2021

Twitter sentiment analysis tool for detecting crime hotspots: a case of Nairobi, Kenya.

Onyango, Kevin Omondi
Faculty of Information Technology
Strathmore University

Twitter Sentiment Analysis Tool for Detecting Crime Hotspots: A Case of Nairobi, Kenya

Onyango Kevin Omondi

A Thesis Submitted to the School of Computing and Engineering Sciences in partial fulfilment of the requirements for the award of Master of Science in Information Technology at Strathmore University

Faculty of Information Technology

Strathmore University

Nairobi, Kenya

October, 2021

Declaration

I declare that this work has not been previously submitted and approved for the award of a degree by this or any other University. To the best of my knowledge and belief, the dissertation contains no material previously published or written by another person except where due reference is made in the dissertation itself.

Kevin Omondi Onyango

……..

……..

Approval

The thesis of Onyango Kevin Omondi was reviewed and approved by the following:

Dr. Joseph Orero, PhD.
Senior Lecturer, School of Computing and Engineering Sciences,
Strathmore University.

Dr. Bernard Shibwabo, PhD.
Director, Directorate of Graduate Studies,
Strathmore University.

……

……..

……

Abstract

Insecurity brought about by crime continues to be a major thorn in the flesh of citizens leaving in Kenya's urban centres. Although, incidents of crime are frequently reported from different regions in Kenya, there are more cases being reported from Kenya's urban centres than the rural ones, especially the informal settlement areas. This has been attributed to the rapid urbanisation of Kenya's major towns. Violent crimes are costly. Murders, rapes, assaults, and robberies impose concrete economic costs on the victims who survive as well as the families of those who lose their lives, in the loss of earnings and their physical and emotional tolls. The Kenyan government has invested heavily in setting up Internet Protocol Circuit Television Cameras (IP CCTVs) in Nairobi's Central Business District in a bid to curb crime. In 2015, the government implemented a community policing strategy at various levels namely, market, estate, house level among others. This community policing is known as Nyumba Kumi. The culture in urban centres especially Nairobi makes it very difficult to implement Nyumaba Kumi. For instance, Nairobi is a city where people are less concerned with the affairs of their neighbours. For Nyumba Kumi to be effective in Nairobi, a culture change has to occur. Culture changes usually take time. On the other hand, CCTVs have proven to be a useful tool in tracking down criminals and bringing them to book. However, maintaining CCTVs is quite expensive and CCTV footages have been reported missing in some cases whenever investigators needed them. Data mining algorithms can be employed to fetch useful patterns on Social Media posts especially Tweets from Twitter to monitor crime. This study proposed a Twitter sentiment analysis tool which was used to detect crime hotspots in Nairobi. The tool employed machine learning techniques to a build binary classifier in detecting crime hotspots. This research fetched sample crime relevant tweets from Twitter which were used to build the corpora. Then a Support Vector Machine model was trained and validated based on the labelled text data using bigram features and term frequency-inverse document frequency weighting. In order to determine what combination of features provided the most desirable performance outcome on the data collected, the SVM model was compared to Naive Bayes, K-nearest neighbour and Random forest machine learning algorithms. Based on the results from the experiments, it was found that the best way to create a model for detecting crime hotspots using Twitter is the use of an SVM machine learning algorithm with bigram features weighted using tf-idf. The SVM model produced an accuracy of 88% making it the most accurate compared to the rest.

Keywords: Insecurity; Crime; Twitter; Machine Learning; Support Vector Machine, TF-IDF, Bigram

Table of Contents

Declaration ii

Abstract iii

List of Figures viii

List of Equations ix

List of Tables x

Abbreviations/ Acronyms xi

Acknowledgements xii

Dedication xiii

Chapter 1: Introduction 1

1.1 Background of the Study 1

1.2 Problem Statement 2

1.3 Objectives 3

1.3.1 General Objectives 3

1.3.2 Specific objective 3

1.4 Research questions 3

1.5 Justification for the Study 3

1.6. Scope and Limitations 4

Chapter 2: Literature Review 5

2.1 Introduction 5

2.2 Crime in Kenya 5

2.3 Crime in Nairobi 5

2.4 Crime Hotspots Detection in Kenya 6

2.5 Machine Learning Approach to Detecting Crime Hotspots In Nairobi 6

2.6 Machine Learning (ML) Algorithms 7

2.6.1 Supervised learning 7

2.7 Document Representation 10

2.8 Related Works 11
2.8.1 Crime prediction using Twitter sentiment and weather 11
2.8.2 Crime pattern detection using online social media 11
2.9 Comparative Analysis 11
2.9.1 Gaps Identified in Related Systems 11
2.10 Conceptual Framework 12
Chapter 3: Methodology 14
3.1 Introduction 14
3.2 System Development Methodology 14
3.2.1 Requirements Planning 14
3.2.3 System Design 14
3.2.4 Development 15
3.2.5 Implementation 15
3.3 Research Design 15
3.4 Data Collection 15
3.4.1 Mining Twitter 16
3.4.2 Bag Of Words Model 16
3.4.3 Construction of Corpus 17
3.4.4 Preprocessing 18
Chapter 4: System Analysis, Design and Architecture 19
4.1 Introduction 19
4.2 System Analysis 19
4.2.1 Requirement Gathering 19
4.2.2 Functional Requirements 19
4.2.3 Non-Functional Requirements 20
4.3 System Architecture 20
4.4 System Designs 21

4.4.1 Use Case Diagram 21
4.4.2 Sequence Diagram 24
4.4.3 Context Diagram 25
4.4.4 Data Flow Diagram 26
Chapter 5: System Implementation and Testing 27
5.1 Introduction 27
5.2 Building the corpus 27
5.3 Preprocessing 27
5.4 Determining positive and negative tweets 32
5.5 Identifying a Location's General Score 34
5.6 Training the SVM model 34
5.7 Testing the model 36
5.7.1 Precision 37
5.7.2 Recall 37
5.7.3 F-measure 37
5.8 ROC Curve for the SVM model 38
5.9 Using the model to detect crime hotspots 38
Chapter 6 : Discussions 41
6.1 Sentiment analysis experiments 41
6.1.1 Using different classifiers 41
6.1.2 Experiment 1: SVM with different feature types 45
6.1.3 Experiment 2: K-nearest neighbour with different feature type 45
6.1.4 Experiment 3: Random Forest with different feature type 46
6.1.5 Experiment 4: Naive Bayes with different feature type 46
6.2 Discussions 47
Chapter 7: Conclusion and Recommendation 48
7.1 Conclusion 48

