/dist-packages/IPython/core/interactiveshell.py:2718: DtypeWarning: Columns (10) have mixed types.Specify dtype option on import or set low_memory=False.
        interactivity=interactivity, compiler=compiler, result=result)
```

Figure 8: Loading data sets in notebook

Below is the list of columns for the dataset.

```
[ ]  df.columns

     Index(['id', 'conversation_id', 'created_at', 'date', 'time', 'timezone',
            'user_id', 'username', 'name', 'place', 'tweet', 'language', 'mentions',
            'urls', 'photos', 'replies_count', 'retweets_count', 'likes_count',
            'hashtags', 'cashtags', 'link', 'retweet', 'quote_url', 'video',
            'thumbnail', 'near', 'geo', 'source', 'user_rt_id', 'user_rt',
            'retweet_id', 'reply_to', 'retweet_date', 'translate', 'trans_src',
            'trans_dest'],
           dtype='object')
```

Figure 9: Dataset columns

After loading the data, the dataset reading was checked by viewing the head and tail of the data as read in python. Figure 11 and Figure 12 shows results of checking on the loaded dataset.

```
data.head()
```

	id	conversation_id	created_at	date	time	timezone	user_id	username	name	place	tweet	language	mentions	urls
0	1454558148522565641	1454557178174611458	2021-10-30 23:17:41 South Africa Standard Time	2021-10-30	23:17:41	200	2900688735	bustoptv	#HealthPeopleHealthNation	NaN	@drDendere Thank you Dr mune Ecocash here	en	[]	[]
1	1454549529668292608	1454549529668292608	2021-10-30 22:43:26 South Africa Standard Time	2021-10-30	22:43:26	200	3334888083	drdendere	Chipo Dendere	NaN	Guys if you're enjoying #wadiwawepamoyo you ca...	en	[]	[]
2	1454519309947084802	1454355609105670144	2021-10-30 20:43:21 South Africa Standard Time	2021-10-30	20:43:21	200	358977716	niggamumu	Nigga	NaN	@_NobleSavage Car park in town ndinenge ndisin...	en	[]	[]
3	1454509738343944195	1454443372962361351	2021-10-30 20:05:19 South Africa Standard Time	2021-10-30	20:05:19	200	1654770775	ecocashzw	EcoCash Zimbabwe	NaN	@TAWANDAMACHING4 @Ecocash40541216 Issue is bei...	en	[]	[]
4	1454498022457491463	1454493262551650312	2021-10-30 19:18:46 South Africa Standard Time	2021-10-30	19:18:46	200	1654770775	ecocashzw	EcoCash Zimbabwe	NaN	@FaithSharleen Good day. Thanks for reaching o...	en	[]	[]

22s completed at 12:53 PM

Figure 10: Dataset head

```
data.tail()
```

	id	conversation_id	created_at	date	time	timezone	user_id	username	name	place	tweet	language	me
407919	682806052795105280	682674914080473088	1451628623000	2017-01-01	08:10:23	South Africa Standard Time	1602751363	econet_support	Econet Customer Care	NaN	Airtime transfer is free @NashMcRonzie , pleas...	NaN	['nashm
407920	682797771036954624	682797771036954624	1451626649000	2017-01-01	07:37:29	South Africa Standard Time	820894273	chinakaclive	Stay at home	NaN	@econet_support mandibira ka apa	NaN	['econet_s
407921	682797739906838528	682645314956832771	1451626641000	2017-01-01	07:37:21	South Africa Standard Time	1602751363	econet_support	Econet Customer Care	NaN	Compliments of the new season @DivineGunner , ...	NaN	['divine
407922	682797512219070464	682797512219070464	1451626587000	2017-01-01	07:36:27	South Africa Standard Time	221668086	dorkatcooltable	Walking Stick Man	NaN	@econet_support if i ecocash the wrong number ...	NaN	['econet_s
407923	682797487116140544	682797487116140544	1451626581000	2017-01-01	07:36:21	South Africa Standard Time	820894273	chinakaclive	Stay at home	NaN	@econet_support hd 66c & I was tryin to cal sm...	NaN	['econet_s

22s completed at 12:53 PM

Figure 11: Dataset tail

Both the head and tail of the dataset confirm the same number and ordering of features as the dataset that was retrieved and presented in the csv data files. Both the head and tail are validating eleven features that were identified throughout the dataset's feature check.

An exploration for the data was performed to gain understanding of all the data types in the data set. Descriptive statistics were carried out on all data columns to have a better understanding of the count of values, unique value, frequency, minimum, mean, maximum, standard deviation etc.

The screenshot below shows descriptive statistics of the data:

```
[ ] data.describe(include='all')
```

	id	conversation_id	created_at	date	time	timezone	user_id	username	name	place	tweet	language	mentions	urls	photos
count	4.094530e+05	4.094530e+05	4.094530e+05	409453	409453	409453	4.094530e+05	409453	409439	8	409453	1529	409453	409453	409453
unique	NaN	NaN	3.705930e+05	1657	68691	2	NaN	36262	34354	8	342170	23	80269	5564	10627
top	NaN	NaN	1.527712e+12	2021-05-05	11:32:37	South Africa Standard Time	NaN	econet_support	Econet Customer Care	Trek Petroleum	@econet_support	en	['econet_support']	[]	[]
freq	NaN	NaN	8.000000e+00	2314	28	407924	NaN	141153	141153	1	1279	1146	124605	397508	397573
mean	1.098245e+18	1.097089e+18	NaN	NaN	NaN	NaN	2.419641e+17	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
std	1.468933e+17	1.471067e+17	NaN	NaN	NaN	NaN	4.229796e+17	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
min	6.827975e+17	3.770077e+17	NaN	NaN	NaN	NaN	2.958951e+06	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
25%	1.043721e+18	1.042756e+18	NaN	NaN	NaN	NaN	1.602751e+09	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
50%	1.121117e+18	1.120569e+18	NaN	NaN	NaN	NaN	1.602751e+09	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
75%	1.203342e+18	1.202188e+18	NaN	NaN	NaN	NaN	7.033058e+17	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
max	1.454558e+18	1.454557e+18	NaN	NaN	NaN	NaN	1.452625e+18	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN

22s completed at 12:53 PM

Figure 12: Dataset descriptive statistics

Our data set has 409453 rows and 39 columns. After running some descriptive statistics on this data, we can see that no column contains null values as each column has a count of 409453.

Let's explore the size and shape of the data.

Figure 13: Dataset shape and size

The data set contains 409453 rows and 36 columns

Panda's code df.dtypes explains the types of the data columns for the data to be analyzed.

Figure 14: Dataset data types

3.3.3 Data preparation and cleaning

Data preparation and cleaning entails the preparation of the data to make it suitable for use in the intended model. On data cleaning, things like punctuations, numbers and special characters were removed as they do not hold value in sentiment analysis. More so, hyperlinks, links to pictures, trailing 'co' and/or 'zw' and everything that is not a character or hashtag were also removed.

3.3.3.1 Selecting relevant features

There are features on our data that aren't all that useful to our objective in this project, so it won't hurt to drop them. These include 'tweet_id', 'user_id' and 'time_zone', so we picked only crucial columns which include date, likes_count, tweet and so on.

```
data=data[['date','likes_count','replies_count','retweets_count','tweet','username','mentions']]
```

```
[8] data.head(3)
```

	date	likes_count	replies_count	retweets_count	tweet	username	mentions
0	2021-10-30	1	2	0	@drDendere Thank you Dr mune Ecocash here	bustoptv	[]
1	2021-10-30	31	1	12	Guys if you're enjoying #wadiwawepamoyo you ca...	drdendere	[]
2	2021-10-30	1	0	0	@_NobleSavage Car park in town ndinenge ndisin...	niggamumu	[]

Figure 15: Selecting relevant features

3.3.3.2 Descriptive statistics

Descriptive statistics after selecting crucial columns from the data is shown below:

```
[9] data.describe(include='all')
```

	date	likes_count	replies_count	retweets_count	tweet	username	mentions
count	409453	409453.000000	409453.000000	409453.000000	409453	409453	409453
unique	1657	NaN	NaN	NaN	342170	36262	80269
top	2021-05-05	NaN	NaN	NaN	@econet_support	econet_support	['econet_support']
freq	2314	NaN	NaN	NaN	1279	141153	124605
mean	NaN	0.383458	0.697052	0.083863	NaN	NaN	NaN
std	NaN	4.254655	1.722642	2.261421	NaN	NaN	NaN
min	NaN	0.000000	0.000000	0.000000	NaN	NaN	NaN
25%	NaN	0.000000	0.000000	0.000000	NaN	NaN	NaN
50%	NaN	0.000000	1.000000	0.000000	NaN	NaN	NaN
75%	NaN	0.000000	1.000000	0.000000	NaN	NaN	NaN
max	NaN	984.000000	574.000000	849.000000	NaN	NaN	NaN

Figure 16: Descriptive statistics of crucial columns

3.3.3.3 Removing tweets by official MMO accounts

If our twitter search returns tweets by accounts belonging to Ecocash, Cassava Smartech, Econet, OneMoney, NetOne, Telecel or StewardBank, we need to exclude them.

```
[112] exclude_rows = data[(data['username']=='econet_support') | (data['username']=='econetzimbabwe') | (data['username']=='EcoCashZW') | (data['username']=='stewardbank')| (data['username'
      print(exclude_rows.shape[0])
      exclude_rows.head(3)
```

```
145290
```

	date	likes_count	replies_count	retweets_count	tweet	username	mentions
5	2021-10-30	0	2	0	Hi @Dlodlo_princess, its a true story. Send a ...	econet_support	[{'screen_name': 'dlodlo_princess', 'name': 'c...
24	2021-10-30	0	2	0	Hi @vallshng, please share the buying number, ...	econet_support	[{'screen_name': 'vallshng', 'name': 'vall', '...
33	2021-10-30	1	0	1	Swipe into EcoCash using your bank card on any...	stewardbank	[]

Figure 17: View data to be excluded

The data shows that these results do exist, and it has 145290 rows. Upon scrutiny of the text column, we can see that it has nothing to do with inquiries which originated from a customer, so we can remove the rows since we do not need the data.

```
[113] exclude_list = list(exclude_rows.index.values)
      data.drop(exclude_list, inplace=True)

[41] data.shape[0]

     263923
```

Figure 18: Removing unwanted data

3.3.3.4 Removing tweets without specific keywords

Next, we excluded all the other rows or tweets which do not contain the keywords like Ecocash, Econet, Cassava, Telecash and so on from the data and selected only those which contain the key words.

```
# filtering the rows
include_rows = data[(data['tweet'].str.contains('ecocash')|data['tweet'].str.contains('EcoCash')|data['tweet'].str.contains('@EcoCash')|data['mentions'].str.contains('@ecocashzw')|dat
print(include_rows.shape[0])

34485

[ ] data=include_rows
    print(data.shape[0])
    data.head()

34485
```

	date	likes_count	replies_count	retweets_count	tweet	username	mentions
1	2021-10-30	31	1	12	Guys if you're enjoying #wadiwawepamoyo you ca...	drdendere	[]
2	2021-10-30	1	0	0	@_NobleSavage Car park in town ndinenge ndisin...	niggamumu	[]
6	2021-10-30	0	2	0	@EcoCashZW please do explain how do I access m...	faithsharleen	[]
20	2021-10-30	0	1	0	@TAWANDAMACHING4 @CabsZimbabwe @EcoCashZW Hell...	ecocash40541216	[]
26	2021-10-30	0	3	0	@CabsZimbabwe Good day to your. I sent Zipit t...	tawandamaching4	[{'screen_name': 'ecocashzw', 'name': 'ecocash...

```
[ ] exc_list = list(exc_rows.index.values)
    data.drop(exc_list, inplace=True)
    print(data.shape[0])
    data.head()

34442
```

	date	likes_count	replies_count	retweets_count	tweet	username	mentions
1	2021-10-30	31	1	12	Guys if you're enjoying #wadiwawepamoyo you ca...	drdendere	[]
2	2021-10-30	1	0	0	@_NobleSavage Car park in town ndinenge ndisin...	niggamumu	[]
6	2021-10-30	0	2	0	@EcoCashZW please do explain how do I access m...	faithsharleen	[]
20	2021-10-30	0	1	0	@TAWANDAMACHING4 @CabsZimbabwe @EcoCashZW Hell...	ecocash40541216	[]
26	2021-10-30	0	3	0	@CabsZimbabwe Good day to your. I sent Zipit t...	tawandamaching4	[{'screen_name': 'ecocashzw', 'name': 'ecocash...

Figure 19: Selecting and viewing crucial rows

Our data frame now has 34485 rows remaining, down from the original 409453. The non crucial rows have now been dropped.

3.3.3.5 Removing philanthropic and donation tweets

In Zimbabwe, Ecocash is a very big Mobile Money Operator. As a result, it is often involved in philanthropy or disaster relief efforts. Examples are the recent 'Cyclone Idai' disaster. As a result, we should remove as many of these instances as we can.

Let's add a column which notes whether key words are found in a tweet.

```
[ ] def find_key_words(x):
        r = re.compile(r'\bcyclone\b | \bebola\b | \bidai\b | \bdonate\b | \bdonation\b | \bnatural disaster\b | \bcorporate social responsibility\b | \bdonations\b  | \bphilanthropy\b| \
        if r.findall(x):
            return 1
        else:
            return 0
```

```
[ ] data['key_word_cleaner'] = data['tweet'].apply(find_key_words)
```

```
[ ] data.tail()
```

	date	likes_count	replies_count	retweets_count	tweet	username	mentions	key_word_cleaner
407725	2017-01-06	0	0	0	@econet_support guys may you please give me fe...	luvjiz52	['econet_support']	0
407759	2017-01-05	0	1	0	@econet_support you just took money from my ec...	luvjiz52	['econet_support']	0
407776	2017-01-05	0	1	0	@econet_support whats up with these random eco...	nhlallo	['econet_support']	0
407887	2017-01-01	0	0	0	@econet_support why am I charged $1.00 for the...	faraij	['econet_support']	0
407922	2017-01-01	0	1	0	@econet_support if i ecocash the wrong number ...	dorkatcooltable	['econet_support']	0

```
[ ] data[data['key_word_cleaner']==1].shape[0]
78
```

Figure 20: Checking existence of donations tweets

We can see that there are 78 rows which contain keywords, and their variants, for donating cyclone ideas, scholarship, philanthropy, etc. Let us go ahead and drop them.

```
[ ] data = data[data['key_word_cleaner']==0]
```

```
[ ] data.shape[0]
34364
```

Figure 21: Removing donations tweets

Our number of rows has dropped from 34485 to 34364.

To keep up with our data cleaning, so far, we have removed:

- tweets that were tweeted by accounts belonging to the company(s) we are investigating,
- tweets which involve Econet, Ecocash, Telecel, Telecash, OneMoney and Netone were performing philanthropic acts.

The second point could have been achieved later when we remove tweets which have a positive sentiment (after running an algorithm). However, after going through some tweets, there are some philanthropic tweets which contain negative terms like 'sad' and 'bad' (describing cyclone

Idai, for example). These would have confused the sentiment analysis algorithm and possibly produced a false negative. The above method just adds to our thoroughness.

Let us perform our descriptive stats again and see if we can find any red flags.

3.3.3.6 Removing outliers

```
[ ] data.describe(include='all')
```

	date	likes_count	replies_count	retweets_count	tweet	username	mentions	key_word_cleaner
count	34364	34364.000000	34364.000000	34364.000000	34364	34364	34364	34364.0
unique	1563	NaN	NaN	NaN	30136	13329	5817	NaN
top	2021-06-01	NaN	NaN	NaN	@EcoCashZW	ecocashzw	['econet_support']	NaN
freq	378	NaN	NaN	NaN	126	920	10031	NaN
mean	NaN	0.565155	0.954691	0.168898	NaN	NaN	NaN	0.0
std	NaN	6.438463	1.762591	1.946328	NaN	NaN	NaN	0.0
min	NaN	0.000000	0.000000	0.000000	NaN	NaN	NaN	0.0
25%	NaN	0.000000	0.000000	0.000000	NaN	NaN	NaN	0.0
50%	NaN	0.000000	1.000000	0.000000	NaN	NaN	NaN	0.0
75%	NaN	0.000000	1.000000	0.000000	NaN	NaN	NaN	0.0
max	NaN	744.000000	145.000000	171.000000	NaN	NaN	NaN	0.0

Figure 22: Descriptive statistics to check red flags

Looking at the distribution of our tweets, we can see that the minimum number of likes, and retweets is both 0. That is understandable and expected. However, the maximum numbers of likes and retweets are 744 and 145, respectively. It is doubtful that a tweet regarding enquiry of service would have such high numbers. It is then probably an outlier. The IQR suggests that most of our tweets have likes ranging between 2 & 24 and retweets, ranging between 2 & 11. This is much more believable. Let us investigate further visually.

```
[ ] mydata=data[['likes_count', 'retweets_count', 'replies_count']]
    mydata.plot(kind='box',title='Outlier Checking on Likes,Retweets & Replies counts',vert=0,figsize=(10,6))
    plt.xticks(rotation=25)
    plt.yticks(rotation=25)
    plt.tight_layout()
    plt.show()
```

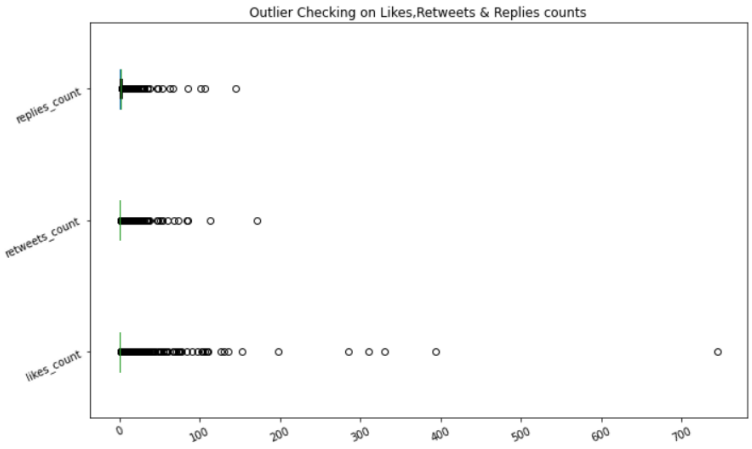


Figure 23: Likes, retweets and replies visualization

The outliers are so significant that the scale of our boxplots is compromised.

Let us look at the data associated with these outliers. From the graph above, we can see that they begin at over 300 replies (safe bet that these tweets are also the ones with the significant retweets).