7.2 Recommendations.....48
7.3 Further work.....49
References 50

List of Figures

Figure 2.1 Supervised Learning overview 7
Figure 2.2 Linear Support Vector Machine Classifier 8
Figure 2.3 KNN algorithm decision process 9
Figure 2.4 Conceptual Framework 13
Figure 3.1 Rapid Application Development (RAD) 14
Figure 3.2 CountVectorizer() function from the Sk-learn library in Python 17
Figure 4.1 System Architecture 21
Figure 4.2 Use Case Diagram 22
Figure 4.3 Sequence Diagram 25
Figure 4.4 Context Diagram 25
Figure 4.5 Data Flow Diagram 26
Figure 6.1 Sample code for retrieving tweets 27
Figure 5.2 Raw JSON data from Twitter search API 28
Figure 5.3 Sample preprocessed tweets 29
Figure 5.4 Sample code for cleaning tweets 30
Figure 5.5 Sample clean tweets 30
Figure 5.6 Labelled tweets 31
Figure 5.7 VADAR Sentiment Analyser code snippet 33
Figure 5.8 Possible Hyperplanes 35
Figure 5.9 Implementation of SVM Model 36
Figure 5.10 ROC Curve for the SVM Classifier 38
Figure 5.11 Usalama home screen 39
Figure 6.1 ROC Curve for the Naive Bayes Classifier 42
Figure 6.3 ROC Curve for the Random Forests Classifier 43
Figure 6.4 ROC Curve for the KNN Classifier 44

List of Equations

Equation 2.1 Prediction rui^ 9
Equation 5.1 General Location Score 34
Equation 5.2 Precision 37
Equation 5.3 Recall 37
Equation 5.4 F-measure 37

List of Tables

Table 2.2 Parameters for Equation 1 10
Table 3.1 Example categorized tweets.....18
Table 4.1 Enter Location, Search Tweets, Retrieve Tweets 22
Table 3.2 Pre-process Tweets, Transform Features, Classify Tweets 23
Table 4.3 Receive Feedback 24
Table 5.1 Labelling Of Tweets.....31
Table 5.2 Lexicon-Classification Performance.....32
Table 5.3 Output from the confusion matrix 36
Table 5.4 Values from the confusion matrix 36
Table 6.1 Naive Bayes Results.....41
Table 6.2 Random Forest Results.....42
Table 6.3 K-nearest neighbour Results 43
Table 6.4 Performance of Different Classifiers 44
Table 6.5 SVM Performance.....45
Table 6.6 K-nearest neighbour's Performance.....45
Table 4.7 Random Forest Performance.....46
Table 6.8 Naïve Bayes Performance.....46

Abbreviations/ Acronyms

AI	Artificial Intelligence
CCTV	Circuit Television Cameras
IP	Internet Protocol
NCRC	The National Crime Research Center
SVM	Support Vector Machine
tf-idf	frequency-inverse document frequency weighting
CSV	comma-separated values

Acknowledgements

I would start by acknowledging the almighty God the creator of heaven and earth for His grace, love and immense favour during the entire process of preparing this research. Secondly, I would like to acknowledge the efforts and motivation of my Supervisor, Dr. Joseph Orero for his continuous motivation and guidance to see me through the completion of my work.

I wish to also thank Dr. Vincent Omwenga, the Research Director at Strathmore University for providing insights on the processes of carrying out a research.

Dedication

To my parents Mr. and Mrs. Moses Onyango for believing in me, supporting me throughout my education journey and instilling in me the skills that have pushed me to this level that I am in today.

Chapter 1: Introduction

1.1 Background of the Study

Insecurity caused by criminal activities has been a major, dark and strong undercurrent in human society throughout history. Many countries around the world have put up different measures for crime prevention. Various technologies and methodologies have been implemented ever since the introduction of the World Wide Web. Dubai for instance, launched an E-Crime platform in 2019, a tool that enables online crime victims to report crimes (Gibbs, 2020). However, there are still a lot of offline crimes such as rape, robberies, assaults and murders that a technology such as E-Crime would not be able to cub.

Kenya, like many other countries, has been reporting an increase in criminal activities year in year out. According to the 2018 Annual Crime Report released by Kenya Police, in the year 2018 there were 88,268 reported cases as compared to 77,992 in 2017 which was an increase of 10,276 cases or 13%. The major increases were recorded in the individual crimes of Possession of Dangerous Drugs (Cannabis Sativa) by 2,268 cases, followed by Assault 1,544, Defilement 1,450 and General Stealing 1,258.

Insecurity caused by crime hurts a country's economic growth and development. It also has a huge impact on social cohesion, governance and general state stability. On individual level, crime imposes economic costs on the victims who survive as well as the families of those who lose their lives, in the loss of earnings and their physical and emotional tolls (Dinisman & Moroz, 2017).

A study done by (Oduor et al., 2014a) on The Adoption of Mobile Technology as a Tool for Situational Crime Prevention in Kenya found that most of the crimes reported in Kenya could be solved by employing technologies that are convenient to the public and the police. The objective of this study was to digitise police operations in Kenya and to develop a mobile application that the public can use to report criminal incidents to the police. However, this study entirely depends on Kenya Police choosing to go with the researchers' solution. In which case the Kenya Police have not adopted it yet.

This study proposes the development of a Twitter Sentiment Analysis Tool that will have the public report insecurity incidents on Twitter. The tool will then fetch these tweets inclusive of geolocation and categorize them into three security risk codes namely; High Risk (code Red), Medium Risk (code Orange) and Low Risk (code Green). From this tool, which will be in form of a mobile application, the public will be able to check the security of various locations from a tap of a button. It will aide in informing the public on the security status of various locations.

1.2 Problem Statement

Currently, the crime hotspots detection methods applied have their own shortcomings. For instance, a study done on the effectiveness of the Nyumba Kumi initiative ("Effectiveness of the Nyumba Kumi community policing initiative in Kenya | Wangari Maathai Institute for Peace and Environmental Studies," n.d.) finds that more youths should be included for it to achieve its fullest potential. The Kenyan government has a central database which contains citizens' data. This database is linked to other databases such as the National Transport and Safety Authority database, Kenya Revenue Authority and Kenya Lands Commission. These databases coherently aid in pursuing and managing crime (Samoei, 2018). These databases provide a reactive rather than a proactive approach to cabbing crime. Crimes reported in Kenya could be solved by employing technologies that are convenient to both the public and the police (Oduor et al., 2014b).

Twitter, a social media platform that has over 140 million users who post 340 million tweets in a day provides ("Predicting Crime Using Twitter and Kernel Density Estimation | Request PDF," n.d.) a perfect data source where mining algorithims can be employed to fetch useful patterns from posted tweets in order to monitor crime.

This study proposes a Twitter Sentiment Analysis Tool that applies machine learning techniques to automatically fetch tweets that are related to crime and tie them to particular locations. These tweets will then be classified and the end result is to classify locations into three security risk codes. These codes are, High Risk (code Red), Medium Risk (code Orange) and Low Risk (code Green). With locations classified based on their security risks, the public and relevant authorities will be aware of high security risk areas and will be able to take

required actions. The public will be proactive by avoiding these areas while relevant authorities will use this information to put up measures that will help improve the security of these areas.

1.3 Objectives

1.3.1 General Objectives

The purpose of this study is to develop a Twitter Sentiment Analysis Tool for detecting crime hotspots mainly in urban areas.

1.3.2 Specific objective

i. To investigate existing techniques used in detecting crime hotspots,
ii. To review the current machine learning techniques used to detect crime hotspots,
iii. To develop a system for detecting crime hotspots,
iv. To test the system for detecting crime hotspots.

1.4 Research questions

i. What are the existing techniques used to detect crime hotspots?
ii. What are the current machine learning techniques used to detect crime hotspots?
iii. How can a system for detecting crime hotspots be developed?
iv. How can the system for detecting crime hotspots be tested?

1.5 Justification for the Study

Kenya needs a strategic security tool that will provide information to both the public and relevant authorities on security (Oduor et al., 2014b). This will aid in preparedness and development of a response plan that can help the country improve its security in urban areas. Insecurity wreaks havoc on all our lives including our economy and social fabric. Insecurity affects everyone, it does not spare anyone based on their social status, age or even gender (Dinisman & Moroz, 2017). With rising incidents of insecurity in recent years, the county's sustainable growth is highly hindered. Insecurity comprises a part of dilemmas of urban economic development that include persistent poverty, slow economic growth, labour market

difficulties and shifting demographics. With lack of sufficient proactive mechanisms put in place, the most vulnerable in our society are left exposed.

The need to protect property, human injury, loss of lives and fight against crime continues to be an urgent task for authorities. It has been proven that for a country to attract investors both foreign and local, its urban centres ought to have proper security (Rodríguez-Pose & Cols, 2017).

A tool that provides security information to the public and relevant authorities enables a proactive as opposed to a reactive approach towards fighting crime. People will be more aware of places to avoid while the relevant authorities will beef up security in high risk areas. The end result is having more secure urban centres which in turn will lead to both economic development and stronger social cohesion.

Classification of texts into positive and negative mentions about the security of particular locations is an important technique for handling information retrieval. Currently Kenya police is relying on CCTVs, plain clothes undercover police and community policing to gather intelligence information. Classification of huge amount of tweets gathered from different locations at once and in real-time remarkably reduces police response turnaround time and could not only save lives and property but also prevent crime from happening.

1.6. Scope and Limitations

While this study acknowledges that there is a need for both offline and online security information gathering. The study focuses mainly on developing A Twitter Sentiment Analysis Tool for Detecting crime hotspots that will run on mobile devices. This tool will only consider tweets expressed in English. Tweets made in other formats such as videos, images and audios will not be considered.

Chapter 2: Literature Review

2.1 Introduction

This section provides a review of relevant literature to further understand the concept and investigate the research problem. The current status of insecurity brought about by crime in Kenya's urban areas and the various processes for detecting crime hotspots is reviewed. Relevant and significant publications and studies are further reviewed to understand the application of machine learning techniques in text classification. A conceptual framework is then presented at the conclusion of the literature review.

2.2 Crime in Kenya

Crime continues to be a major concern to citizens, Kenya police and the entire government of Kenya. As much as crime incidents have been reported across the nation, urban centres especially informal settlement have more cases of insecurity being reported. Since 2011, Security Research and Information Centre (SRIC), a body that supports the Government of Kenya through National Steering Committee on Peace Building and Conflict Management and Kenya National Focal Point on Small Arms and Light Weapons with the help of UNDP Kenya has been conducting crime surveys in its capacity as a crime observatory. These surveys on crime have consistently found that crime is a major threat to peace and security in Kenya's urban centres especially the slums (Musoi et al., 2014).

2.3 Crime in Nairobi

According to a study of Crime In Urban Slums In Kenya done by the Security Research and Information Center (Musoi et al., 2014). It was established that a relationship existed between crime occurrence and the general environment in which it occurs. In general, informal settlements have more crime hotspots than affluent, well planned residential estates. This is because the residents of affluent residential areas can afford to invest in better security around their estates. The Nature, Challenges and Consequences of Urban Youth Unemployment: A Case of Nairobi City, Kenya (Muiya, 2014) found that most youth in Mathare were unemployed due to lack of education and necessary skills. This study also found that cases of rape, marginalization and early marriages in Mathare was a result of urban youth

unemployment. Another study done by Lucy Mburu (University of Salzburg, 2015) found that there is a correlation between the weather and crime. It was observed that during rainy seasons there was a spike in crime rate.

2.4 Crime Hotspots Detection in Kenya

The need for coming up with various methods of fighting crime in Kenya was recognized by the government following the serious threat that insecurity poses.

In 1997, The National Crime Research Center (NCRC), a State Corporation under the office of the Attorney General and Department of Justice came into existence. This was through an act of parliament, the National Crime Research Act (1997 CAP 62 Laws of Kenya). NCRC was mandated to carry out research on causes of crime, how to prevent them and to disseminate the research findings and recommendations to Government Agencies concerned with the administration of criminal justice, NCRC's stakeholders and the public.

In 2017, NCRC launched an android mobile application for reporting and monitoring crime. This was a paradigm shift in fighting crime through research by use of pocket technology (mobile phone). The application collects, collates, reports and uses data to make individuals and communities more inclusive, safer and free by giving them opportunity to report crime incidences in specific localities. Based on the application's reviews on Google Play Store, most Kenyans think that it is a good starting point though, more publicity about the application and user experience improvement need to be done.

2.5 Machine Learning Approach to Detecting Crime Hotspots In Nairobi

Numerous studies have been conducted on sentiment analysis using both unsupervised and supervised learning techniques. The global Artificial Intelligence (AI) market size was valued at USD 39.9 billion in 2019 and is expected to grow at a compound annual growth rate (CAGR) of 42.2% from 2020 to 2027 ("Artificial Intelligence Market Size & Share Report, 2020-2027," n.d.). This growth has made it possible for this study to provide an algorithm that could help in analysing text data on Twitter with the intention of detecting crime hotspots in Nairobi. Classification is a subcategory of supervised learning where the aim is to predict the categorical class labels (discrete, unordered values, group membership) of new instances based on past observations.

2.6 Machine Learning (ML) Algorithms

Machine learning (ML) is made up of a wide range of algorithms and modelling tools that are used for numerous array of data processing tasks (Carleo et al., 2019). Machine learning techniques are better placed to tackle real world problems with high dimensionality. This section discusses the two major classifications of machine learning algorithms. Machine learning algorithms are classified into Supervised learning and Unsupervised learning.

2.6.1 Supervised learning

Supervised learning is the search for machine learning algorithms that reason from cases that are provided externally to construct broad hypotheses. These hypotheses are able to predict future instances (Akinsola, 2017). Intelligent systems highly depend on Supervised Learning to perform vital tasks. Supervised Learning includes Machine Learning algorithms that deduce patterns from a set of input (the X's) and the desired output (Y). Figure 2.1 below depicts an overview of Supervised Learning.