```
[ ] data[data['likes_count']>300]
```

	date	likes_count	replies_count	retweets_count	tweet	username	mentions	key_word_cleaner
453	2021-10-22	744	145	171	The CID has come alive. They are investigating...	ngarivhumejacob	[]	0
4809	2021-05-22	394	31	48	This demonization of @EcoCashZW when we know w...	advocatemahere	['ecocashzw']	0
5770	2021-05-19	310	66	83	How to make money in Zimbabwe: \n\n- Get an Ag...	ptchimusoro	['ecocashzw']	0
113350	2020-11-18	330	101	85	Shity ecocash what does this even mean. Fire y...	josephmakuni	['ecocashzw', 'econetzimbabwe', 'econet_support']	0

Figure 24: Tweets with likes over 300

Only six rows are returned. Let us investigate the text and see what they contain. (Note that this is also the tweets responsible for the retweets outlier)

```
[ ] data[data['likes_count']>300]['tweet']

453       The CID has come alive. They are investigating...
4809      This demonization of @EcoCashZW when we know w...
5770      How to make money in Zimbabwe: \n\n- Get an Ag...
113350    Shity ecocash what does this even mean. Fire y...
Name: tweet, dtype: object
```

Figure 25: Further view of Tweets with likes over 300

The tweets seem to contain retweets which were not loaded fully. Let us delete it all the same as it does not provide any useful information for us.

```
[ ] data = data[data['likes_count']<300]

[ ] data[data['likes_count']>300]

    date  likes_count  replies_count  retweets_count  tweet  username  mentions  key_word_cleaner
```

Figure 26: Verify if outliers still exist

The tweet no longer exists in our data frame. Let us plot our box plot again

```
[ ] mydata=data[['likes_count', 'retweets_count', 'replies_count']]
    mydata.plot(kind='box',title='Outlier Checking on Likes,Retweets & Replies counts',vert=0,figsize=(10,6))
    plt.xticks([20, 40, 60, 80, 100, 120, 140, 160, 180, 200, 220, 240, 260, 280,300])    # changing x scale by own
    plt.xticks(rotation=25)
    plt.yticks(rotation=25)
    plt.tight_layout()
    plt.show()
```

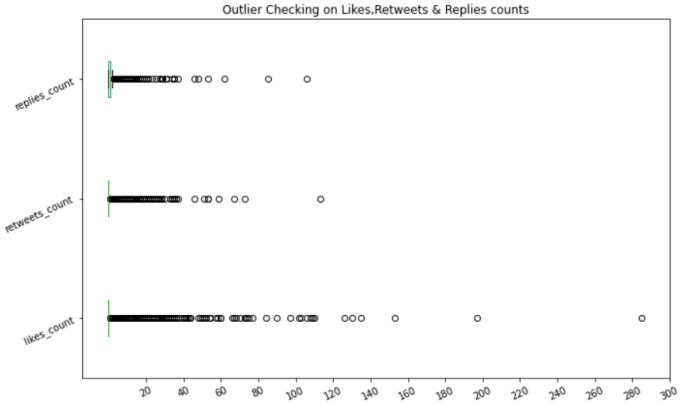


Figure 27: Visualizing and checking for outliers

Three tweets with over 140 likes still exist. Let us also look at the tweets.

```
data[data['likes_count']>140]
```

	date	likes_count	replies_count	retweets_count	tweet	username	mentions	key_word_cleaner
4570	2021-05-23	153	8	21	So a top official of @EcoCashZW who is a membe...	pedzisairuhanya	['ecocashzw']	0
4837	2021-05-22	197	35	67	1. The Financial Intelligence Unit is charging...	daddyhope	['ecocashzw']	0
226552	2020-03-25	285	106	20	hie @EcoCashZW @econet_support I bought $1 dai...	faffiemanhuhwa	['ecocashzw', 'econet_support']	0

Figure 28: Filtering likes with more than 140 count

Let’s go ahead and drop the tweets since they can introduce noise in the data.

```
[ ] data = data[data['likes_count']<140]

[ ] data[data['likes_count']>140]
```

date	likes_count	replies_count	retweets_count	tweet	username	mentions	key_word_cleaner

Figure 29: Removing outliers

The tweet is not talking anything about Ecocash business and is instructing twitter users to retweet a #Twimbos hashtag. We can safely drop it from the dataset.

```
[ ] mydata=data[['likes_count', 'retweets_count', 'replies_count']]
    mydata.plot(kind='box',title='Outlier Checking on Likes,Retweets & Replies counts',vert=0,figsize=(10,6))
    plt.xticks([20, 40, 60, 80, 100, 120, 140, 160, 180, 200, 220, 240, 260, 280,300])    # changing x scale by own
    plt.xticks(rotation=25)
    plt.yticks(rotation=25)
    plt.tight_layout()
    plt.show()
```

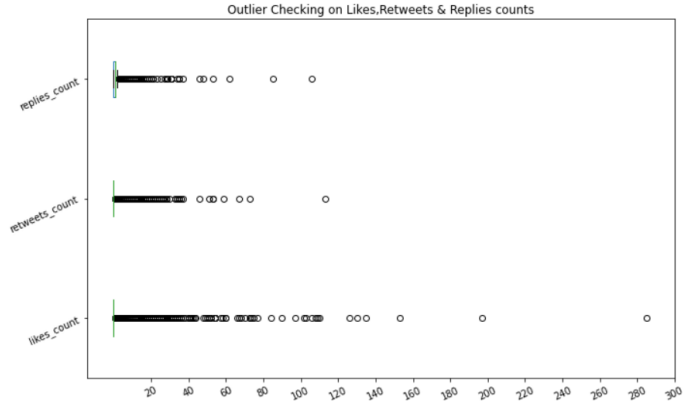


Figure 30: Checking outliers after dropping unwanted columns

As seen in the box plots above, our data still contains many outliers. Let us investigate all tweets with likes over 140, to begin with.

```
data[data['likes_count']>140]
```

	date	likes_count	replies_count	retweets_count	tweet	username	mentions	key_word_cleaner
4570	2021-05-23	153	8	21	So a top official of @EcoCashZW who is a membe...	pedzisairuhanya	['ecocashzw']	0
4837	2021-05-22	197	35	67	1. The Financial Intelligence Unit is charging...	daddyhope	['ecocashzw']	0
226552	2020-03-25	285	106	20	hie @EcoCashZW @econet_support I bought $1 dai...	faffiemanhuhwa	['ecocashzw', 'econet_support']	0

Figure 31: Investigating tweets with more than 140 likes

A quick scroll through the tweets shows that most of them are very legitimate but the researcher noticed that a few of these tweets are about internet shutdown. It'd be very biased to include this data in the sentiment analysis as it is something Ecocash or Econet had no control over. So, let's remove all rows that contain the phrases 'internet shutdown', 'shutdownzimbabwe', 'whiletheinternetwasoff', etc.

```
[ ] def find_key_words_2(x):
        r = re.compile(r'\#whiletheinternetwasoff | \#shutdownzimbabwe', flags=re.I | re.X)
        if r.findall(x):
            return 1
        else:
            return 0

[ ] data['key_word_cleaner_2'] = data['tweet'].apply(find_key_words_2)

[ ] data[data['key_word_cleaner_2']==1].head()
```

	date	likes_count	replies_count	retweets_count	tweet	username	mentions	key_word_cleaner	key_word_cleaner_2
18696	2021-03-23	0	0	1	No goin back on #ShutDownZimbabwe2016 on Wed 1...	bornfreezim	['ntandon', 'kangara89', 'ecocashzw', 'econet_...	0	1
73486	2021-03-23	0	0	1	No goin back on #ShutDownZimbabwe2016 on Wed 1...	bornfreezim	['ntandon', 'kangara89', 'ecocashzw', 'econet_...	0	1
91650	2021-01-13	0	0	1	No goin back on #ShutDownZimbabwe2016 on Wed 1...	bornfreezim	['tmafundikwa', 'ecocashzw', 'econet_support',...	0	1
94780	2020-12-28	0	0	1	No goin back on #ShutDownZimbabwe2016 on Wed 1...	bornfreezim	['chumet_', '263chat', 'econet_support', 'ecoc...	0	1
97712	2020-12-17	0	0	1	No goin back on #ShutDownZimbabwe2016 on Wed 1...	bornfreezim	['lebirdchief', 'the_thinker_ii', 'alingulube1...	0	1

Figure 32: Investigating tweets about national shutdown

```
[ ]  data[data['key_word_cleaner_2']==1].shape[0]

     11
```

The data returned above is that with the hashtags #shutdownzimbabwe and #WhileTheInternetWasOff. There are only 11 rows.

Let's transform our likes, retweets and replies columns to a shorter scale so that we can investigate the shape of the spread of our data.

```
[ ]  import numpy as np

     data["log(likes)"] = data["likes_count"] + 7
     data["log(likes)"] = data["log(likes)"].apply(np.log)

     data["log(retweets)"] = data["retweets_count"] + 7
     data["log(retweets)"] = data["log(retweets)"].apply(np.log)

     data["log(replies)"] = data["replies_count"] + 7
     data["log(replies)"] = data["log(replies)"].apply(np.log)
```

```
data.head()
```

date	likes_count	replies_count	retweets_count	tweet	username	mentions	key_word_cleaner	key_word_cleaner_2	key_word_cleaner_3	log(likes)	log(retwee
2021-10-30	1	0	0	@_NobleSavage Car park in town ndinenge ndisin...	niggamumu	[]	0	0	0	2.079442	1.94
2021-10-30	0	2	0	@EcoCashZW please do explain how do I access m...	faithsharleen	[]	0	0	0	1.945910	1.94
2021-10-30	0	1	0	@TAWANDAMACHING4 @CabsZimbabwe @EcoCashZW Hell...	ecocash40541216	[]	0	0	0	1.945910	1.94
2021-	0	3	0	@CabsZimbabwe Good day to your. I sent Zipit	tawandamaching4	[{'screen_name': 'ecocashzw',	0	0	0	1.945910	1.94

Figure 33: Transformation of likes, retweets and replies

```
[ ]  mydata=data[['likes_count', 'retweets_count', 'replies_count']]
     mydata.plot(kind='box',title='Outlier Checking on Likes,Retweets & Replies counts after scalling',vert=0,figsize=(10,6))
     plt.xticks([20, 40, 60, 80, 100, 120, 140, 160, 180, 200, 220, 240, 260, 280,300])    # changing x scale by own
     plt.xticks(rotation=25)
     plt.yticks(rotation=25)
     plt.tight_layout()
     plt.show()
```

Figure 34: Checking for outliers after transformation

For both likes and retweets, the data in the upper quartile seems to be more spread than that in the bottom. The replies seem to be more evenly balanced around the median. Let us get a rough idea of the number of outliers we have compared to the rest of our data.

```
[ ] data.describe()
```

	likes_count	replies_count	retweets_count	key_word_cleaner	key_word_cleaner_2	key_word_cleaner_3	log(likes)	log(retweets)	log(replies)
count	32773.000000	32773.000000	32773.000000	32773.0	32773.000000	32773.0	32773.000000	32773.000000	32773.000000
mean	0.498520	0.939767	0.156348	0.0	0.000275	0.0	1.990910	1.960779	2.064853
std	3.251046	1.306000	1.540860	0.0	0.016570	0.0	0.166641	0.094965	0.107703
min	0.000000	0.000000	0.000000	0.0	0.000000	0.0	1.945910	1.945910	1.945910
25%	0.000000	0.000000	0.000000	0.0	0.000000	0.0	1.945910	1.945910	1.945910
50%	0.000000	1.000000	0.000000	0.0	0.000000	0.0	1.945910	1.945910	2.079442
75%	0.000000	1.000000	0.000000	0.0	0.000000	0.0	1.945910	1.945910	2.079442
max	135.000000	85.000000	113.000000	0.0	1.000000	0.0	4.955827	4.787492	4.521789

```
[ ] print('Outliers: {}'.format(data[data['log(likes)']>5.2].shape[0]))
    print('Rest of data {}'.format(data.shape[0]-data[data['log(likes)']>5.2].shape[0]))

    Outliers: 0
    Rest of data 32773
```

Figure 35: Descriptive statistics on the dataset

Our descriptive stats show that 'key_word_cleaner_3' has a maximum value of 0.

The is no tweets existing for the keyword

```
data[(data['username']=='EcoSureZW') | (data['username']=='elevateyouthzw')].head()
```

date	likes_count	replies_count	retweets_count	tweet	username	mentions	key_word_cleaner	key_word_cleaner_2	key_word_cleaner_3	log(likes)	log(retweets)	log(replies)

Figure 36: Dropping unwanted data rows

We have cleaned our data by removing irrelevant rows to our analysis as much as possible given the information that we know so far. Further rows can be deleted later when new insights appear. For now, this is the best we can do.

The researcher also dropped the columns that we no longer needed, see code snippet below:

```
[ ] data.drop(columns=['key_word_cleaner', 'key_word_cleaner_2', 'key_word_cleaner_3',
          'log(likes)', 'log(retweets)', 'log(replies)','mentions'], inplace=True)
```

```
[ ] data.head(3)
```

date	likes_count	replies_count	retweets_count	tweet	username
2021-10-30	1	0	0	@_NobleSavage Car park in town ndinenge ndisin...	niggamumu
2021-10-30	0	2	0	@EcoCashZW please do explain how do I access m...	faithsharleen
2021-10-30	0	0	0	@takudzwamapako1 @econet_support @narshy3842 H...	ecocash40541216

Figure 37: Dropping unwanted columns

3.3.3.7 Removing Punctuations, Numbers, and Special Characters

The researcher removed punctuations like (.,'?), special characters like (%$#@) and numbers and substituted them with spaces except for characters and hashtags since they are not useful in sentiment analysis. The researcher also removed hyperlinks and to trail 'co' and 'zw'. We also want to remove links to twitter pictures.

First, we need to import the required libraries for cleaning tweets

```
[ ]  # import necessary packages

     import string
     import nltk

     warnings.filterwarnings("ignore", category=DeprecationWarning)
```

Figure 38: Importing required libraries

```
[ ] data['tidy_tweet'] = data['tweet'].apply(lambda x: re.sub(r'http\S+', '', x))
    data['tidy_tweet'] = data['tidy_tweet'].apply(lambda x: re.sub(r'pic\.\S+', '', x))

[ ] data['tidy_tweet'] = data['tidy_tweet'].str.replace("[^a-zA-Z#]", " ")
    data['tidy_tweet'] = data['tidy_tweet'].apply(str.lower)

[ ] data['tidy_tweet'] = data['tidy_tweet'].apply(lambda x: re.sub(r'\bco\b|\bzw\b|twitter', '', x))

[ ] data.head()
```

date	likes_count	replies_count	retweets_count	tweet	username	tidy_tweet
2021-10-30	1	0	0	@_NobleSavage Car park in town ndinenge ndisin...	niggamumu	noblesavage car park in town ndinenge ndisin...
2021-10-30	0	2	0	@EcoCashZW please do explain how do I access m...	faithsharleen	ecocashzw please do explain how do i access m...
2021-10-30	0	0	0	@takudzwamapako1 @econet_support @narshy3842 H...	ecocash40541216	takudzwamapako econet support narshy h...
2021-10-30	0	0	0	@traeyung @econetzimbabwe @NetOneCellular Hell...	ecocash40541216	traeyung econetzimbabwe netonecellular hell...
2021-10-30	0	1	0	@TAWANDAMACHING4 @CabsZimbabwe @EcoCashZW Hell...	ecocash40541216	tawandamaching cabszimbabwe ecocashzw hell...

Figure 39: Removing punctuations, Numbers and Special characters

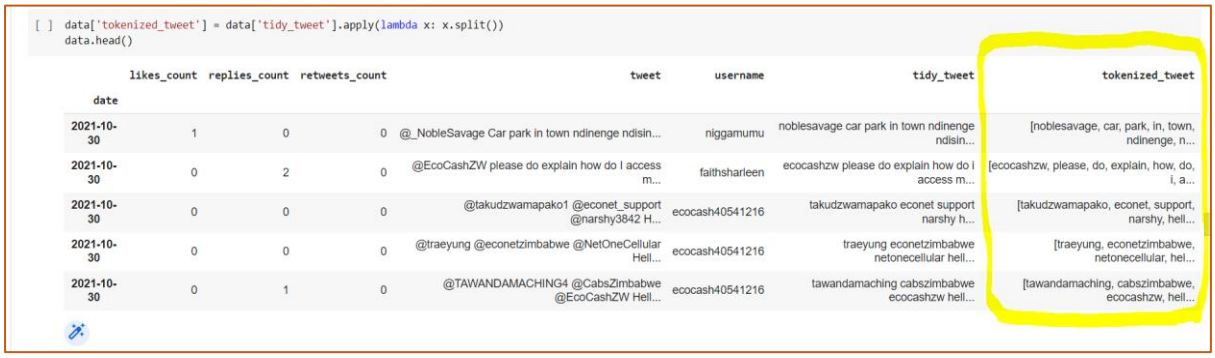
3.3.3.8 Tokenization

Singh (2019) defines tokenization as a way of splitting a sentence, or text document into word segments named tokens. An example below illustrates how tokenization shall be performed.

[‘Ecocash, ‘please, ‘help, ‘reverse, ‘my’, ‘failed, ‘transaction, ‘#’, ‘@’].

Tokens could be words, numbers or punctuation marks as shown above. This is done by locating word boundaries. Tokenization is done soon before stemming and lemmatization in the preprocessing stage. Tokenization is essential since it makes it easy to understand the meaning and sentiment of a text when it is shown in a single word that is grouped. Putting together tokens is then known as the bag of words that we then use to train our classifier (Singh, 2019).

As demonstrated above, tokens can be words, numbers, or punctuation marks. This is accomplished by determining the limits of words. In the pre-processing stage, tokenization comes before stemming and lemmatization. Tokenization is important because it makes it simple to understand the content and feeling of a text when it is displayed as a single word. The bag of words that we use to train our classifier is made up of tokens that have been put together (Singh, 2019). Tokenization can be done in a variety of methods, however in this example, we'll use the natural language toolkit.



```
[ ] data['tokenized_tweet'] = data['tidy_tweet'].apply(lambda x: x.split())
    data.head()
```

date	likes_count	replies_count	retweets_count	tweet	username	tidy_tweet	tokenized_tweet
2021-10-30	1	0	0	@_NobleSavage Car park in town ndinenge ndisin...	niggamumu	noblesavage car park in town ndinenge ndisin...	[noblesavage, car, park, in, town, ndinenge, n...
2021-10-30	0	2	0	@EcoCashZW please do explain how do I access m...	faithsharleen	ecocashzw please do explain how do i access m...	[ecocashzw, please, do, explain, how, do, i, a...
2021-10-30	0	0	0	@takudzwamapako1 @econet_support @narshy3842 H...	ecocash40541216	takudzwamapako econet support narshy h...	[takudzwamapako, econet, support, narshy, hell...
2021-10-30	0	0	0	@traeyung @econetzimbabwe @NetOneCellular Hell...	ecocash40541216	traeyung econetzimbabwe netonecellular hell...	[traeyung, econetzimbabwe, netonecellular, hel...
2021-10-30	0	1	0	@TAWANDAMACHING4 @CabsZimbabwe @EcoCashZW Hell...	ecocash40541216	tawandamaching cabszimbabwe ecocashzw hell...	[tawandamaching, cabszimbabwe, ecocashzw, hell...

Figure 40: Tokenization of dataset

Our tweets are now much cleaner and better to apply a sentiment analysis algorithm on. However, we still don't know which tweets belong to which mobile money operator. Let's find a way to label the tweets.

- ✓ ecocash: 'e'
- ✓ onemoney or telecash: 'c'
- ✓ other: 0

If a tweet contains 'ecocash', 'onemoney' and\or 'telecash' within it, we attribute the tweet as a complaint to ecocash and a threat to switch to another provider.

3.3.3.9 Categorizing and Labelling tweets

```
def labeller(x):
    if (re.search('ecocash', x) and re.search('onemoney', x)) or (re.search('ecocash', x) and re.search('telecash', x)):
        return 'e'
    elif  re.search('ecocash', x) or re.search('steward', x) or re.search('cassava', x):
        return 'e'
    elif re.search('telecash', x) or re.search('onemoney', x):
        return 'c'
    else:
        return 0
```

```
[ ] data["MMoneyOperator_label"] = data['tidy_tweet'].apply(labeller)
```

```
[ ] data['MMoneyOperator_label'].value_counts()

e    32487
c      197
0       89
Name: MMoneyOperator_label, dtype: int64
```

Figure 41: Labelling tweets according to Mobile Money Operator

From the tweets labelling, we can observe that ecocash has 32487 tweets, onemoney and telecash combined has 197 tweets and the remaining 89 tweets could not be labelled under any Mobile Money Operator and had to be classified under Other.

```
data[data['MMoneyOperator_label']==0].tail(10)
```

date	likes_count	replies_count	retweets_count	tweet	username	tidy_tweet	tokenized_tweet	MMoneyOperator_label
2019-05-12	0	0	0	Hi, for terms and conditions of the promotion ...	ecocashzw	hi for terms and conditions of the promotion ...	[hi, for, terms, and, conditions, of, the, pro...	0
2019-05-10	0	0	0	You need a minimum of 5 points to stand a chan...	ecocashzw	you need a minimum of points to stand a chan...	[you, need, a, minimum, of, points, to, stand,...	0
2018-09-24	0	1	0	@econet_support bought 2gig data early this mr...	itsemantee	econet support bought gig data early this mr...	[econet, support, bought, gig, data, early, th...	0
2018-09-17	0	0	0	Hi SterKinetar. Please take the opportunity to...	ecocashzw	hi sterkinetar please take the opportunity to...	[hi, sterkinetar, please, take, the, opportuni...	0
2018-09-17	0	0	0	Hi Hon Mapahla. Please take the opportunity to...	ecocashzw	hi hon mapahla please take the opportunity to...	[hi, hon, mapahla, please, take, the, opportun...	0
2018-05-05	0	1	1	Chakachaya for real loving this promotion bn r...	yeyendlovu	chakachaya for real loving this promotion bn r...	[chakachaya, for, real, loving, this, promotio...	0
2018-02-23	0	0	0	online transaction failed Number:0774408349.Vi...	samuraironald	online transaction failed number vi...	[online, transaction, failed, number, virtualc...	0
2017-06-03	0	1	0	Hi @OarabileOsborne @econet_support @econetzim...	ecocashzw	hi oarabileosborne econet support econetzim...	[hi, oarabileosborne, econet, support, econetz...	0
2017-04-27	0	1	0	@econet_support what is needed to integrate yo...	indundimag	econet support what is needed to integrate yo...	[econet, support, what, is, needed, to, integr...	0
2017-03-23	0	0	0	@CynzRChik @econet_support Find details on how...	ecocashzw	cynzrchik econet support find details on how...	[cynzrchik, econet, support, find, details, on...	0

Figure 42: Tweets labelled as Other

The tweets which do not contain either 'ecocash', 'telecash' or 'onemoney' seem to be replies to tweets to Ecocash or Econet. Let's drop them since they'd simply be a duplication of the original tweet in terms of value counts

```
[ ] data.shape[0]

32773
```

```
[ ] data = data[data['MMoneyOperator_label'] != 0]
```

```
[ ] data.shape[0]

32684
```

Figure 43: Removing tweets labelled as Other

3.3.3.10 Language checking and conversion

We are almost done with cleaning our text. However, we are aware that our tweets contain English, shona and'or ndebele. Although, the sentiment analyzer we're going to use skips over non-English words and looks for sentiment in the available English ones, this might not be accurate enough. Since there are no translators good enough to convert shona\ndebele to English now, we might need to drop the rows which have too much vernacular. Let's build a function that does this for us.

We need to first install and import the required libraries

```
[ ]  !pip install pyenchant
     import enchant


     Collecting pyenchant
       Downloading pyenchant-3.2.2-py3-none-any.whl (55 kB)
          |████████████████████████████████| 55 kB 3.8 MB/s
     Installing collected packages: pyenchant
     Successfully installed pyenchant-3.2.2
```

Figure 44: Installing and Importing required libraries

```
[ ]  def lang_checker(x):
         d = enchant.Dict("en_US")
         en_count=0
         sn_count=0
         for i in x:
             if d.check(i) is True:
                 en_count+=1
             else:
                 sn_count+=1
         if en_count>=sn_count:
             return True
         else:
             return False
```

Figure 45: Language checker function

```
[ ] data['lang'] = data['tokenized_tweet'].apply(lang_checker)
```

```
[ ] data.head()
```

date	likes_count	replies_count	retweets_count	tweet	username	tidy_tweet	tokenized_tweet	MMoneyOperator_label	lang
2021-10-30	1	0	0	@_NobleSavage Car park in town ndinenge ndisin...	niggamumu	noblesavage car park in town ndinenge ndisin...	[noblesavage, car, park, in, town, ndinenge, n...	e	True
2021-10-30	0	2	0	@EcoCashZW please do explain how do I access m...	faithsharleen	ecocashzw please do explain how do i access m...	[ecocashzw, please, do, explain, how, do, i, a...	e	True
2021-10-30	0	1	0	@TAWANDAMACHING4 @CabsZimbabwe @EcoCashZW Hell...	ecocash40541216	tawandamaching cabszimbabwe ecocashzw hell...	[tawandamaching, cabszimbabwe, ecocashzw, hell...	e	True
2021-10-30	0	3	0	@CabsZimbabwe Good day to your. I sent Zipit t...	tawandamaching4	cabszimbabwe good day to your i sent zipit t...	[cabszimbabwe, good, day, to, your, i, sent, z...	e	True
2021-10-30	3	0	0	@doughbyzo Do you do delivery and do you also ...	mrmuvezwa	doughbyzo do you do delivery and do you also ...	[doughbyzo, do, you, do, delivery, and, do, yo...	e	True

Figure 46: Applying language checker

```
[ ] "There are {} majority english tweets and {} majority non-english tweets.".format(data[data['lang']==True].shape[0], data[data['lang']==False].shape[0])
```

```
'There are 28455 majority english tweets and 4229 majority non-english tweets.'
```

Figure 47: Check English tweets and non-English tweets

After applying the language checker, we can see that 28455 are English tweets and 4229 non-English tweets

Now, let's visualize the non-English tweets to verify if the language checker has been applied correctly.

```
[ ] dat_conv=data[data['lang']==False]
dat_conv
```

date	likes_count	replies_count	retweets_count	tweet	username	tidy_tweet	tokenized_tweet	MMoneyOperator_label	lang
2021-10-30	1	0	0	@drjaytee87 @DarkForceGirl01 vakuona kuona fac...	eng_brianm	drjaytee darkforcegirl vakuona kuona fac...	[drjaytee, darkforcegirl, vakuona, kuona, face...	e	False
2021-10-30	1	1	0	@shanon_the_gem ini ndinotoipiwa ndisina kuiku...	moyoabigail2	shanon the gem ini ndinotoipiwa ndisina kuiku...	[shanon, the, gem, ini, ndinotoipiwa, ndisina,...	e	False
2021-10-30	1	0	0	Hunza ecocash number. https://t.co/bOfTXoCNdG	drankoceann	hunza ecocash number	[hunza, ecocash, number]	e	False
2021-10-29	1	1	0	@Tynoe97 bhoo wangu ecocash ukuiziva	lionel_sauro	tynoe bhoo wangu ecocash ukuiziva	[tynoe, bhoo, wangu, ecocash, ukuiziva]	e	False
2021-10-29	2	0	0	@smileessM @Dipholo_TJ Eseng ba nhlolise... ha...	nthawbiee	smileessm dipholo tj eseng ba nhlolise ha...	[smileessm, dipholo, tj, eseng, ba, nhlolise, ...	e	False
...	...	...	...	...	...	...	...	...	...
2017-02-03	0	0	0	@econet_support ehh guys refund yangu isai hen...	kudzi3_b3rry	econet support ehh guys refund yangu isai hen...	[econet, support, ehh, guys, refund, yangu, is...	e	False
2017-01-29	0	1	0	@econet_support :-):-) tryng to move ec...	divantierukwav1	econet support tryng to move ec...	[econet, support, tryng, to, move, ecocash, ac...	e	False
2017-01-26	3	0	2	Econet Wireless maintains top position http:/...	fingazlive	econet wireless maintains top position lov...	[econet, wireless, maintains, top, position, l...	e	False
2017-01-23	0	0	0	Econet Free Twitter is Back\n http://goo.gl/8J...	techunzipped	econet free is back econet support ecoca...	[econet, free, is, back, econet, support, ecoc...	e	False

Figure 48: Display non-English tweets

We've got non-English tweets in our data and to eliminate false positives when we carry out our sentiment analysis, let's drop them.

```
[ ]  data = data[data['lang']==True]

[ ]  data.shape[0]

     28455
```

Figure 49: Dropping non-English tweets

3.3.3.11 Analysis on tweets key words with word cloud.

Let's visualize our most common words so that we can have an idea of which words occur the most in our tweets. We can proceed to clean out some of these words or even delete entire tweets we deem unnecessary to our analysis. One way to do this is to use a wordcloud.

3.3.3.12 Plotting Word Cloud

Jin (2017) categorised word clouds into three classes, which are frequency, categorisation and mixed, accordingly we are going to utilise frequency which uses the size of fonts to represent appearances of a word, hence the more commonly a word shall be stated on twitter the more visually it will be represented. We can proceed to clean out some of these words or even delete entire tweets we deem unnecessary to our analysis

Importing the libraries and creating a function for plotting the word cloud.

```
[ ]  import cv2

     from urllib.request import urlopen
     from wordcloud import ImageColorGenerator

[ ]  def load_mask(mask_url):
         with urlopen(mask_url) as response:
             mask = np.asarray(bytearray(response.read()), dtype="uint8")
             mask = cv2.imdecode(mask, cv2.IMREAD_COLOR)
             mask = cv2.cvtColor(mask, cv2.COLOR_BGR2RGB)

         return mask
```

Figure 50: Creating wordcloud plot function

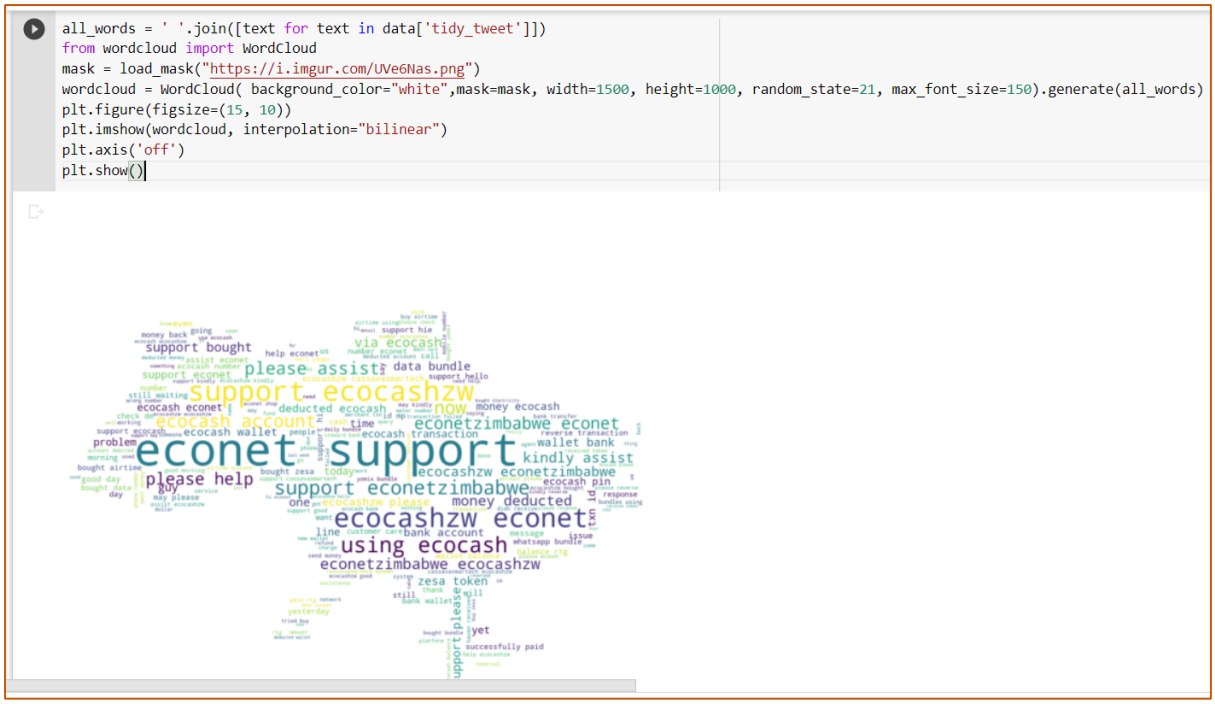


Figure 51: Plotting word cloud for cleaned data

We can see from our word cloud that Econet is overwhelmingly in the lead when it comes to word count. However, there are other variations of Ecocash like 'ecocashzimbabwe' and 'Econet support'. Let us delete their suffixes so that we remain with one instance of Ecocash. We can also delete extensions to words like telecash, onemoney, etc.

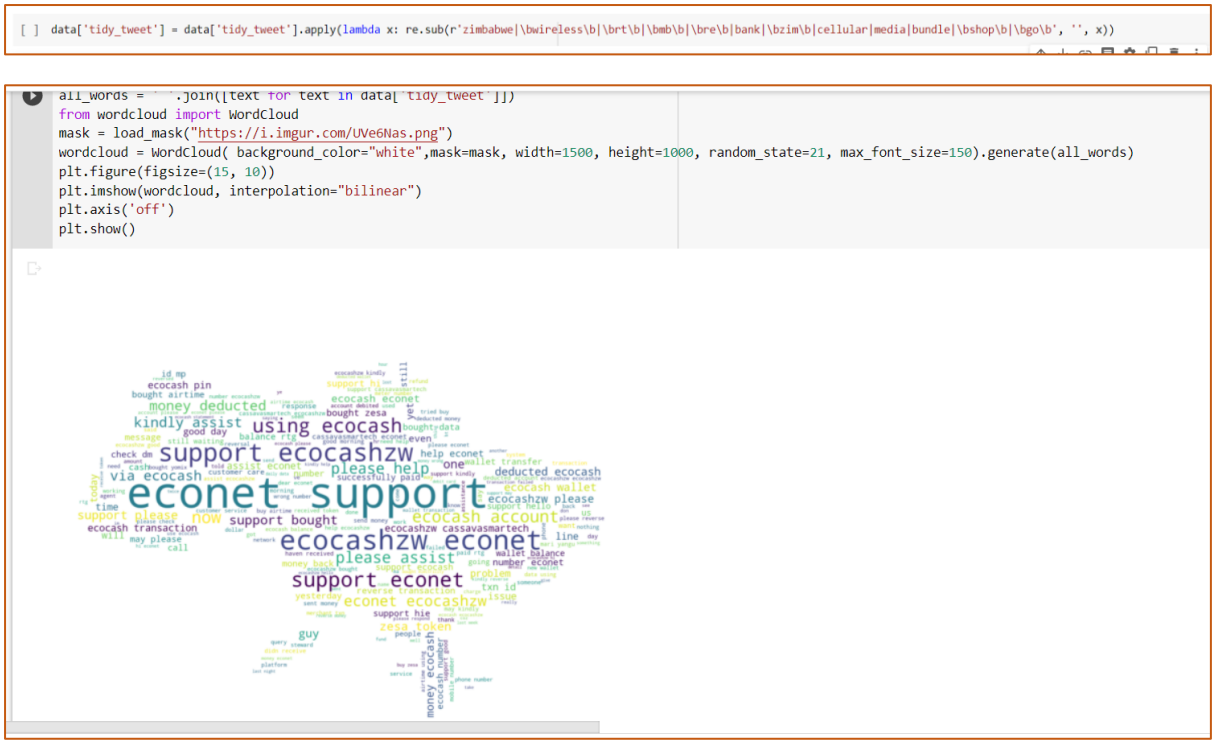


Figure 52: Plotting word cloud for cleaned data after further analysis

We've done all we reasonably can to remove data that isn't relevant to our analysis. To remove any remaining noise, let's apply a sentiment analysis algorithm to the remaining data to pick up data with a positive sentiment. This data is of no use to use since we want tweets that either represent enquiries to a service provider, or a complaint.

3.3.3.13 Polarity calculation

With the data scrapped so far, we went on to calculate our polarity. We chose to clean our data first and remove all unnecessary noise before calculating our polarity to focus our calculation on words that carry sentimental value and for memory management for quick processing. The table below therefore shows a new table generated with a new additional column named Polarity, which is meant for rating each word extracted within a scale of below zero and above zero, as polarity above zero represents a positive sentiment and at zero represents neutral sentiment and the one below zero represents a negative sentiment. Though they are various libraries that can be utilized, we have resorted to Vader sentiment analyzer.