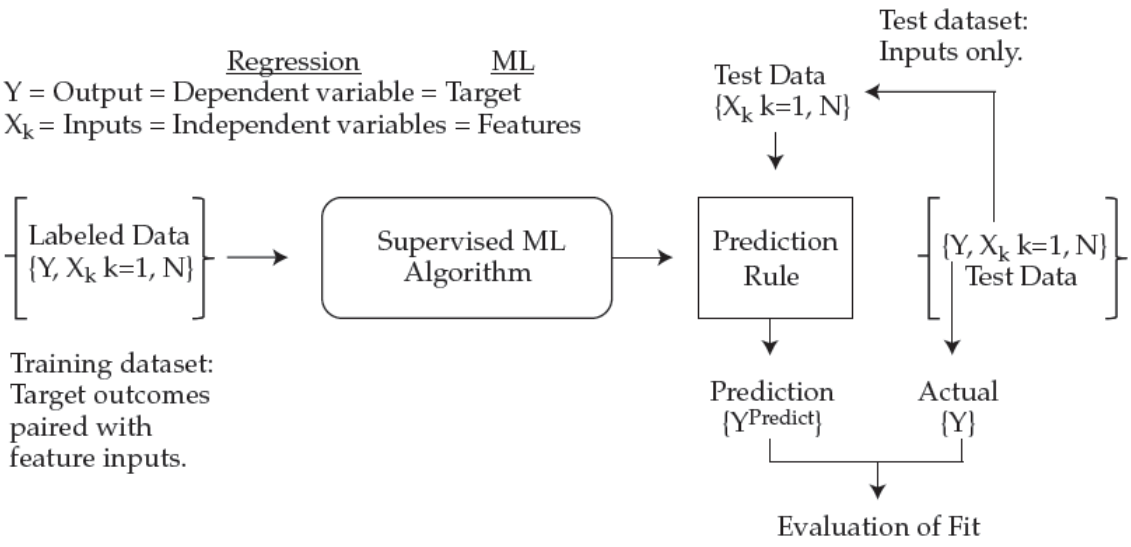


Figure 2.1 Supervised Learning overview

2.6.1.1 Support Vector Machines

A highly used supervised learning model with associated learning algorithms that is based on a statistical concept in Machine Learning is known as Support vector machine (SVM). The main aim of a Support Vector Machine algorithm is increasing the probability of making an accurate prediction finding out the observation that is furthest away from all observations. Figure 2.2 below shows a perfect classification of an observation by a Support Vector Machine.

The Support Vector Machine separates the data by the highest margin, where the margin is the shaded strip that separates the observations into two sets. The straight line in the middle of the shaded strip is the boundary that does the separation.

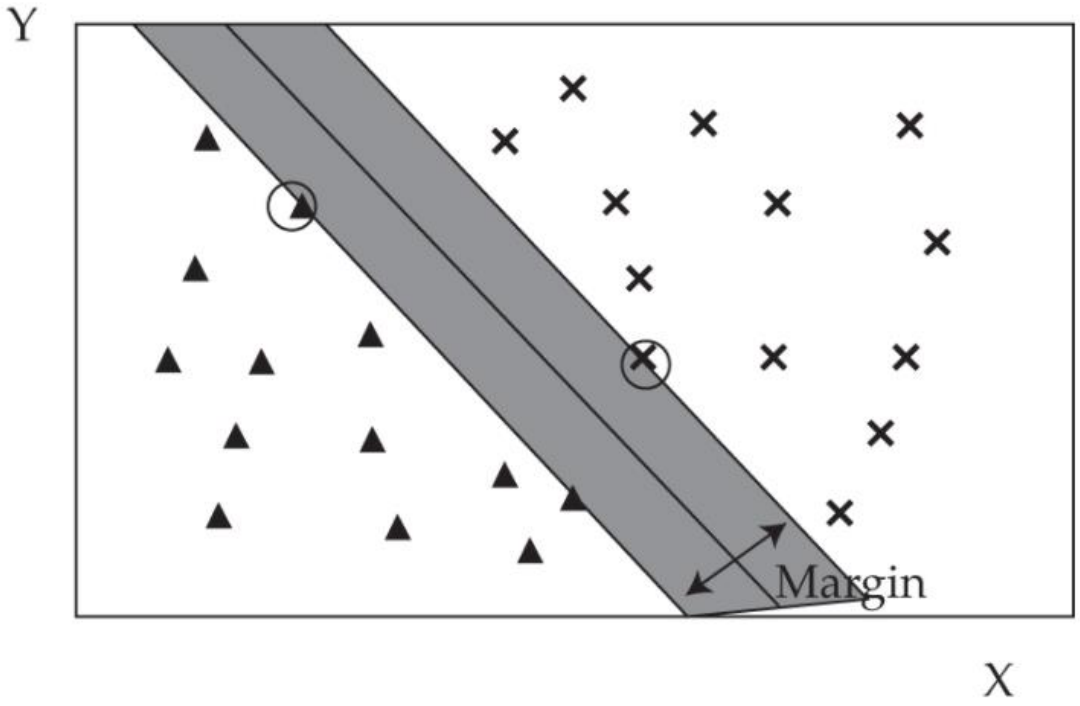


Figure 2.2 Linear Support Vector Machine Classifier

2.6.1.2 K-nearest neighbour

What K-nearest neighbour algorithm seeks to achieve is that, if most of the k samples of the most acacia of a sample belong to a particular set, the sample belongs to this set as depicted in Figure 2.3 below (Li and Zhang, 2018).

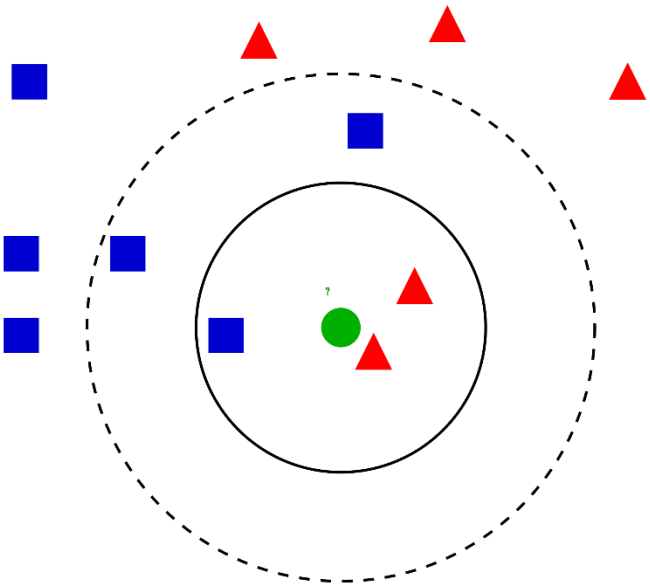


Figure 2.3 KNN algorithm decision process

The prediction rui^ is set as:

$$\widehat{r_{ui}} = \frac{\sum_{v \in N_i^k(u)} sim(u,v)^* r_{vi}}{\sum_{v \in N_i^k(u)} sim(u,v)}$$

Equation 2.1 Prediction rui^

Table 2.1 below displays the parameters for Equation 2.1.

Table 2.1 Parameters for Equation 1

parameters	meaning
$\widehat{r_{ui}}$	the estimated rating of user u for item i.
$N_i^k(u)$	the k nearest neighbors of user u that have rated item i.
r_{vi}	the true rating of user v for item i.
Sim_options(dict)	a dictionary of options for the similarity measure.
k(int)	the number of neighbors to take into account for aggregation.