```
nltk.download('vader_lexicon')

import nltk
from nltk.sentiment.vader import SentimentIntensityAnalyzer
sia = SentimentIntensityAnalyzer()
```

```
[nltk_data] Downloading package vader_lexicon to /root/nltk_data...
[nltk_data]   Package vader_lexicon is already up-to-date!
```

```
[ ] pol_score = []
    for tweet in data['tweet']:
        sent = sia.polarity_scores(tweet)
        pol_score.append(sent)
```

Figure 53: Applying Vader sentiment analyzer

After applying the Vader sentiment analyzer, let's view a sample tweet from the data and add the polarity score data to the original data frame.

```
[ ] pol = pd.DataFrame(pol_score)
    pol.head()
```

	neg	neu	pos	compound
0	0.000	1.000	0.000	0.0000
1	0.072	0.852	0.075	0.0258
2	0.000	0.787	0.213	0.4019
3	0.066	0.598	0.335	0.8625
4	0.000	0.794	0.206	0.3818

```
[ ] data['tweet'][4]

    '@doughbyzo Do you do delivery and do you also accept ecocash'

[ ] data['pol_score'] = list(pol['compound'])
```

Figure 54: Adding polarity score to the data frame

Let's plot average daily sentiments for the data.

```
#Plotting Average daily Sentiments
data['pol_score'].resample('M').mean().plot(figsize=(10,10))
plt.show()
```

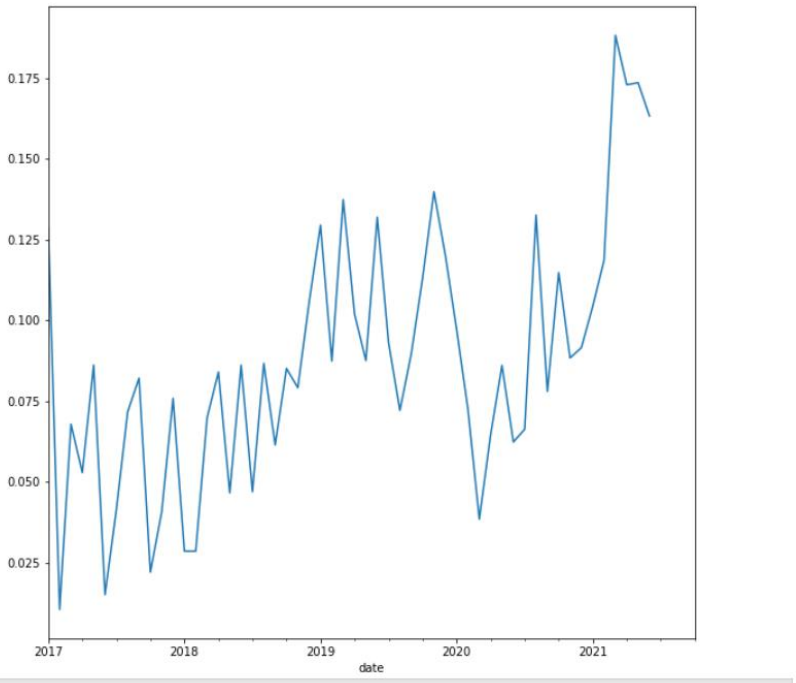


Figure 55: Plotting average monthly sentiments

3.3.3.14 Categorizing tweets

There is need for categorizing tweet sentiments into negative, positive, and neutral based on the calculated polarity score. Below is the snippet code for categorising the tweets.

```
[ ]  #Categorising Tweet Sentiments
     data['sentiment'] = np.where(data['pol_score']<0,'negative',np.where(data['pol_score']>0,'positive','neutral'))

     data_neg = data[data['sentiment']=='negative']
     data_pos = data[data['sentiment']=='positive']
     data_neu = data[data['sentiment']=='neutral']
```

Figure 56: Categorizing tweet sentiments

Now let's view the data to check the sentiments for the tweets after categorization of the tweets.

```
[ ] data.head()
```

date	likes_count	replies_count	retweets_count	tweet	username	tidy_tweet	tokenized_tweet	MMoneyOperator_label	lang	pol_score	sentiment
2021-10-30	1	0	0	@_NobleSavage Car park in town ndinenge ndisin...	niggamumu	noblesavage car park in town ndinenge ndisin...	[noblesavage, car, park, in, town, ndinenge, n...	e	True	0.0000	neutral
2021-10-30	0	2	0	@EcoCashZW please do explain how do I access m...	faithsharleen	ecocashzw please do explain how do i access m...	[ecocashzw, please, do, explain, how, do, i, a...	e	True	0.0258	positive
2021-10-30	0	1	0	@TAWANDAMACHING4 @CabsZimbabwe @EcoCashZW Hell...	ecocash40541216	tawandamaching cabs ecocashzw hello inbox ...	[tawandamaching, cabszimbabwe, ecocashzw, hell...	e	True	0.4019	positive
2021-10-30	0	3	0	@CabsZimbabwe Good day to your. I sent Zipit t...	tawandamaching4	cabs good day to your i sent zipit to a wron...	[cabszimbabwe, good, day, to, your, i, sent, z...	e	True	0.8625	positive
2021-10-30	3	0	0	@doughbyzo Do you do delivery and do you also ...	mrmuvezwa	doughbyzo do you do delivery and do you also ...	[doughbyzo, do, you, do, delivery, and, do, yo...	e	True	0.3818	positive

Figure 57: Check tweets sentiment

Let's look at the unique value counts of the values in the sentiment column.

```
[ ]  data['sentiment'].value_counts()

     positive    14342
     negative     7754
     neutral      6359
     Name: sentiment, dtype: int64
```

Figure 58: Sentiment counts

Let’s plot word cloud for negative sentiments.

```
word_cloud(pd.Series([text for text in data_neg.tweet]).str.cat(sep=' '))
```

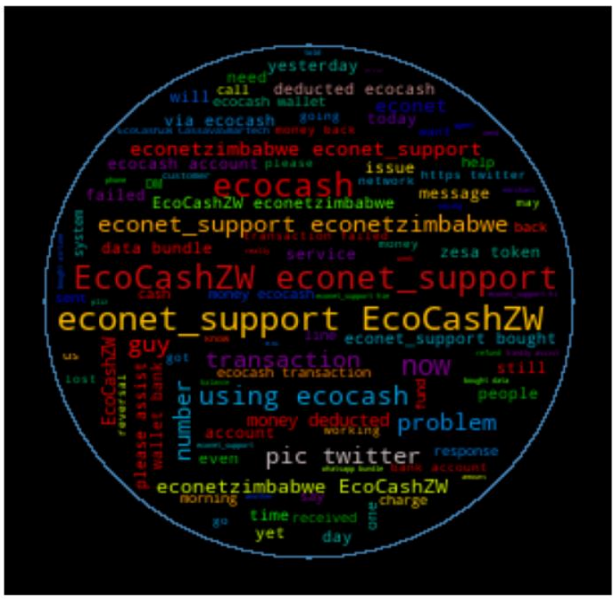


Figure 59: Word cloud for negative tweets

3.3.3.15 Active Mobile Money Subscribers Data

We now import the data for active mobile subscribers for the three Mobile Money Operators (MMO) that are Ecocash, OneMoney and Telecash. This data was obtained from POTRAZ Quarterly Reports from 2016 to 2020.

```
[ ]  mmnsdata = pd.read_csv('/content/drive/My Drive/MobileMoneyOperators_Data2.csv', index_col=0)
```

Figure 60: Import Active Mobile Money Subscriptions data

Let’s perform EDA on the Active Mobile Money Subscribers data. Let’s start by displaying the crucial information about the dataset.

```
[ ]  mmnsdata.info()

<class 'pandas.core.frame.DataFrame'>
Index: 20 entries, Q1_2016 to Q4_2020
Data columns (total 3 columns):
 #   Column    Non-Null Count  Dtype
---  ------    --------------  -----
 0   Ecocash   20 non-null     int64
 1   Telecash  20 non-null     int64
 2   OneMoney  20 non-null     int64
dtypes: int64(3)
memory usage: 640.0+ bytes
```

Figure 61: Categorizing tweet sentiments

Let’s perform some descriptive statistics on the data.

```
[ ]  mmnsdata.describe(include='all')
```

	Ecocash	Telecash	OneMoney
count	2.000000e+01	20.000000	20.000000
mean	5.050135e+06	99992.600000	233916.050000
std	1.468103e+06	197512.587152	279260.778622
min	3.121683e+06	19198.000000	5222.000000
25%	3.286062e+06	52442.000000	12508.250000
50%	5.427270e+06	54039.500000	115691.500000
75%	6.367791e+06	68223.750000	358481.250000
max	7.065382e+06	936479.000000	892963.000000

Figure 62: Descriptive Statistics for AMMS data

3.3.3.16 Merging Active Mobile Subscriber data with Twitter data

Let’s start by defining the labeller function for OneMoney

```
[ ]
    def labeller_OneMoney(x):
        if (re.search('2016-01', x) or re.search('2016-02', x)) or (re.search('2016-03', x)):
            return mmnsdata[mmnsdata['Period']=='Q1_2016']['OneMoney'].values[0]
        elif (re.search('2016-04', x) or re.search('2016-05', x)) or (re.search('2016-06', x)):
            return mmnsdata[mmnsdata['Period']=='Q2_2016']['OneMoney'].values[0]
        elif (re.search('2016-07', x) or re.search('2016-08', x)) or (re.search('2016-09', x)):
            return mmnsdata[mmnsdata['Period']=='Q3_2016']['OneMoney'].values[0]
        elif (re.search('2016-10', x) or re.search('2016-11', x)) or (re.search('2016-12', x)):
            return mmnsdata[mmnsdata['Period']=='Q4_2016']['OneMoney'].values[0]
            #----------------------2017-----------------------------------------------------#
        elif (re.search('2017-01', x) or re.search('2017-02', x)) or (re.search('2017-03', x)):
            return mmnsdata[mmnsdata['Period']=='Q1_2017']['OneMoney'].values[0]
        elif (re.search('2017-04', x) or re.search('2017-05', x)) or (re.search('2017-06', x)):
            return mmnsdata[mmnsdata['Period']=='Q2_2017']['OneMoney'].values[0]
        elif (re.search('2017-07', x) or re.search('2017-08', x)) or (re.search('2017-09', x)):
            return mmnsdata[mmnsdata['Period']=='Q3_2017']['OneMoney'].values[0]
        elif (re.search('2017-10', x) or re.search('2017-11', x)) or (re.search('2017-12', x)):
            return mmnsdata[mmnsdata['Period']=='Q4_2017']['OneMoney'].values[0]
            #---------------------2018-----------------------------------------------------#
        elif (re.search('2018-01', x) or re.search('2018-02', x)) or (re.search('2018-03', x)):
            return mmnsdata[mmnsdata['Period']=='Q1_2018']['OneMoney'].values[0]
        elif (re.search('2018-04', x) or re.search('2018-05', x)) or (re.search('2018-06', x)):
            return mmnsdata[mmnsdata['Period']=='Q2_2018']['OneMoney'].values[0]
        elif (re.search('2018-07', x) or re.search('2018-08', x)) or (re.search('2018-09', x)):
            return mmnsdata[mmnsdata['Period']=='Q3_2018']['OneMoney'].values[0]
        elif (re.search('2018-10', x) or re.search('2018-11', x)) or (re.search('2018-12', x)):
            return mmnsdata[mmnsdata['Period']=='Q4_2018']['OneMoney'].values[0]
            #---------------------2019-----------------------------------------------------#
```

Figure 63: Defining labeller function for Telecash

Defining labeller function for Ecocash

```
[ ]
    def labeller_Ecocash(x):
        if (re.search('2016-01', x) or re.search('2016-02', x)) or (re.search('2016-03', x)):
            return mmnsdata[mmnsdata['Period']=='Q1_2016']['Ecocash'].values[0]
        elif (re.search('2016-04', x) or re.search('2016-05', x)) or (re.search('2016-06', x)):
            return mmnsdata[mmnsdata['Period']=='Q2_2016']['Ecocash'].values[0]
        elif (re.search('2016-07', x) or re.search('2016-08', x)) or (re.search('2016-09', x)):
            return mmnsdata[mmnsdata['Period']=='Q3_2016']['Ecocash'].values[0]
        elif (re.search('2016-10', x) or re.search('2016-11', x)) or (re.search('2016-12', x)):
            return mmnsdata[mmnsdata['Period']=='Q4_2016']['Ecocash'].values[0]
            #-----------------------2017-----------------------------------------------------#
        elif (re.search('2017-01', x) or re.search('2017-02', x)) or (re.search('2017-03', x)):
            return mmnsdata[mmnsdata['Period']=='Q1_2017']['Ecocash'].values[0]
        elif (re.search('2017-04', x) or re.search('2017-05', x)) or (re.search('2017-06', x)):
            return mmnsdata[mmnsdata['Period']=='Q2_2017']['Ecocash'].values[0]
        elif (re.search('2017-07', x) or re.search('2017-08', x)) or (re.search('2017-09', x)):
            return mmnsdata[mmnsdata['Period']=='Q3_2017']['Ecocash'].values[0]
        elif (re.search('2017-10', x) or re.search('2017-11', x)) or (re.search('2017-12', x)):
            return mmnsdata[mmnsdata['Period']=='Q4_2017']['Ecocash'].values[0]
            #----------------------2018-----------------------------------------------------#
        elif (re.search('2018-01', x) or re.search('2018-02', x)) or (re.search('2018-03', x)):
            return mmnsdata[mmnsdata['Period']=='Q1_2018']['Ecocash'].values[0]
        elif (re.search('2018-04', x) or re.search('2018-05', x)) or (re.search('2018-06', x)):
            return mmnsdata[mmnsdata['Period']=='Q2_2018']['Ecocash'].values[0]
        elif (re.search('2018-07', x) or re.search('2018-08', x)) or (re.search('2018-09', x)):
            return mmnsdata[mmnsdata['Period']=='Q3_2018']['Ecocash'].values[0]
        elif (re.search('2018-10', x) or re.search('2018-11', x)) or (re.search('2018-12', x)):
            return mmnsdata[mmnsdata['Period']=='Q4_2018']['Ecocash'].values[0]
            #----------------------2019-----------------------------------------------------#
        elif (re.search('2019-01', x) or re.search('2019-02', x)) or (re.search('2019-03', x)):
            return mmnsdata[mmnsdata['Period']=='Q1_2019']['Ecocash'].values[0]
        elif (re.search('2019-04', x) or re.search('2019-05', x)) or (re.search('2019-06', x)):
```

Figure 64: Defining labeller function for Ecocash

Defining labeller function for Telecash

```
        if (re.search('2016-01', x) or re.search('2016-02', x)) or (re.search('2016-03', x)):
            return mmnsdata[mmnsdata['Period']=='Q1_2016']['Telecash'].values[0]
        elif (re.search('2016-04', x) or re.search('2016-05', x)) or (re.search('2016-06', x)):
            return mmnsdata[mmnsdata['Period']=='Q2_2016']['Telecash'].values[0]
        elif (re.search('2016-07', x) or re.search('2016-08', x)) or (re.search('2016-09', x)):
            return mmnsdata[mmnsdata['Period']=='Q3_2016']['Telecash'].values[0]
        elif (re.search('2016-10', x) or re.search('2016-11', x)) or (re.search('2016-12', x)):
            return mmnsdata[mmnsdata['Period']=='Q4_2016']['Telecash'].values[0]
            #-----------------------2017-----------------------------------------------------#
        elif (re.search('2017-01', x) or re.search('2017-02', x)) or (re.search('2017-03', x)):
            return mmnsdata[mmnsdata['Period']=='Q1_2017']['Telecash'].values[0]
        elif (re.search('2017-04', x) or re.search('2017-05', x)) or (re.search('2017-06', x)):
            return mmnsdata[mmnsdata['Period']=='Q2_2017']['Telecash'].values[0]
        elif (re.search('2017-07', x) or re.search('2017-08', x)) or (re.search('2017-09', x)):
            return mmnsdata[mmnsdata['Period']=='Q3_2017']['Telecash'].values[0]
        elif (re.search('2017-10', x) or re.search('2017-11', x)) or (re.search('2017-12', x)):
            return mmnsdata[mmnsdata['Period']=='Q4_2017']['Telecash'].values[0]
            #----------------------2018-----------------------------------------------------#
        elif (re.search('2018-01', x) or re.search('2018-02', x)) or (re.search('2018-03', x)):
            return mmnsdata[mmnsdata['Period']=='Q1_2018']['Telecash'].values[0]
        elif (re.search('2018-04', x) or re.search('2018-05', x)) or (re.search('2018-06', x)):
            return mmnsdata[mmnsdata['Period']=='Q2_2018']['Telecash'].values[0]
        elif (re.search('2018-07', x) or re.search('2018-08', x)) or (re.search('2018-09', x)):
            return mmnsdata[mmnsdata['Period']=='Q3_2018']['Telecash'].values[0]
        elif (re.search('2018-10', x) or re.search('2018-11', x)) or (re.search('2018-12', x)):
            return mmnsdata[mmnsdata['Period']=='Q4_2018']['Telecash'].values[0]
            #----------------------2019-----------------------------------------------------#
        elif (re.search('2019-01', x) or re.search('2019-02', x)) or (re.search('2019-03', x)):
            return mmnsdata[mmnsdata['Period']=='Q1_2019']['Telecash'].values[0]
        elif (re.search('2019-04', x) or re.search('2019-05', x)) or (re.search('2019-06', x)):
            return mmnsdata[mmnsdata['Period']=='Q2_2019']['Telecash'].values[0]
        elif (re.search('2019-07', x) or re.search('2019-08', x)) or (re.search('2019-09', x)):
            return mmnsdata[mmnsdata['Period']=='Q3_2019']['Telecash'].values[0]
```

Figure 65: Defining labeller function for OneMoney

Now let's apply the defined labeller functions for Ecocash, Telecash and OneMoney to merge the Active Mobile Subscriber data with the twitter data.