2.6.1.3 Naïve Bayes

Naive Bayes has greatly been implemented in document classification. It has proven to produce very high performance. The main aim is to use the joint probabilities of words and categories to estimate the probabilities of categories given a document (Chirawichitchai, 2013).
This algorithm assumes that each input variable is independent. It is quite a naive assumption to make when it comes to real-world data sets. For instance, if Naive Bayes is used for sentiment analysis, given a phrase 'I like Strathmore', the algorithm will take into consideration the individual words as opposed to the full sentence.

2.7 Document Representation

It is prudent to convert text into a format that a machine learning algorithm can process because text data cannot be processed as it is. Bag of Words (BOW) and Vector Space Model (VSM) are techniques that can be used to convert text into a format that a machine learning algorithm can process. BOW represents documents as a collection of words without any order but maintains their multiplicity. Every unique word found in the corpus forms the dictionary. Every document is then represented in a vector of word frequencies. This model assumes that: the order of words does not matter and that words are independent of each other. Furthermore, this model does not allow for weighting of terms in specific documents (Mazzonello et al., 2013).

2.8 Related Works

2.8.1 Crime prediction using Twitter sentiment and weather

Social Media has been used in recent times to reveal valuable insights when statistical analysis is applied to the unstructured data. As found by a previous research GPS-tagged Twitter data enabled the prediction of future crimes in Chicago, Illinois, of the United States of America. The crime prediction model that uses data from Twitter are incapable of describing criminal incidents due to the lack of sentiment polarity and whether factors. This study (Chen et al., 2015) included the addition of sentiment analysis and weather predictors which provided vital insights on crime. The aim of this particular study was to predict the time and location in which a specific type of crime will occur. This study's approach was based on sentiment analysis by applying lexicon-based methods and understanding of categorized weather data, combined with kernel density estimation based on historical crime incidents and prediction via linear modelling.

2.8.2 Crime pattern detection using online social media

In this study (Bolla, 2014), crime is defined as harm caused not only to individuals involved, but also to the entire community as a whole. A filter was designed to extract tweets from cities deemed to be either the most dangerous or the safest in the United States of America. A geographic analysis revealed a correlation between these tweets and the crimes that occurred in the corresponding cities. Over 100,000 crime-related tweets were collected over a period of 20 days. Sentiment analysis techniques were conducted on these tweets to analyse the crime intensity of a particular location. The aim of this study was to help reveal the crime rate of a certain location in real-time. Although the results of this test helped in detecting crime patterns, the sentiment analysis techniques did not always guarantee the proper results.

2.9 Comparative Analysis

2.9.1 Gaps Identified in Related Systems

The reviewed systems above have gaps that this study seeks to fill.

This study (Chen et al., 2015), uses lexicon-based methods in understanding of categorized weather data, combined with kernel density estimation based on historical crime incidents and prediction via linear modelling. The disadvantage of this approach is the assignment of positive

sentiment values to negative phrases. For instance, in sentence “I don’t hate this city”, the sentiment assigned to the sentence will be -100 (“hate” has value −100 in the lexicon) and the sentence will be considered as negative.

The other study (Bolla, 2014) uses ANEW based technique to map every term from ANEW to its equivalent in the tweet. ANEW is also lexicon-based and has the same shortcoming as the other study as stated above.

2.10 Conceptual Framework

According to the literature reviewed and the numerous gaps identified, this study proposes the following conceptual framework to detect crime hotspots in Nairobi. Tweets that are crime related and contain location data will be collected from twitter and used to create the corpus which will be used for learning.

The tweets will be labelled as either secure or insecure spot. During the pre-processing phase, tweets will go through a cleaning process where stop words will be removed. Cleaning will involve the removal of punctuation marks and conversion of text to lower case. The tweets will be represented in a document-term matrix, using unigram terms with TF-IDF feature weighting.

Support vector machine will then be employed in order to learn a model for detecting crime hotspots from the training set. The reason why SVM is preferred is because of its ability to be effective even in highly dimensional spaces.

The performance of the model will be rated based on the metrics: accuracy, precision, recall and the F-Score. Once the model has reached a reasonably acceptable level of performance with lesser error margin, it can be used to detect new instances of crime hotspots in other tweets. A user will define keywords to be used to fetch tweets that have not been observed from the Twitter Search Application Programming Interface (API). Figure 2.4 below represents the conceptual framework of the proposed system.

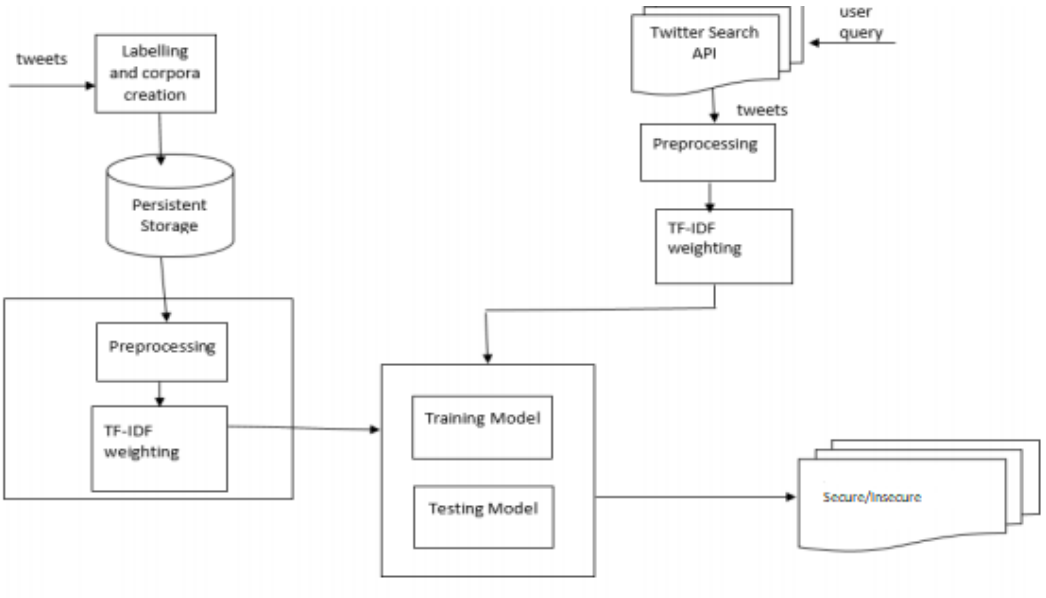


Figure 2.4 Conceptual Framework

Chapter 3: Methodology

3.1 Introduction

Research methodology is the path through which researchers need to conduct their research (Sileyew, 2019). It shows the path through which these researchers formulate their problem and objective and present their result from the data obtained during the study period. This chapter elaborates the development methodology that were used to develop the model, the research strategy and approaches to data collection and data analysis.