```
[ ] data["AMM_Ecocash"] = data['date'].apply(labeller_Ecocash)
    data["AMM_Telecash"] = data['date'].apply(labeller_Telecash)
    data["AMM_OneMoney"] = data['date'].apply(labeller_OneMoney)
```

```
[ ] data.tail(5)
```

	date	likes_count	replies_count	retweets_count	tweet	username	mentions	key_word_cleaner	key_word_cleaner_2	AMM_Ecocash	AMM_Telecash	AMM_OneMoney
407725	2017-01-06	0	0	0	@econet_support guys may you please give me fe...	luvjiz52	['econet_support']	0	0	3189611	52076	10097
407759	2017-01-05	0	1	0	@econet_support you just took money from my ec...	luvjiz52	['econet_support']	0	0	3189611	52076	10097
407776	2017-01-05	0	1	0	@econet_support whats up with these random eco...	nhlallo	['econet_support']	0	0	3189611	52076	10097
407887	2017-01-01	0	0	0	@econet_support why am I charged $1.00 for the...	faraij	['econet_support']	0	0	3189611	52076	10097
407922	2017-01-01	0	1	0	@econet_support if i ecocash the wrong number ...	dorkatcooltable	['econet_support']	0	0	3189611	52076	10097

Figure 66: Applying Active Mobile Subscriber labeller function

Now let's make the date column index of the data for easy grouping and subsetting.

```
[ ] data.set_index('date', inplace=True)
```

```
[ ] data.index = pd.to_datetime(data.index)
```

```
[ ] data.head()
```

date	likes_count	replies_count	retweets_count	tweet	username	mentions	key_word_cleaner	key_word_cleaner_2	AMM_Ecocash	AMM_Telecash	AMM_OneMoney
2021-10-30	1	0	0	@_NobleSavage Car park in town ndinenge ndisin...	niggamumu	[]	0	0	0	0	0
2021-10-30	0	2	0	@EcoCashZW please do explain how do I access m...	faithsharleen	[]	0	0	0	0	0
2021-10-30	0	1	0	@TAWANDAMACHING4 @CabsZimbabwe @EcoCashZW Hell...	ecocash40541216	[]	0	0	0	0	0
2021-10-30	0	3	0	@CabsZimbabwe Good day to your. I sent Zipit t...	tawandamaching4	[{'screen_name': 'ecocashzw', 'name': 'ecocash...	0	0	0	0	0
2021-10-30	3	0	0	@doughbyzo Do you do delivery and do you also ...	mrmuvezwa	[]	0	0	0	0	0

Figure 67: Data indexing

3.3.3.17 Enquiry count trend for Ecocash

```
[ ] ig1, ax1 = plt.subplots()
    ax1.plot(data_ecocash['enquiry_count'].resample("M", convention='end').sum(), marker='.', linewidth=2, label="Ecocash")
    ax1.set_ylabel("Enquiries")
    ax1.set_title("Trend in Enquiry Count (Monthly) for Ecocash")
    ax1.legend()
    plt.grid(which='minor', axis='both')

    fig2, ax2 = plt.subplots()
    ax2.plot(data_ecocash['enquiry_count'].resample("Q", convention='end').sum(), marker='.', linewidth=2, label="Ecocash")
    ax2.set_ylabel("Enquiries")
    ax2.set_title("Trend in Enquiry Count (Quarterly) for Ecocash")


    plt.grid(which='minor', axis='both')
```

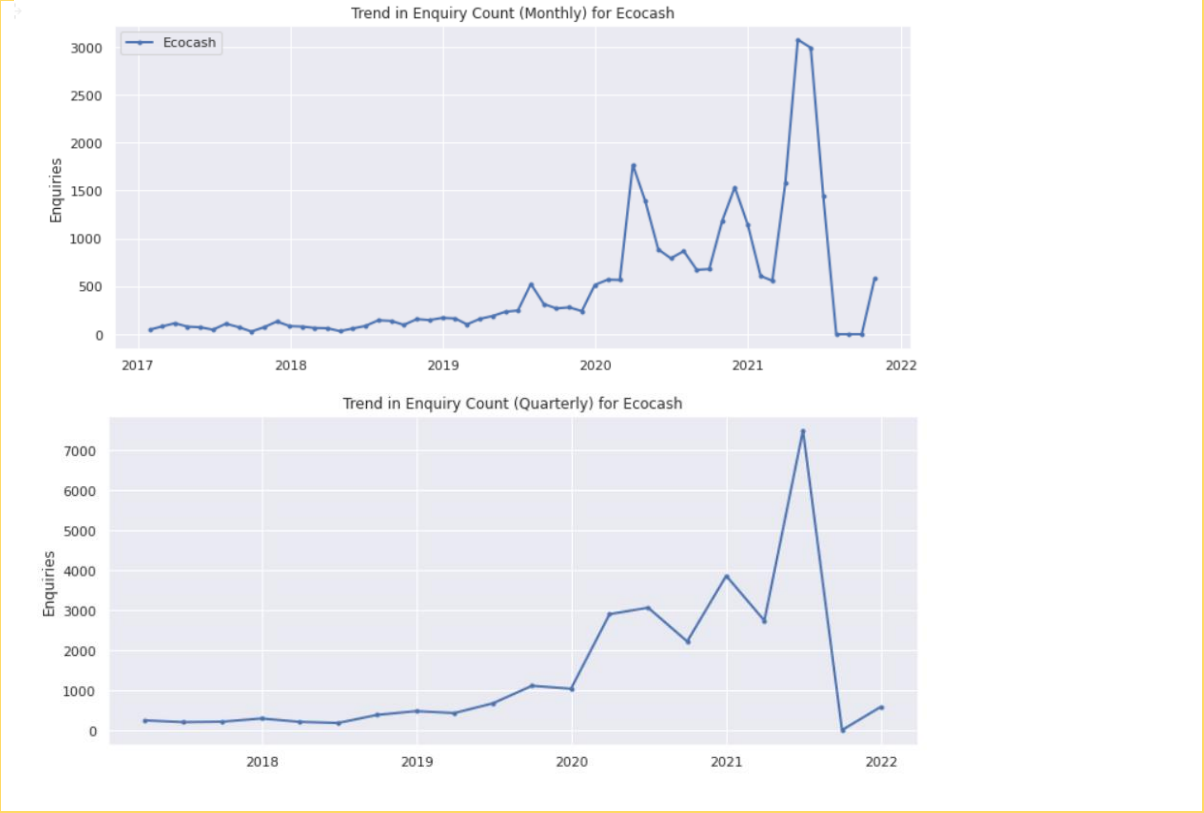


Figure 68: Trend in Enquiry count per month and per quarter

- The trend in both aggregations of by month and by quarter show that the enquiries are mostly average from 2017 to the first quarter of 2019.
- From then on, there's a steady rise to 2020. The numbers take a sharp rise in the first quarter of 2020 then stabilize until the second quarter of 2020.
- From here, the numbers take some ups and downs up to the end of first quarter of 2021, and then make a sudden sharp rise through to the end of second quarter 2021.

3.3.3.18 Complaints Count and Active Mobile Money Subscriptions trends

```
[ ]  fig1, ax1 = plt.subplots()
     ax1.plot(data_ecocash['enquiry_count'].resample("Q", convention='end').sum(), marker='.', linewidth=2, label="Ecocash")
     ax1.set_ylabel("Enquiries")
     ax1.set_title("Trend in Enquiry Count (Quarterly) for Ecocash")
     ax1.legend()
     plt.grid(which='minor', axis='both')

     fig2, ax2 = plt.subplots()
     ax2.plot(mmnsdata['Period'],mmnsdata['Ecocash'], marker='.', linewidth=2, label="Ecocash")
     ax2.set_ylabel("Number of Active Mobile Money Subscribers")
     plt.xticks(rotation=25)
     plt.yticks(rotation=25)
     ax2.set_title("Trend in Active Mobile Money Subscribers (Quarterly) for Ecocash")


     plt.grid(which='minor', axis='both')
```

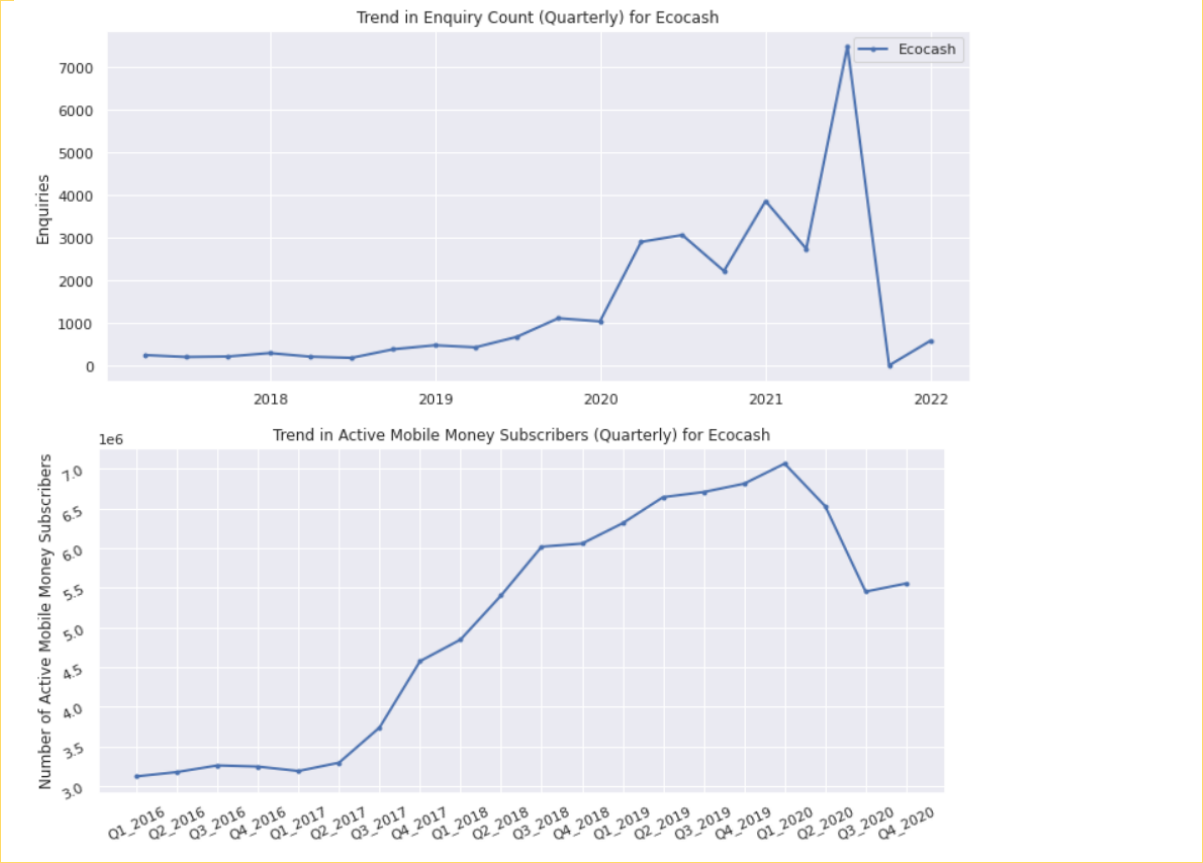


Figure 69: Trend in Enquiry count and Active Mobile Money Subscriptions per quarter

- From the Quarterly graphs above, we can see that there has been a steady rise in the number of complaints and a steep increase in Active Mobile Subscribers for Ecocash from 2016 to the end of 2019.
- However, there was a sharp increase in complaints count from Q1, and Q2 2020 coupled with a deep fall in Active Mobile Money Subscribers for Ecocash within the same period.
- Ecocash started to experience a steady rise in Active Mobile Money Subscribers from Q3 2020

```
[ ]  all_words = ' '.join([text for text in data_ecocash['tidy_tweet']])
     from wordcloud import WordCloud
     mask = load_mask("https://i.imgur.com/UVe6Nas.png")
     wordcloud = WordCloud( background_color="white",mask=mask, width=1500, height=1000, random_state=21, max_font_size=150).generate(all_words)
     plt.figure(figsize=(15, 10))
     plt.imshow(wordcloud, interpolation="bilinear")
     plt.axis('off')
     plt.show()
```

Figure 70: Word cloud for sharp rise in complaints for 2020

The word cloud above has a lot of words in it. Let's list the ones that are most likely to be considered enquiries.

Threats to switch to a competitor (telecel or netone)

- econet
- ecocash
- data
- steward bank
- service
- network

```
[ ]  sns.set_style("whitegrid")

     plt.figure(figsize=(9,5))
     ax = sns.barplot(data=d, x= "Percentage", y = "Word", orient="h")
     ax.set(ylabel = 'Count %')

     for i, v in enumerate(d['Percentage']):
         ax.text(v + 0.3, i + .25, str(v), color='black')

     plt.show()
```

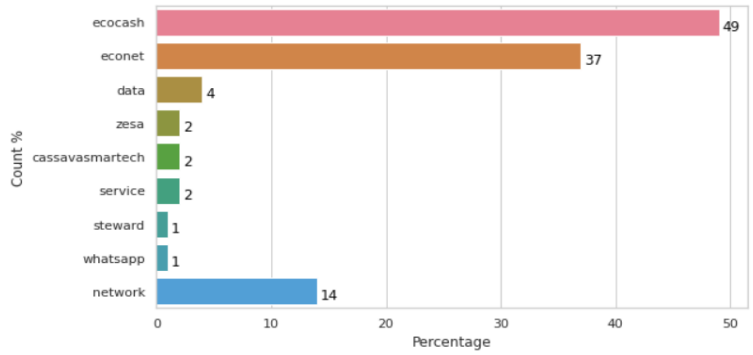


Figure 71: Most enquiry words percentage count for Ecocash in 2020

- Econet and Ecocash are the most prevalent complaints about Ecocash customers on twitter. They make up 49% and 37% respectively for all the complaints investigated. Majority of the complaints seem to point out that Econet data is too expensive compared to its competitors Netone and Telecel. For Ecocash related complaints, the standard-issue is on the system not working correctly.

3.3.3.19 Monthly average number of enquiries

```
     fig3, ax3 = plt.subplots()
     ax3.pie(high_month1['Percentage'], labels=high_month1.index, autopct='%1.1f%%', startangle=90)
     #draw circle
     centre_circle = plt.Circle((0,0),0.70,fc='white')
     fig = plt.gcf()
     fig.gca().add_artist(centre_circle)
     # Equal aspect ratio ensures that pie is drawn as a circle
     ax3.axis('equal')
     ax3.set_title("Monthly Average Number of Enquiries", fontweight="bold", fontsize=14)

     plt.tight_layout()
     plt.show()
```

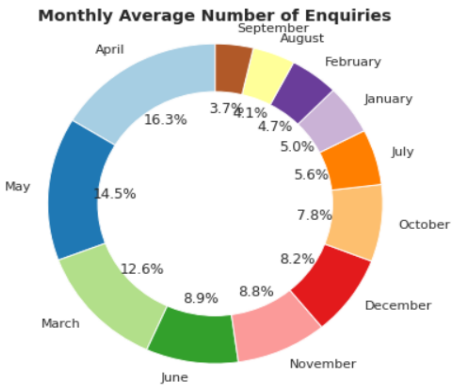
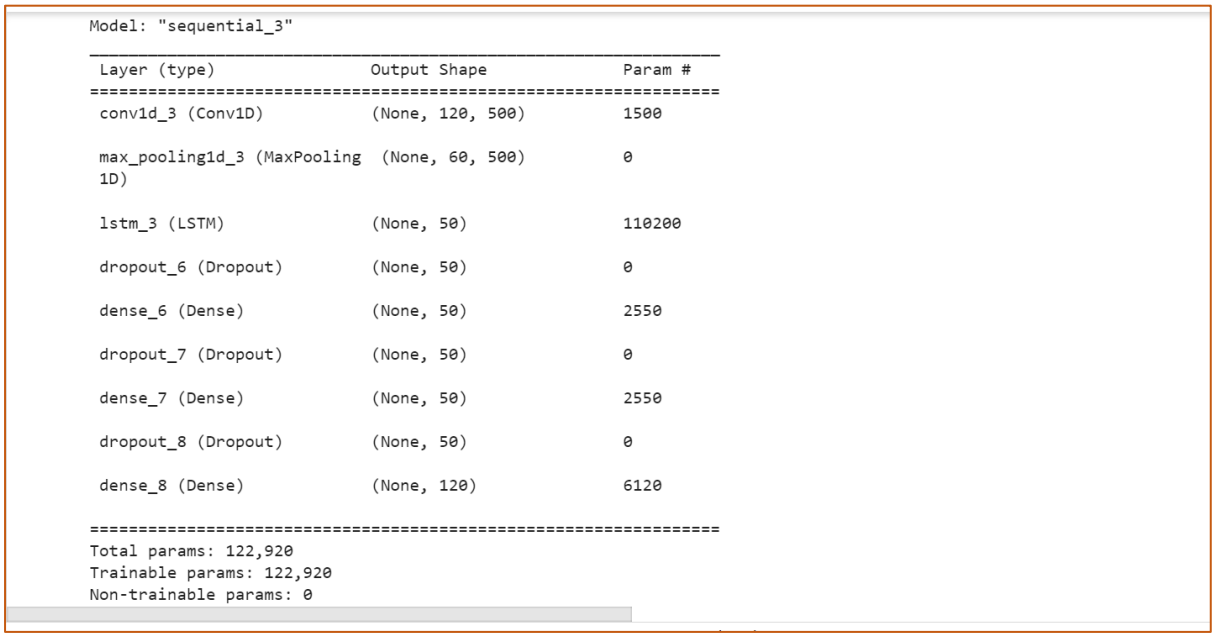


Figure 72: Most enquiry words percentage count for Ecocash in 2020

- From the above donut graph, it is clear that March, April and May seem to be the months when Ecocash experiences most of its complaints on twitter.

3.3.4 Modelling

In the modelling phase, a hybrid model was used to handle various tasks in the machine learning pipeline. This hybrid model combined the CNN (Convolutional Neural Networks) and LSTM (Long Short-Term Memory) machine learning algorithms to form a hybrid model known as LSTM_CNN. This model has been used by several researchers achieving the best performance results and accuracy.



```
Model: "sequential_3"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 conv1d_3 (Conv1D)           (None, 120, 500)          1500

 max_pooling1d_3 (MaxPooling  (None, 60, 500)          0
 1D)

 lstm_3 (LSTM)               (None, 50)                110200

 dropout_6 (Dropout)         (None, 50)                0

 dense_6 (Dense)             (None, 50)                2550

 dropout_7 (Dropout)         (None, 50)                0

 dense_7 (Dense)             (None, 50)                2550

 dropout_8 (Dropout)         (None, 50)                0

 dense_8 (Dense)             (None, 120)               6120

=================================================================
Total params: 122,920
Trainable params: 122,920
Non-trainable params: 0
```

Figure 72: Model structure

As shown above, the structure of the CNN-LSTM hybrid model developed has CNN and LSTM, an input layer, 1 dimensional convolution layer (1D convolutional), pooling layer with a pooling size of 2, dropouts layers with a rate or probability of 0.7 each to prevent the neural network from overfitting, an LSTM (hidden) hidden layer, 3 dense layers of 50 neurons each and layer full connection with 120 neurons. The model used ReLu (rectified linear activation) function as it is the mostly used and default function when developing multilayer Perceptron and convolutional neural networks and it also overcomes the vanishing gradient problem, allowing models to learn faster and perform better.

The model developed had the parameters shown in Figure 3.23 below.

```
model_LSTM_CNN = Sequential()

model_LSTM_CNN.add(Conv1D(filters=500, kernel_size=2, input_shape=(look_back,1), padding='same', activation='relu'))
model_LSTM_CNN.add(MaxPooling1D(pool_size=2))
model_LSTM_CNN.add(LSTM(50))
model_LSTM_CNN.add(Dropout(0.7))
model_LSTM_CNN.add(Dense(50))
model_LSTM_CNN.add(Dropout(0.7))
model_LSTM_CNN.add(Dense(50))
model_LSTM_CNN.add(Dropout(0.7))
model_LSTM_CNN.add(Dense(look_forward))

monitor = EarlyStopping(monitor='val_loss', min_delta=1e-3, patience=2, verbose=1, mode='auto')
#checkpointer
checkpointer = ModelCheckpoint(filepath="best_weights_lstm.hdf5", verbose=0, save_best_only=True) # save best model


model_LSTM_CNN.compile(loss='mean_squared_error', optimizer='adam', metrics=['accuracy'] )
#----------------------------------

print(model_LSTM_CNN.summary())
```

Figure 73: Model parameters

The model was developed using artificial neural networks to come up with a sequential model using Tensorflow and Keras.

```
Model: "sequential_3"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 conv1d_3 (Conv1D)           (None, 120, 500)          1500

 max_pooling1d_3 (MaxPooling  (None, 60, 500)          0
 1D)

 lstm_3 (LSTM)               (None, 50)                110200

 dropout_6 (Dropout)         (None, 50)                0

 dense_6 (Dense)             (None, 50)                2550

 dropout_7 (Dropout)         (None, 50)                0

 dense_7 (Dense)             (None, 50)                2550

 dropout_8 (Dropout)         (None, 50)                0

 dense_8 (Dense)             (None, 120)               6120

=================================================================
Total params: 122,920
Trainable params: 122,920
Non-trainable params: 0
```

Figure 74: Model summary

The model summary shows that the model developed has 3 dense layers of 50 neurons each, 3 dropout layers with a dropout rate of 0.7 and an outer layer with a single prediction value. The model also has 22, 322 total parameters and all are trainable. This entails that the model can be fitted on all the parameters for training.

3.3.4.2 Hyper Parameter Tuning

Machine learning algorithms allow parameters to be changed to improve the model efficiency. Researchers and data science practitioners should change the parameters until they are satisfied with the model results. These performance tuning switches are known as the hyper parameters, and they help us a great deal in controlling the behavior of machine learning algorithms when we try to optimize them for better performance. One needs to have the artistic skills to be able to tune the model for the best performance optimization.

```
[ ]  embed_dim = 140
     lstm_out = 196

     model = Sequential()
     model.add(Embedding(max_features, embed_dim,input_length = X.shape[1]))
     model.add(SpatialDropout1D(0.4))
     model.add(LSTM(lstm_out, dropout=0.2, recurrent_dropout=0.2))
     model.add(Dense(3,activation='softmax'))
     model.compile(loss = 'categorical_crossentropy', optimizer='adam',metrics = ['accuracy'])
     print(model.summary())
```

```
Model: "sequential"
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
embedding (Embedding)        (None, 62, 140)           1400000
_________________________________________________________________
spatial_dropout1d (SpatialDr (None, 62, 140)           0
_________________________________________________________________
lstm (LSTM)                  (None, 196)               264208
_________________________________________________________________
dense (Dense)                (None, 3)                 591
=================================================================
Total params: 1,664,799
Trainable params: 1,664,799
Non-trainable params: 0
_________________________________________________________________
None
```

Figure 75: Hyper Parameter Tuning

3.3.5 Evaluation

In the evaluation phase, each of the models used in carrying out different functions in the sentiment analysis project will be evaluated to check if they can achieve the business objectives. Furthermore, the evaluation phase includes testing the model on test applications in a real-world setting. The evaluation was done using performance metrics which are accuracy, F1 score, precision, and recall score as well as the confusion matrix. Once the model is evaluated with satisfactory performance, the model is then deployed.

The outcome of the evaluation is shown in Figure 76 below:

```
model.evaluate(X_test,Y_test)
15/15 [==============================] - 0s 3ms/step - loss: 34762616832.0000 - mean_squared_error: 34762616832.0000
[34762616832.0, 34762616832.0]
```

Figure 76: Model Evaluation

The model evaluation outcome shows that model predicted with a minimum loss and low mean squared error of and respectively. The values of the metrics confirm the model effectiveness in the prediction of the targeted feature.

Model evaluation was further done on unseen data to simulate the real world by predicting the unseen data. The model was tested on the prediction of the targeted feature which is the market share or active mobile money subscribers in relation to customer sentiment changes. The outcome of the prediction is depicted in Figure 77 below:

```
[ ] predictions = predictions.flatten()
    pred_df=pd.DataFrame({'Date': pd.date_range('2021-01-01', periods=120, freq='1D'),
                          'AMM_Ecocash': predictions})
    pred_df['Date'] = pred_df['Date'].dt.date
    pred_df
```

	Date	AMM_Ecocash
0	2021-01-01	462374
1	2021-01-02	418927
2	2021-01-03	518400
3	2021-01-04	337373
4	2021-01-05	345160
...	...	...
115	2021-04-26	406056
116	2021-04-27	559658
117	2021-04-28	416996
118	2021-04-29	480030
119	2021-04-30	397118

120 rows × 2 columns

3.3.6 Deployment

In the deployment phase, the evaluation results are used to determine a deployment strategy, develop a suitable model or models, and document the deployment procedure. Based on the requirement, the deployment phase can be a simple generated report or a repeated process. In this study, the developed model is deployed on various servlet containers. The case for this model needs to be used by other researchers. In figure 77 below the model is being saved for future use.

```
[ ]  model_save_name = 'lstm_cnn.h5'
     path = F"/content/gdrive/My Drive/{model_save_name}"
     model.save(model.state_dict(), path)
```

Figure 77: Saving Model

3.4 Target population

The target population will be all social media data that talk about Ecocash and can be mined to get customer sentiment. This includes but is not limited to all data from social media platforms like WhatsApp, Facebook, Instagram, Twitter, and other platforms that are used by people in their social interactions.

3.5 Sample size

The research focuses only on Ecocash twitter data mined for the period 2017 up to July 2021, where different customers expressed their sentiments and concern on Ecocash products and services offering. The research also focused on Active Mobile Money Subscribers data obtained from POTRAZ Quarterly reports from the official Potraz website for the period 2017 to 2020.

3.6 Reliability and Validity

All data for the study will be mined straight from Twitter servers without any modification by any third-party sources, making it trustworthy. Several machine learning approaches proven to show valid results have been used on sentiment analysis for Twitter which will render the findings of the study reliable.

3.7 Ethical Consideration

This study will adhere to many ethical guidelines to protect social and moral values and ensure that the research does not hurt others. These guidelines on norms and moral standards can allow a wide range of ethical positions (Saunders; et al 2011). To safeguard customer information privacy, the researcher will not individualize twitter sentiments to point to a single person but will instead analyse customer sentiment collectively. In addition to English, the study considers the sentiment of diverse languages such as Shona and Ndebele when analysing client sentiment. Above all, the researcher will credit the work of other researchers by referring to it according to the APA reference style, which is offered by Chinhoyi University of Technology.

3.8 Chapter Summary

The research methodology was outlined in Chapter Three, and it addressed the procedures that were followed in the data collecting, analysis, and generation of insights that were valuable in addressing the research questions as led by the study's objectives. The CRISP-DM Framework was addressed in this chapter, which is a data analysis model characterized by a six-step technique for understanding the business context, the data used, data preparation, modelling, model evaluation, and ultimately model deployment. The ethics that the study considered were also explored in this chapter.

CHAPTER 4: Results and Discussion

4.0 Introduction

This chapter aims to practically test the theory behind sentiment analysis and machine learning modelling using the LSTM_CNN hybrid algorithm approach. This Chapter gives the presentation and analysis of the results of the study with respect to the computations done in the previous Chapter. A sentiment analysis and time series forecasting model was designed and implemented using deep learning to perform sentiment analysis and predict market share for Ecocash mobile money operator in Zimbabwe. The results presented follows the methodology adopted in terms of the machine learning pipeline used to implement the CRISP model used in the study. A discussion of the results is presented at the end of the study to give insights on the findings of the study.

4.1 Research Objectives and Results

Research Objective 1: *To construct a time series analysis of how the frequencies of customer complaints have changed over the last 5 years.*

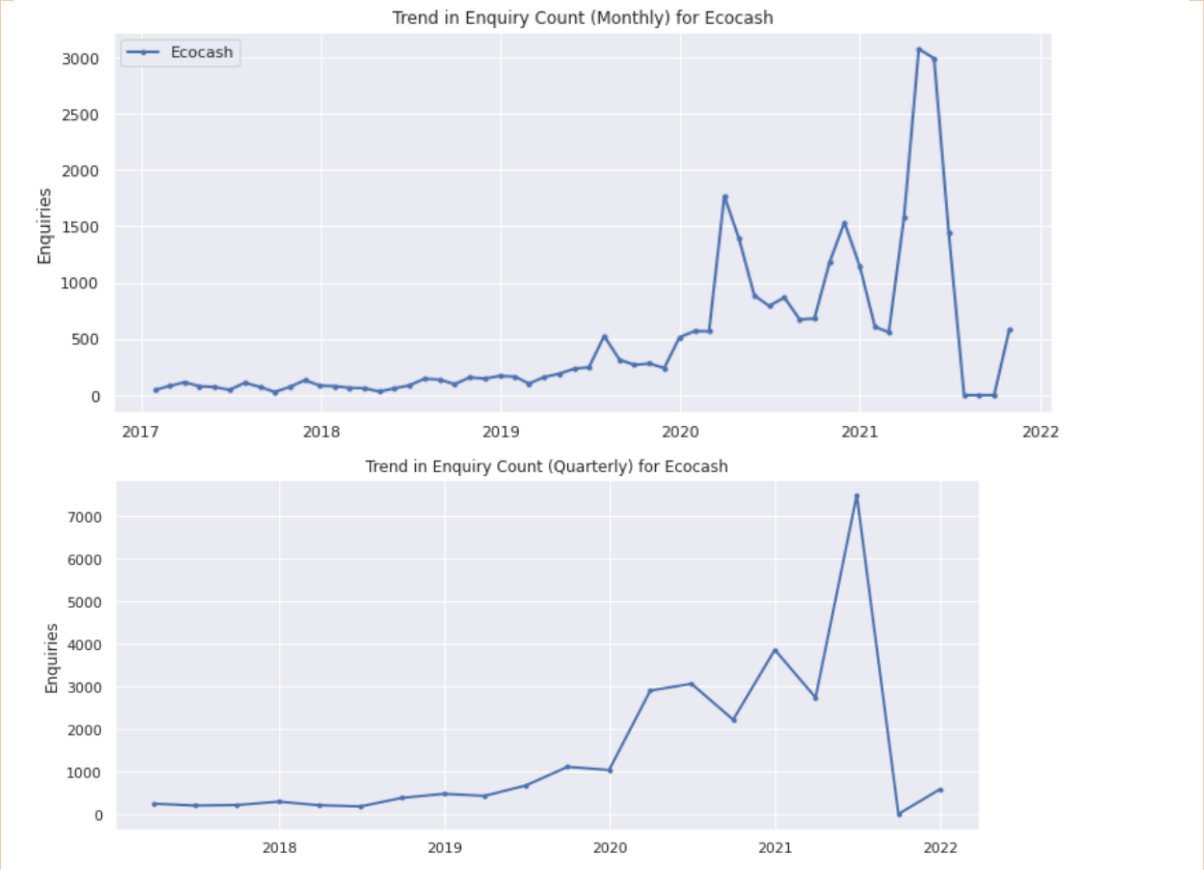


Figure 78: Customer Complaints trends

The trend in both aggregations of by month and by quarter show that the enquiries are mostly average from 2017 to the first quarter of 2019. From then on, there's a steady rise to 2020. The numbers take a sharp rise in the first quarter of 2020 then stabilize until the second quarter of 2020. From here, the numbers take some ups and downs up to the end of first quarter of 2021, and then makes a sudden sharp rise through to end of second quarter 2021.

Research Objective 2: *To present the most prevalent customer complaints categories.*

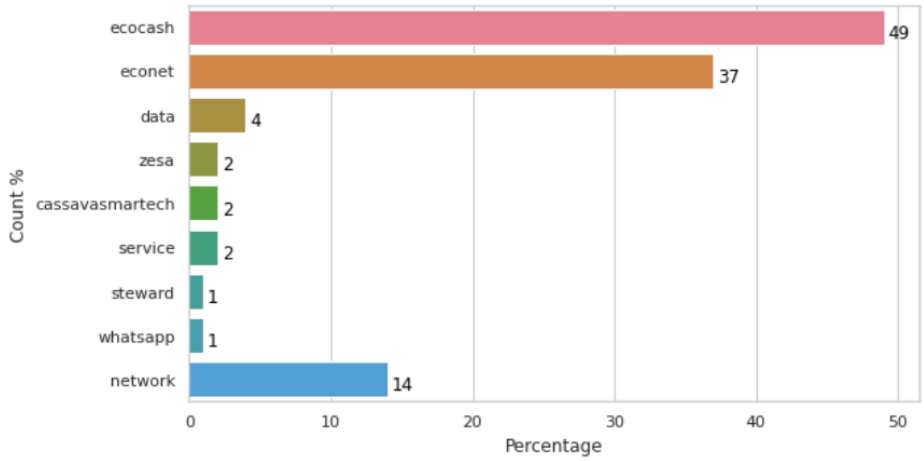


Figure 79: Customer Complaints Categories

Econet and Ecocash are the most prevalent complaint about Ecocash customers on twitter. They make up 49% and 37% respectively for all the complaint investigated. Majority of the complaints seems to point out that Econet data is too expensive compared to its competitors Netone and Telecel. For Ecocash related complains, the standard-issue is on the system not working correctly.

Research Objective 3: *To evaluate the impact of social media sentiments to the actual business market share.*

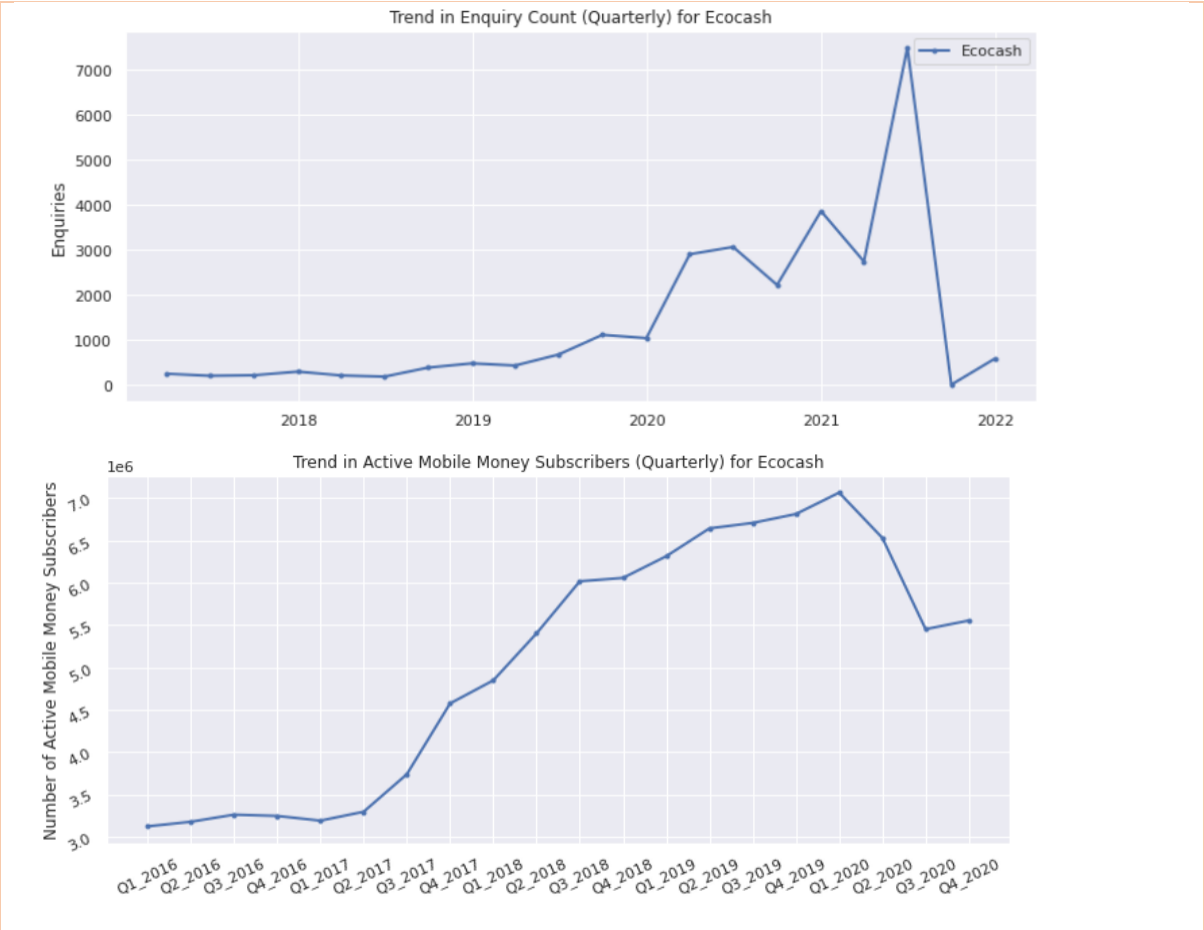


Figure 90: Relationship between customer complaints and Active Mobile Money Subscribers trend

From the Quarterly trends graphs, it can be clearly deduced that there has been a steady rise in the number of complaints with a steep increase in Active Mobile Subscribers for Ecocash from 2016 to end of 2019. However, there was sharp increase in complaints count from Q1, and Q2 2020 coupled with a deep fall in Active Mobile Money Subscribers for Ecocash within the same period. Ecocash started to experience a steady rise in Active Mobile Money Subscribers from Q3 2020. This clearly shows that there is a relationship between customer complaints changes and active mobile money subscription changes for Ecocash, though there are some other factors that strongly influence trends in Active Mobile Money subscriptions. These factors include change to Mobile Money regulations or policies by the regulatory authority (RBZ). For example, the one subscriber one mobile money wallet per mobile money operator which led to closure of several accounts by the mobile money operators.

Research Objective 4: *To forecast future market share changes in relation to customer complaint changes.*

The model was tested on the prediction of the targeted feature which is the market share or active mobile money subscribers. The outcome of the prediction is depicted in Figure 91 below:

```
[ ] predictions = predictions.flatten()
    pred_df=pd.DataFrame({'Date': pd.date_range('2021-01-01', periods=120, freq='1D'),
                          'AMM_Ecocash': predictions})
    pred_df['Date'] = pred_df['Date'].dt.date
    pred_df
```

	Date	AMM_Ecocash
0	2021-01-01	462374
1	2021-01-02	418927
2	2021-01-03	518400
3	2021-01-04	337373
4	2021-01-05	345160
...	...	...
115	2021-04-26	406056
116	2021-04-27	559658
117	2021-04-28	416996
118	2021-04-29	480030
119	2021-04-30	397118

120 rows × 2 columns

Figure 91: Model Predictions

From the research, we can see that the predicted trends in active mobile money subscriptions for Ecocash shows a steady rise in the number of active mobile money subscribers thus resulting in increased market share for the operator.

4.2 Model Training History

Model training history attribute is a dictionary recording training loss values and metrics values at successive epochs as well as the validation loss values and validation metrics where applicable. The model training history retained loss and validation loss (val_loss). The history of the model which was trained using 1000 epochs is indicated in Figure 4.1

```
Epoch 10/20
250/250 [==============================] - 52s 208ms/step - loss: 3.3112 - val_loss: 0.0642
Epoch 11/20
250/250 [==============================] - 52s 209ms/step - loss: 3.1151 - val_loss: 0.0476
Epoch 12/20
250/250 [==============================] - 52s 208ms/step - loss: 3.1075 - val_loss: 0.0526
Epoch 13/20
250/250 [==============================] - 52s 208ms/step - loss: 2.9499 - val_loss: 0.0434
Epoch 14/20
250/250 [==============================] - 52s 207ms/step - loss: 2.8804 - val_loss: 0.0549
Epoch 15/20
250/250 [==============================] - 52s 208ms/step - loss: 2.8852 - val_loss: 0.0569
Epoch 16/20
250/250 [==============================] - 52s 208ms/step - loss: 2.8290 - val_loss: 0.1062
Epoch 17/20
250/250 [==============================] - 52s 208ms/step - loss: 2.8002 - val_loss: 0.0489
Epoch 18/20
250/250 [==============================] - 52s 208ms/step - loss: 2.7160 - val_loss: 0.0419
Epoch 19/20
250/250 [==============================] - 52s 208ms/step - loss: 2.7162 - val_loss: 0.0611
Epoch 20/20
250/250 [==============================] - 52s 207ms/step - loss: 2.6427 - val_loss: 0.0615
```

Figure 92: Model Training History

The history of the model retained the loss, val_loss and val_accuracy. The state of the history logged shows that the model loss continued to drop with each epoch completed as well as the val_loss while the accuracy and val_accuracy converged towards 1. The results suggest that the model trained effectively and efficiently as desired. The accuracy of the model is as high as 0.897 which is high enough to produce best result in the prediction task.

4.3 Model Loss History

The model loss refers to a number indicating how bad the model's prediction was on a single example such that, if the model's prediction is perfect, the loss is zero, otherwise, the loss is greater. The loss of the model is as presented in Figure 79.

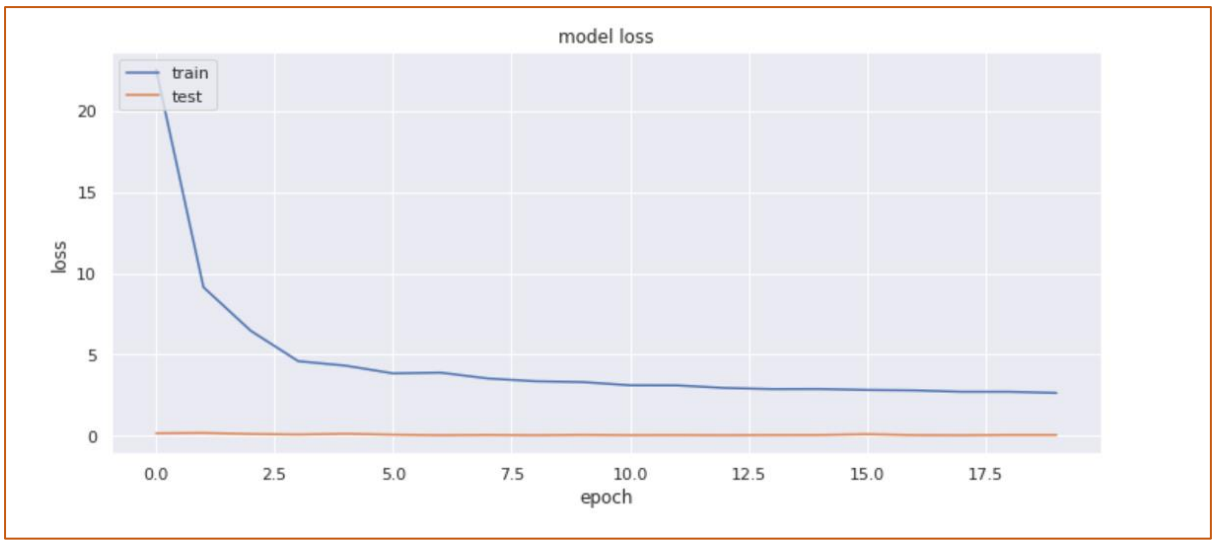


Figure 93: Model Loss History

The model loss presented a behavior which is in conformity with the ideal trend of loss in model training. The loss continued to converge towards 0 as the training process progressed. The loss dropped significantly from 26.0 at the initial training stage to 2.1 at epoch 10 which continued to slightly decrease until the last epoch at which the least loss on both training and validation were realized. The model loss therefore suggests that the model trained effectively and managed to understand the patterns in the data with minimum loss which is acceptable enough to validate the model as effective and efficient.

4.5 Chapter Summary

This chapter focused on the presentation of results obtained from the research in relation to the research objectives evaluations of the developed machine learning model using the evaluation matrices. The results presentation was on the model evaluation score as the model outcome. The chapter also included the prediction of the active mobile money subscriptions trends (market share), a presentation and analysis of customer sentiments and market share past trends The next Chapter provides a summary of findings, conclusions, and implications of the study.

CHAPTER 5: Summary, Conclusions, and Implications

5.1 Summary of Findings

The main aim of the study was to develop a machine learning solution to perform customer sentiment analysis and predict market share trends using twitter dataset for Ecocash using an LSTM_CNN hybrid model approach. The chapter seeks to answer the research questions raised in the first chapter and is guided by the literature review and theoretical framework.

5.2 Conclusions

The study pointed out that ML can be used to come up with a solution in solving complex problems found in twitter sentiment analysis. The researcher has managed to come up with a model that performs sentiment analysis and market share trends forecasting, which is the primary objective. Different small pieces of code found in python programming language proved to be very helpful in sentiment classification and market share prediction using twitter dataset for Ecocash. Hence, we can conclude that:

1. The time series analysis clearly shows the trend in both aggregations by month and by quarter and reveals that the enquiries for Ecocash are mostly average from first quarter of 2017 to the first quarter of 2019. From then on, there's a steady rise to 2020. The numbers take a sharp rise in the first quarter of 2020 then stabilize until the second quarter of 2020 and then, the numbers take some ups and downs up to the end of first quarter of 2021, and then make a sudden sharp rise through to the end of second quarter 2021. To get more insights into its existing customers, Ecocash needs to introspect as to what caused these dramatic changes in the last two years.
2. From the Quarterly trends graphs, it can be clearly deduced that there has been a steady rise in the number of complaints with a steep increase in Active Mobile Subscribers for Ecocash from 2016 to end of 2019. However, there was a sharp increase in complaints count from Q1, and Q2 2020 coupled with a deep fall in Active Mobile Money Subscribers for Ecocash within the same period. Ecocash started to experience a steady rise in Active Mobile Money Subscribers from Q3 2020. This clearly shows that there is a relationship between change in customer complaints with changes in active mobile money subscribers for Ecocash, though there are some other factors that strongly influence trends in Active Mobile Money subscriptions. These factors include changes

to Mobile Money regulations or policies by the regulatory authority (RBZ). For example, *"one subscriber one mobile money wallet"* per mobile money operator which led to closure of several accounts by the mobile money operators.

3. From this study, the researcher can conclude that March, April and May seem to be the months when Ecocash experiences most of its complaints on twitter.
4. Econet and Ecocash are the most prevalent complaints about Ecocash customers on twitter. They make up 49% and 37% respectively for all the complaints investigated. Majority of the complaints seem to point out that Econet data is too expensive compared to its competitors Netone and Telecel. For Ecocash related complaints, the standard-issue is on the system not working correctly.
5. From the research, we can see that several tweets were in Shona and Ndebele hence there is a need to research better ways of improving sentiment detection for vernacular languages (Shona and Ndebele).
6. From the research, we can see that the predicted trends in active mobile money subscriptions for Ecocash shows a steady rise in the number of active mobile money subscribers thus resulting in increased market share for the operator.

5.2 Recommendations

The research has uncovered new areas of research in Zimbabwe. The following recommendations that can improve research in this area:

1. The dataset that was used contained several tweets in the local language (Shona and Ndebele). There is a need for further research in finding the best way for sentiment detection and classification of local languages.
2. For ML models to work well, much data is required; hence more data is required for the models to be able to produce meaningful results.
3. Some collection methods are not collecting much so it will be prudent to come to find other methods of catching mosquitoes.

The researcher recommends Ecocash to come up with some refactoring of the system to minimize the inefficiencies that lead to customer inconveniences. It would be a good idea for the tech team to start focusing on developing an automated reconciliation system that processes and validates wrongful deductions of accounts such that customers might be

refunded without much hassle. This will result in high customer retention because of high customer satisfaction through a reliable service delivery

Finally, the study reveals that customer sentiment against Ecocash has become increasingly negative year-on-year, exponentially, especially over the past two years. The Ecocash brand is fast being associated with extortion, inefficiency and out of touch with its customer base; hence the company is at the edge of losing its customers to competition as witnessed by the fall in active mobile money subscriptions trends. As such the company needs to take proactive measures to restore its good standing in the market.

REFERENCES

Ahmad, A. K., Jafar, A., & Aljoumaa, K. (2019). Customer churn prediction in telecom using machine learning in big data platform. *Journal of Big Data*, *6*(1). https://doi.org/10.1186/s40537-019-0191-6

Amin, A., Al-Obeidat, F., Shah, B., Adnan, A., Loo, J., & Anwar, S. (2019). Customer Churn Prediction in Telecommunication Industry under Uncertain Situation. *Journal of Business Research*, *94*, 290–301. http://repository.uwl.ac.uk/id/eprint/4800/1/Customer churn prediction.pdf

Amin, A., Shah, B., Khattak, A. M., Lopes Moreira, F. J., Ali, G., Rocha, A., & Anwar, S. (2019). Cross-company customer churn prediction in telecommunication: A comparison of data transformation methods. *International Journal of Information Management*, *46*, 304–319. https://doi.org/10.1016/j.ijinfomgt.2018.08.015

Adnan, A., Feras, A.-O., Babar, S., Awais, A., Jonathan, L., & Sajid, A. (2019). Customer churn prediction in telecommunication industry using data certainty. *Journal of Business Research*, 290-301.

Ali, M. N., El-Hamid, M. M., & Youssif, A. (2019). Sentiment analysis for movies reviews dataset using deep learning models. *Int Journal of Data Mining & Knowledge Management Process*, 19–27.

Augenstein, I., Rocktaschel, T., Vlachos, A., & Bontcheva, K. (2016). Stance detection with bidirectional conditional encoding. *Proceedings of EMNLP*, (pp. 876-885).

Bahdanau, D., Cho, K., & Bengio, Y. (2014). Neural machine translation by jointly learning to align and translate. *Neural machine translation by jointly learning to align and translate*.

Bhardwaj, D., Narayan, Y., Vanraj, Kumar, P., & Dutta, M. (2015). Sentiment Analysis for Indian Stock Market Prediction Using Sensex and Nifty. *Sentiment Analysis for Indian Stock Market Prediction Using Sensex and Nifty*, 85-91.

Chen, D., & Manning, C. D. (2014). A fast and accurate dependency parser using neural networks. *A fast and accurate dependency parser using neural networks*, 740-750.

Collobert, R., Weston, J., Leon, B., Michael, K., Koray, K., & Pavel, K. (2011). Natural Language Processing (Almost) from Scratch. *Journal of Machine Learning Research 12*, 25-33.

Creswell, J. W. (2014). Research Design: Qualitative, Quantitative and Mixed Methods Approaches (4th ed.). *Research Design: Qualitative, Quantitative and Mixed Methods Approaches (4th ed.)*.

Del Pra, M. (2020, November 02). *Time Series Forecasting with Deep Learning and Attention Mechanism*. Retrieved from Towards Data Science Web Site: https://towardsdatascience.com/time-series-forecasting-with-deep-learning-and-attention-mechanism-2d001fc871fc

Caigny, A. De, Coussement, K., Bock, K. W. De, & Lessmann, S. (2019). *Version of Record:*

https://www.sciencedirect.com/science/article/pii/S0169207019301499. 0–36.

Chen, J., Hu, Y., Liu, J., Xiao, Y., & Jiang, H. (2019). Deep short text classification with knowledge powered attention. *33rd AAAI Conference on Artificial Intelligence, AAAI 2019, 31st Innovative Applications of Artificial Intelligence Conference, IAAI 2019 and the 9th AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2019*, 6252–6259. https://doi.org/10.1609/aaai.v33i01.33016252

D'Andrea, A., Ferri, F., Grifoni, P., & Guzzo, T. (2015). Approaches, Tools and Applications for Sentiment Analysis Implementation. *International Journal of Computer Applications*, *125*(3), 26–33. https://doi.org/10.5120/ijca2015905866

Dang, C. N., Moreno-Garcia, M. N., & Prieta, F. D. la. (2020). *Sentiment Analysis Based on Deep Learning :*

Ehsan, M., Nemati, S., Abdar, M., & Cambria, E. (2021). ABCDM : An Attention-based Bidirectional CNN-RNN Deep Model for sentiment analysis. *Future Generation Computer Systems*, *115*, 279–294. https://doi.org/10.1016/j.future.2020.08.005

Feldman, R. (2013). Techniques and applications for sentiment analysis. *Communications of the ACM*, 82-89.

J. C. B. Gamboa, Deep learning for time-series analysis. arXiv preprint arXiv:1701.01887(2017)

Gargiulo, F., Silvestri, S., Ciampi, M., & De Pietro, G. (2019). Deep neural network for hierarchical extreme multi-label text classification. *Applied Soft Computing Journal*, *79*, 125–138. https://doi.org/10.1016/j.asoc.2019.03.041