3.2 System Development Methodology

The proposed tool was developed using Rapid Application development methodology (RAD). Rapid Application Development (RAD) is a type of agile software development methodology that uses rapid prototype releases and iterations. Compared to Waterfall method, RAD stresses the use of software and user feedback over strict planning and requirements recording. RAD is suitable for this study

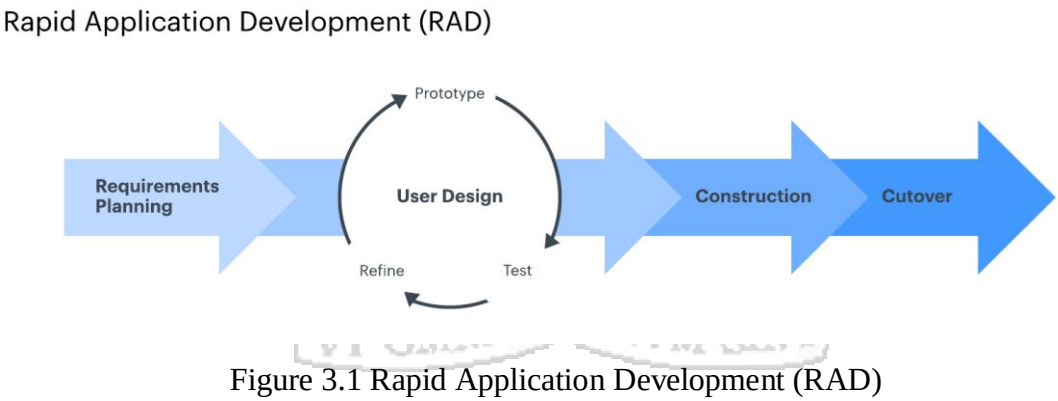


Figure 3.1 Rapid Application Development (RAD)

3.2.1 Requirements Planning

During this step, stakeholders sit together to define and finalize project requirements such as project goals, expectations, timelines, and budget. When they have clearly defined and scoped out each aspect of the project's requirements, they can seek management approvals.

3.2.3 System Design

In this phase, definition of elements of a system such as modules, architecture, components, their interfaces and data for a system are designed. The designs can be represented textual modelling languages or can be graphical. The designs are guided by system requirements. Both functional and non-functional.

3.2.4 Development

This stage constructs the prototypes and beta systems from the design phase and converts them into the working model. This phase can be broken down further into several smaller steps:

a) Preparation for rapid construction
b) Program and application development
c) Coding
d) Unit, integration, and system testing

3.2.5 Implementation

The implementation phase is where the finished product goes to launch. It includes data conversion, testing, and changeover to the new system. It also involves user training.
All final changes are made while the different stakeholders such as coders and clients continue to look for bugs in the system.

3.3 Research Design

The research design refers to the overall strategy that a researcher chooses to combine the different elements of the research in a coherent and logical way, thereby, ensuring that they will effectively address the research problem; it constitutes the blueprint for the collection, measurement, and analysis of data. The two research approaches are qualitative and quantitative methods. This study will implement both quantitative and qualitative research design when developing the system.

3.4 Data Collection

Data collection is the process of gathering and measuring information on variables of interest, in an established systematic fashion that enables one to answer stated research questions, test hypotheses, and evaluate outcomes.

3.4.1 Mining Twitter

Twitter has developed its own language conventions. Below are examples of these conventions.
1. “RT” is an acronym for retweet, normally is put infront of a tweet to show that a user is repeating or reposting someone else’s tweet.
2. “#” hashtag is used to mark and categorize tweets according to a certain topic.
3. “@myusername” represents that a message is a reply to a user whose user name is “myusername”.
4. Emoticons and colloquial expressions are regularly used in tweets, e.g. “:-)”, “lovvve”, “lmao”.
5. External Web links (e.g. http://afmze.ly/Ty4n0t) are also mostly found in tweets to refer to some external sources.
6. Length: Tweets are limited to 280 characters. This is quite different from the ordinary opinionated corpora like blogs and reviews which are normally lengthy.

The major characteristic that sets Twitter apart from other opinionated corpora is its volume. It is estimated that people post about 340 million tweets every day and the number is still increasing rapidly.

To build the corpora, crime related tweets were collected from Twitter. Twitter allows developers to access tweets using two APIs: the Representational State Transfer (REST) API and the Streaming API. Both APIs require the use of Open Authentication (OAuth) to allow applications to get access to them and issue responses in JavaScript Object Notation (JSON) format. The REST API allows developers to read/write Twitter data. A vital component of the REST API is the Search API which enables developers to query against indices of recent tweets up to 7 days old. The Streaming API enables developers to process tweets in real time continuously delivering responses in JSON format over long lived HTTP connections (Twitter, 2017).

3.4.2 Bag Of Words Model

Bag of Words (BOW) and Vector Space Model (VSM) are techniques that can be used to convert text into a format that a machine learning algorithm can process. BOW represents documents as a collection of words without any order but maintains their multiplicity. Every

unique word found in the corpus forms the dictionary. Every document is then represented in a vector of word frequencies. This model assumes that: the order of words does not matter and that words are independent of each other. Furthermore, this model does not allow for weighting of terms in specific documents (Mazzonello et al., 2013).

In this study, a statistical framework which generalized the Bag of Words representation was employed. Words were generated by a statistical process. Well defined fixed-length inputs were achieved through the use of BoW technique which converted variable-length texts into a fixed-length vector. Text was converted into its equivalent vector of numbers. Figure 3.2 below depicts how the CountVectorizer() function from the Sk-learn library in python was used to implement the BoW model using Python.

```
import pandas as pd
from sklearn import svm
from sklearn.feature_extraction.text import CountVectorizer

data = pd.read_csv(open("Twidb11.csv"), sep=' ')
count_vect = CountVectorizer()
X_train_counts = count_vect.fit_transform(data.Text)
count_vect.vocabulary_
```

Figure 3.2 CountVectorizer() function from the Sk-learn library in Python

3.4.3 Construction of Corpus

Data collected using the Twitter API was saved into a comma-separated values (CSV) file which formed the corpus. This dataset which contained over 7,800 crime relevant tweets was large enough and provided high quality data for training. Tweets from this dataset were either labelled as secure or insecure. The class label 1 was used for tweets that were found to be secure and label -1 were for tweets found to be insecure. Table 3.1 below has example tweets of a location in Nairobi.

Table 3.1 Example categorized tweets

Negative	1. I have been mugged in Mathare 2. There is a shootout in Mathare. 3. Two robbers have been killed in Mathare.
Positive	1. Kilimani enjoys random police patrol. 2. I feel safe walking in Kilimani at 1am in the morning.

3.4.4 Preprocessing

Data preprocessing will consists of three steps:

1) tokenization

2) normalization

3) part-of-speech (POS) tagging.

Emoticons and abbreviations (e.g., OMG, WTF, BRB) were identified as part of the tokenization process and treated as individual tokens. For the normalization process, the presence of abbreviations within a tweet were noted and then abbreviations were replaced by their actual meaning (e.g., BRB − > be right back). Informal intensifiers were also identified. Intensifiers such as all-caps (e.g., I LOVE Kilimani!!! and character repetitions (e.g., I've got home safe!! happyyyyyy"), note their presence in the tweet. All-caps words were converted to lower case, and instances of repeated characters were replaced by a single character.

Finally, the presence of any special Twitter tokens were noted (e.g., #hashtags, usertags, and URLs) and placeholders indicating the token type substituted. The intention was to have normalization improve the performance of the POS tagger, which was the last pre-processing step.

Chapter 4: System Analysis, Design and Architecture

4.1 Introduction

Numerous machine learning approaches that were reviewed assisted in selecting a suitable environment for the development of the application. To help in the study off how different components of the system interacted with each other, Use case and sequence diagrams were used. This chapter also sets out the general architecture and detailed design of the system. The various components that the system is comprised of are data collection, data cleansing, data classification, sentiment analysis and mapping of sentiments.

4.2 System Analysis

Systems Analysis and Design (SAD) is a broad term for describing methodologies for developing high quality Information System which combines Information Technology, people and Data to support business requirement.

This study mainly focused on developing a model for detecting crime hotspots in urban areas using Twitter. This section therefore outlines the various functional and non-functional requirements addressed by the system.

4.2.1 Requirement Gathering

Various stakeholders were interviewed. These stakeholders included random citizens and police officers. Data collected was analysed in the aid of coming up with the functional and non-functional requirements.

4.2.2 Functional Requirements

I. System should connect to Twitter and fetch tweets based on set keywords related to crime.
II. System should retrieve the metadata of each tweet along with text content and coordinates (the longitude and the latitude). The tweets are stored in the database with the following headers; tweet_id, time_created, tweet_text, sender, geo_location
III. The system should cleanse the retrieved tweets and store them in a database.

IV. The system should perform sentiment analysis on each tweet text and labelled them as follows; Hotspot or Secure.

V. System should Link the classified tweet text corresponding to the geographic coodinates on to the map Geo-coding plays an important role in representing physical location on visual maps

VI. The system should display to the user the tweets labelled as hotspot.

4.2.3 Non-Functional Requirements

4.2.3.1 Security

The system uses data fetched from Twitter to detect crime hot spots in Nairobi. The system's integrity is key hence the need to ensure robust security measures are put in place to protect its data. Any alteration on the systems' configuration is done only by authorised users.

4.2.3.2 Availability

Due to its critical nature the system should always be available for usage. This is because users will depend on it to get insights on the level of crime for particular locations in Nairobi. In the event that it is not available then its reliability will also be adversely affected.

4.2.3.3 Usability

The system will be used by users of all walks of life hence the need to be quite user friendly. Its graphical user interface should be simple and easy to navigate.

4.3 System Architecture

Figure 4.1 below shows the various components that make up the system. The components present the tasks that a user can carry out using the system. These tasks are; data collection, pre-processing, data classification, analysis and mapping of sentiment onto a chart.

Data Collection: The proposed system will collect the data or the input from Twitterbased on the location entered by the user. Preprocessing: The system will then Omit common words or verbs such as “is”,”are”,”now”,”its” and many others. The remaining data will proceed to Data classification. Data Classification: Here, the system will classify the data with Naïve Bayes

Algorithm and analyze it into Positive or Negativel. Finally they system will map the result onto a chart.

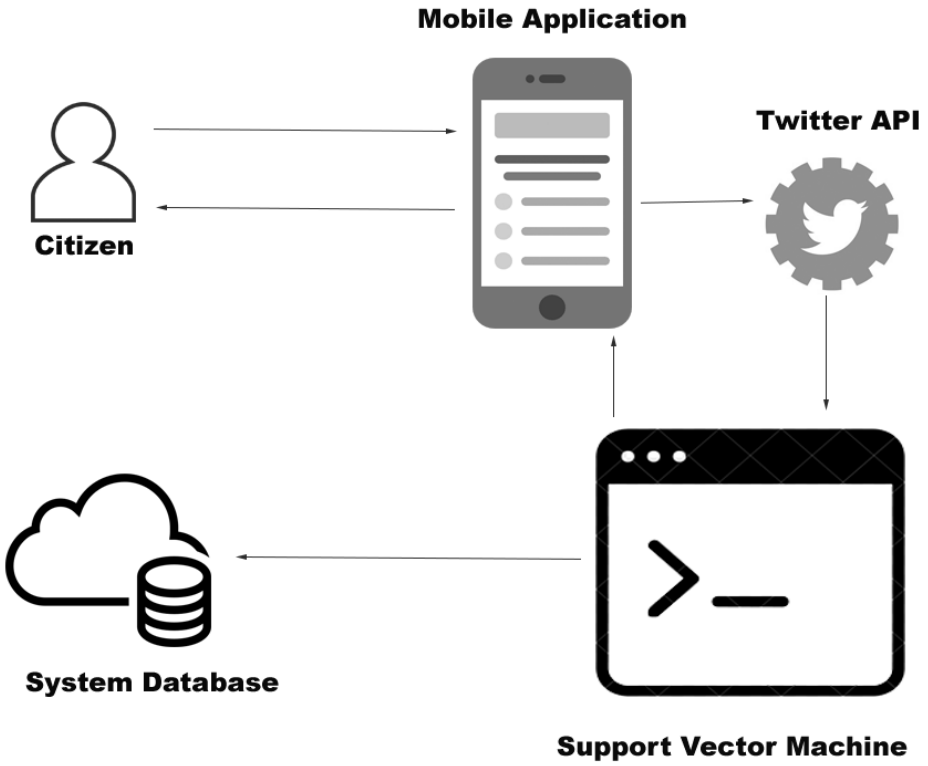


Figure 4.1 System Architecture

4.4 System Designs

4.4.1 Use Case Diagram

Use case diagrams are used to represent the different scenarios in which systems interact with users, organizations, or other external systems. Figure 4.2 below illustrates the scenarios between set actors and the hot spot detection system.