Gohil, S., Vuik, S., & Darzi, A. (2018). Sentiment analysis of health care tweets: Review of the methods used. *JMIR Public Health and Surveillance*, *4*(4). https://doi.org/10.2196/publichealth.5789

Hassan, Y. A., & Medhat, W. M. (2014). Sentiment Analysis Algorithms and Applications: A Survey. *Ain Shams Engineering Journal*, 5.

Ho-Young. (2010, October 11). *Mobile Money Canada.* Retrieved from Mobile Money Canada: http://www.mobilemoneycanada.com/node/4197/How%20significant%20are%20

Kajiva, E. M. (2017). A Tool to Predict the Possibility of Social Unrest Using Sentiments Analysis - Case of Zimbabwe Politics 2017 - 2018. *International Journal of Science and Research*, 1541–1545.

Kim, Y. (2014). Convolutional Neural Networks for Sentence Classification. *2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, (pp. 1746-1751).

Lai, H., Pan, Y., Liu, Y., & Yan, S. (2015). Simultaneous feature learning and hash coding with deep neural networks. *Proceedings of the IEEE conference on computer vision and pattern recognition*, (pp. 3270-3278).

Laudon, K. C., & Laudon, J. P. (2012). *Management Information Systems Managing in the Digital Firm (12th ed.).* London: Person education.

Leishman, P. (2011). Is there Really any Money in Mobile Money? *Mobile Money for the Unbanked*, 2-5.

Lemmens, A., & Croux, C. (2006). Bagging And Boosting Classification Trees To Predict Churn. *Journal of Marketing Research*, 276-286.

Li, C., Zhan, G., & Li, Z. (2018). News Text Classification Based on Improved Bi-LSTM-CNN. *Proceedings - 9th International Conference on Information Technology in Medicine and Education, ITME 2018*, 890–893. https://doi.org/10.1109/ITME.2018.00199

Li, X., Peng, L., Yao, X., Cui, S., Hu, Y., You, C., & Chi, T. (2017). Long short-term memory neural network for air pollutant concentration predictions: Method development and evaluation. *Environmental Pollution*, *231*, 997–1004. https://doi.org/10.1016/j.envpol.2017.08.114

Liu, F., Zhou, X., Wang, T., Cao, J., Wang, Z., Wang, H., & Zhang, Y. (2019). An Attention-based Hybrid LSTM-CNN Model for Arrhythmias Classification. *Proceedings of the International Joint Conference on Neural Networks*, *2019-July*(October 2020), 1–8. https://doi.org/10.1109/IJCNN.2019.8852037

Adnan, A., Feras, A.-O., Babar, S., Awais, A., Jonathan, L., & Sajid, A. (2019). Customer churn prediction in telecommunication industry using data certainty. *Journal of Business Research*, 290-301.

Ali, M. N., El-Hamid, M. M., & Youssif, A. (2019). Sentiment analysis for movies reviews dataset using deep learning models. *Int Journal of Data Mining & Knowledge Management Process*, 19–27.

Augenstein, I., Rocktaschel, T., Vlachos, A., & Bontcheva, K. (2016). Stance detection with bidirectional conditional encoding. *Proceedings of EMNLP*, (pp. 876-885).

Bahdanau, D., Cho, K., & Bengio, Y. (2014). Neural machine translation by jointly learning to align and translate. *Neural machine translation by jointly learning to align and translate*.

Bhardwaj, D., Narayan, Y., Vanraj, Kumar, P., & Dutta, M. (2015). Sentiment Analysis for Indian Stock Market Prediction Using Sensex and Nifty. *Sentiment Analysis for Indian Stock Market Prediction Using Sensex and Nifty*, 85-91.

Chen, D., & Manning, C. D. (2014). A fast and accurate dependency parser using neural networks. *A fast and accurate dependency parser using neural networks*, 740-750.

Collobert, R., Weston, J., Leon, B., Michael, K., Koray, K., & Pavel, K. (2011). Natural Language Processing (Almost) from Scratch. *Journal of Machine Learning Research 12*, 25-33.

Creswell, J. W. (2014). Research Design: Qualitative, Quantitative and Mixed Methods Approaches (4th ed.). *Research Design: Qualitative, Quantitative and Mixed Methods Approaches (4th ed.)*.

Del Pra, M. (2020, November 02). *Time Series Forecasting with Deep Learning and Attention Mechanism*. Retrieved from Towards Data Science Web Site:

https://towardsdatascience.com/time-series-forecasting-with-deep-learning-and-attention-mechanism-2d001fc871fc

Feldman, R. (2013). Techniques and applications for sentiment analysis. *Communications of the ACM*, 82-89.

Hassan, Y. A., & Medhat, W. M. (2014). Sentiment Analysis Algorithms and Applications: A Survey. *Ain Shams Engineering Journal*, 5.

Ho-Young. (2010, October 11). *Mobile Money Canada.* Retrieved from Mobile Money Canada: http://www.mobilemoneycanada.com/node/4197/How%20significant%20are%20

Kajiva, E. M. (2017). A Tool to Predict the Possibility of Social Unrest Using Sentiments Analysis - Case of Zimbabwe Politics 2017 - 2018. *International Journal of Science and Research*, 1541–1545.

Kim, Y. (2014). Convolutional Neural Networks for Sentence Classification. *2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, (pp. 1746-1751).

Lai, H., Pan, Y., Liu, Y., & Yan, S. (2015). Simultaneous feature learning and hash coding with deep neural networks. *Proceedings of the IEEE conference on computer vision and pattern recognition*, (pp. 3270-3278).

Laudon, K. C., & Laudon, J. P. (2012). *Management Information Systems Managing in the Digital Firm (12th ed.).* London: Person education.

Leishman, P. (2011). Is there Really any Money in Mobile Money? *Mobile Money for the Unbanked*, 2-5.

Lemmens, A., & Croux, C. (2006). Bagging And Boosting Classification Trees To Predict Churn. *Journal of Marketing Research*, 276-286.

Mandizha, T. (2013, July 18). *NewsDay Harare.* Retrieved from NewsDay: RBZ pushes for more financial inclusion. NewsDay.

Martin, R. (2018). *Understanding Sentiment Analysis and Sentiment Accuracy*. Retrieved from Understanding Sentiment Analysis and Sentiment Accuracy: http://blog.infegy.com/understanding-sentiment-analysis-and-sentiment-accuracy

Martin, R. (2018). Understanding Sentiment Analysis and Sentiment Accuracy. 85-91.

Mbunge, E., Vheremu, F., & Kajiva, K. (2017). A Tool to Predict the Possibility of Social Unrest Using Sentiments Analysis - Case of Zimbabwe Politics 2017 - 2018. *International Journal of Science and Research (IJSR)*.

Mitkees, I. M., Ibrahim, A., & Elseddawy, B. (2017). Customer Churn Prediction Model using Data . *Customer Churn Prediction Model using Data* , 262–268.

Olah, C. (2015, August 27). *Understanding LSTM Networks.* Retrieved from http://colah.github.io/posts/2015-08-Understanding-LSTMs/

Palangi, H., Li, D., Yelong, S., Jianfeng, G., Xiaodong, H., Jianshu, C., . . . Rabab, W. (2015). Deep Sentence Embedding Using Long Short-Term Memory Networks: Analysis and Application to Information Retrieval. *Transactions on Audio, Speech, and Language Processing*.

POTRAZ. (2020). *POTRAZ Annual Sector Performance Report 2020*. Retrieved from POTRAZ: http://www.potraz.gov.zw/?page_id=527

Reichheld, F. F., & Sasser, E. (1990). *Zero defections: quality comes to services.* London: Havard Review.

Rojas-Barahona, L. (2016). Deep learning for sentiment analysis. *Deep learning for sentiment analysis*, 701-719.

Saunders, M. (2019). Understanding research philosophy and approaches to theory development. In *Research Methods for Business Students.*

TechZim. (2021, 10 03). *TechZim - What impacted EcoCash* . Retrieved from TechZim: https://www.techzim.co.zw/2021/10/what-impacted-ecocash

Zhang, L., Wang, S., & Liu, B. (2018). Deep Learning for Sentiment Analysis : A Survey. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, 8.

Zhou, P., Shi, W., Tian, J., Qi, Z., Bingchen, L., Hongwei, H., & Bo, X. (2016). Attention-based bidirectional long short-term memory net-works for relation classification., (pp. 207–212). Berlin Germany.

Meng, Y., Shen, J., Zhang, C., & Han, J. (2019). Weakly-supervised hierarchical text classification. *33rd AAAI Conference on Artificial Intelligence, AAAI 2019, 31st Innovative Applications of Artificial Intelligence Conference, IAAI 2019 and the 9th AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2019*, 6826–6833. https://doi.org/10.1609/aaai.v33i01.33016826

Martin, R. (2018). *Understanding Sentiment Analysis and Sentiment Accuracy*. Retrieved from Understanding Sentiment Analysis and Sentiment Accuracy: http://blog.infegy.com/understanding-sentiment-analysis-and-sentiment-accuracy

Martin, R. (2018). Understanding Sentiment Analysis and Sentiment Accuracy. 85-91.

Mitkees, I. M., Ibrahim, A., & Elseddawy, B. (2017). Customer Churn Prediction Model using Data . *Customer Churn Prediction Model using Data* , 262–268.

Olah, C. (2015, August 27). *Understanding LSTM Networks.* Retrieved from http://colah.github.io/posts/2015-08-Understanding-LSTMs/

Pei, L., Abdel-Aty, M., & Jinghui, Y. (2020). Real-time crash risk prediction on arterials based on LSTM-CNN. *Accident Analysis and Prevention*, *135*(May), 105371. https://doi.org/10.1016/j.aap.2019.105371

Palangi, H., Li, D., Yelong, S., Jianfeng, G., Xiaodong, H., Jianshu, C., . . . Rabab, W. (2015). Deep Sentence Embedding Using Long Short-Term Memory Networks: Analysis and Application to Information Retrieval. *Transactions on Audio, Speech, and Language Processing*.

POTRAZ. (2020). *POTRAZ Annual Sector Performance Report 2020*. Retrieved from POTRAZ: http://www.potraz.gov.zw/?page_id=527

Reichheld, F. F., & Sasser, E. (1990). *Zero defections: quality comes to services.* London: Havard Review.

Rojas-Barahona, L. (2016). Deep learning for sentiment analysis. *Deep learning for sentiment analysis*, 701-719.

Reagan, A. J., Danforth, C. M., Tivnan, B., Williams, J. R., & Dodds, P. S. (2017). Sentiment analysis methods for understanding large-scale texts: a case for using continuum-scored words and word shift graphs. *EPJ Data Science*, *6*(1). https://doi.org/10.1140/epjds/s13688-017-0121-9

Saunders, M. (2019). Understanding research philosophy and approaches to theory development. In *Research Methods for Business Students.*

Shayaa, S., Jaafar, N. I., Bahri, S., Sulaiman, A., Seuk Wai, P., Wai Chung, Y., Piprani, A. Z., & Al-Garadi, M. A. (2018). Sentiment analysis of big data: Methods, applications, and open challenges. *IEEE Access*, *6*, 37807–37827. https://doi.org/10.1109/ACCESS.2018.2851311

Song, S., Huang, H., & Ruan, T. (2019). Abstractive text summarization using LSTM-CNN based deep learning. *Multimedia Tools and Applications*, *78*(1), 857–875. https://doi.org/10.1007/s11042-018-5749-3

Sosa, P. M. (2017). Twitter Sentiment Analysis using combined LSTM-CNN Models. *Academia.Edu*, 1–9.

S. Sun, C. Luo, and J. Chen, "A review of natural language processing techniques for opinion mining systems," Information Fusion, vol. 36, pp. 10–25, 2017.

TechZim. (2021, 10 03). *TechZim - What impacted EcoCash* . Retrieved from TechZim: https://www.techzim.co.zw/2021/10/what-impacted-ecocash

Tan, J. H., Hagiwara, Y., Pang, W., Lim, I., Oh, S. L., Adam, M., Tan, R. S., Chen, M., & Acharya, U. R. (2018). Application of stacked convolutional and long short-term memory network for accurate identification of CAD ECG signals. *Computers in Biology and Medicine*, *94*(January), 19–26. https://doi.org/10.1016/j.compbiomed.2017.12.023

Vo, Q. H., Nguyen, H. T., Le, B., & Nguyen, M. Le. (2017). Multi-channel LSTM-CNN model for Vietnamese sentiment analysis. *Proceedings - 2017 9th International Conference on Knowledge and Systems Engineering, KSE 2017*, *2017-Janua*(December), 24–29. https://doi.org/10.1109/KSE.2017.8119429

Wu, Y., Zheng, B., & Zhao, Y. (2019). Dynamic Gesture Recognition Based on LSTM-CNN. *Proceedings 2018 Chinese Automation Congress, CAC 2018*, *1*, 2446–2450. https://doi.org/10.1109/CAC.2018.8623035

Xia, K., Huang, J., & Wang, H. (2020). LSTM-CNN Architecture for Human Activity Recognition. *IEEE Access*, *8*, 56855–56866. https://doi.org/10.1109/ACCESS.2020.2982225

Xu, G. (2019). *Sentiment Analysis of Comment Texts Based on BiLSTM*. *7*. https://doi.org/10.1109/ACCESS.2019.2909919

Yao, L., Mao, C., & Luo, Y. (2019a). Clinical text classification with rule-based features and knowledge-guided convolutional neural networks. *BMC Medical Informatics and Decision Making*, *19*(Suppl 3). https://doi.org/10.1186/s12911-019-0781-4

Yao, L., Mao, C., & Luo, Y. (2019b). Graph Convolutional Networks for Text Detection. *The Thirty-Third AAAI Conference on Artificial Intelligence (AAAI-19)*, 19. www.aaai.org

Zhang, B., Zhang, S., & Li, W. (2019). Bearing performance degradation assessment using long short-term memory recurrent network. *Computers in Industry*, *106*, 14–29. https://doi.org/10.1016/j.compind.2018.12.016

Zhang, J., Wang, P., Yan, R., & Gao, R. X. (2018). Long short-term memory for machine remaining life prediction. *Journal of Manufacturing Systems*, *48*(November), 78–86. https://doi.org/10.1016/j.jmsy.2018.05.011

Zhang, L., Wang, S., & Liu, B. (2018). Deep Learning for Sentiment Analysis : A Survey. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, 8.

Zhou, P., Shi, W., Tian, J., Qi, Z., Bingchen, L., Hongwei, H., & Bo, X. (2016). Attention-based bidirectional long short-term memory net-works for relation classification., (pp. 207–212). Berlin Germany.

Zhou, H., Zhang, Y., Yang, L., Liu, Q., Yan, K., & Du, Y. (2019). Short-Term photovoltaic power forecasting based on long short term memory neural network and attention mechanism. *IEEE Access*, *7*, 78063–78074. https://doi.org/10.1109/ACCESS.2019.2923006

Zhu, A., Wu, Q., Cui, R., Wang, T., Hang, W., Hua, G., & Snoussi, H. (2020). Exploring a rich spatial–temporal dependent relational model for skeleton-based action recognition by bidirectional LSTM-CNN. *Neurocomputing*, *414*, 90–100. https://doi.org/10.1016/j.neucom.2020.07.068

Zhu, F., Ye, F., Fu, Y., Liu, Q., & Shen, B. (2019). Electrocardiogram generation with a bidirectional LSTM-CNN generative adversarial network. *Scientific Reports*, *9*(1), 1–11. https://doi.org/10.1038/s41598-019-42516-z

Zhu, Y., & Xiong, Y. (2015). *Towards data science*. Data Science Journal. https://doi.org/10.5334/dsj-2015-008