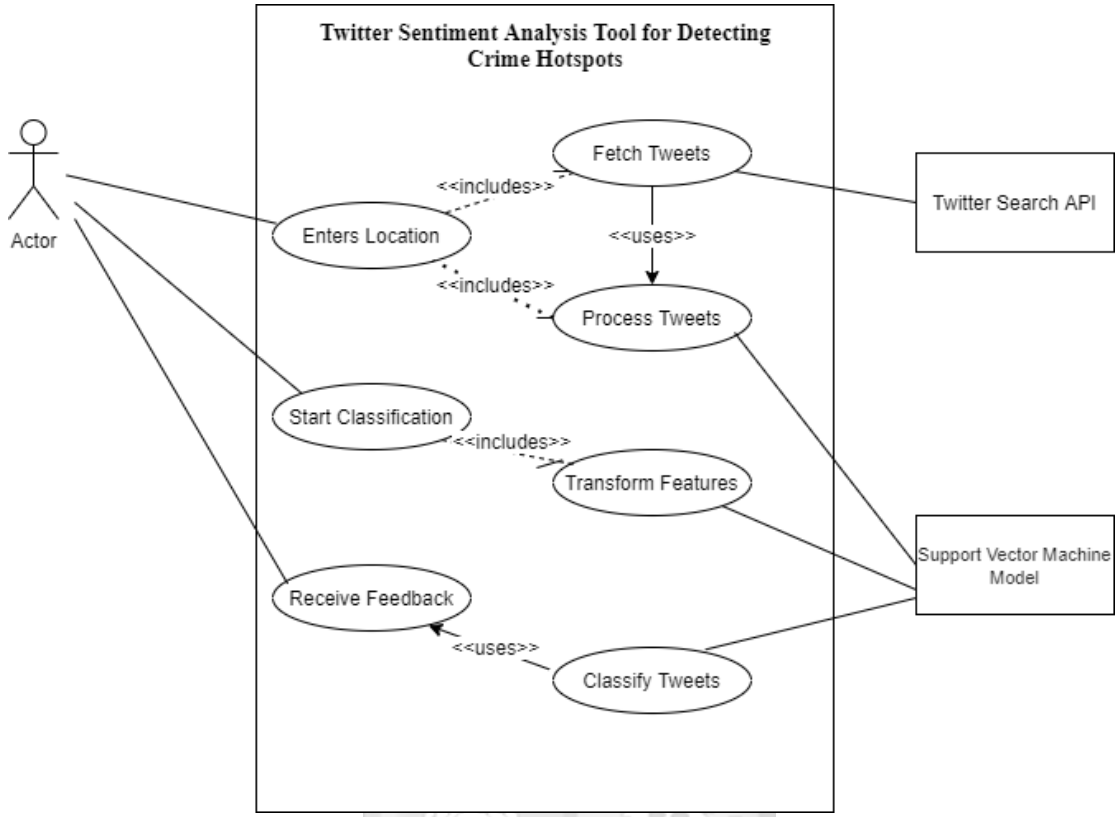


Figure 4.2 Use Case Diagram

Below are the Use Case Descriptions based on figure 4.2 above. The format employed is two column fully dressed.

Table 4.1 Enter Location, Search Tweets, Retrieve Tweets

Use Case Name	**Enter Location, Search Tweets, Retrieve Tweets**
Use Case Description	A user to enter location, the system searches and retrieves tweets based on user location
Actors	User Tweeter Search API
Pre-Condition	Search Tweets use case completed successfully User has access to internet on device used
Post-Condition	System retrieves tweets from Twitter Search API based on the location provided

Main Success Scenarios	**Serial Number**	**Steps**
Actors/Users	1	Enter the location
System Responsibility	2	Use location entered as parameter to be used to retrieve tweets
	3	Retrieve tweets from Twitter Search API using the parameters entered
	4	Save fetched tweets
Actors/Users	5	View fetched tweets
Extensions		No internet System shows error

Table 2.2 Pre-process Tweets, Transform Features, Classify Tweets

Use Case Name		**Pre-process Tweets, Transform Features, Classify Tweets**
Use Case Description		System to pre-process tweets, transform and classify into either insecure or secure
Actors		User System
Pre-Condition		Tweets were fetched and saved successfully
Post-Condition		Tweet accurately classified as secure or insecure
Main Success Scenarios	**Serial Number**	**Steps**
Actors/Users	1	User begins classification
System Responsibility	2	Pre-processes the tweets to clean them as per the Support Vector Machine model
	3	Converts tweets into document-term matrix format as per the Support Vector Machine model
	4	Classify tweets as Insecure or Insecure using the Support Vector Machine model

Table 4.3 Receive Feedback

Use Case Name		**Receive Feedback**
Use Case Description		User views feedback
Actors		User
Pre-Condition		Successful classification of the tweets by the system
Post-Condition		User gets result of a location labelled as either secure or insecure
Main Success Scenarios	**Serial Number**	**Steps**
Actors/Users	1	User requests results of classification
System Responsibility	2	Return the output of classification

4.4.2 Sequence Diagram

Figure 7 below shows the sequence of interactivity that occurs between user and the system. It also demonstrates interactivity between other components of the system. The user enters location to be used as a search parameter for Twitter through a mobile application. Once the location has been obtained, it is then passed on to the Twitter Search API which returns the results of a location, labelled as either secure or insecure. When the user enters a location, the message search(location) is passed from the mobile application to the Twitter Search API. A message classifylocation() is used to classify the retrieved tweets after they undergo a cleaning process. The cleaning message is cleantweets(). locationresult() is then used to display the end result of the classification.

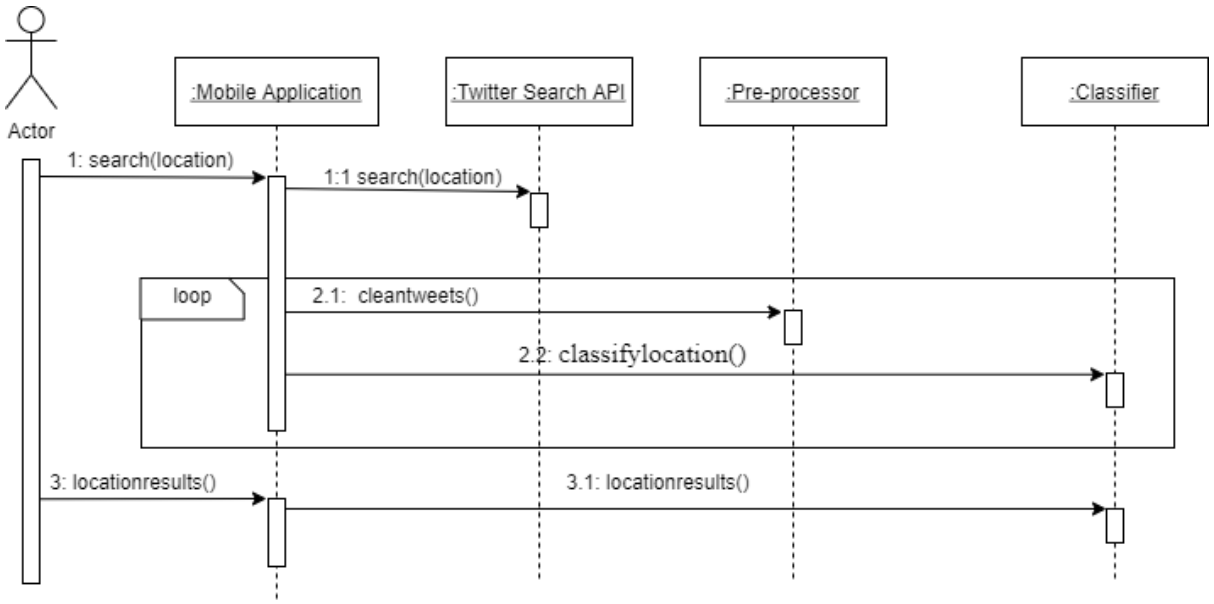


Figure 4.3 Sequence Diagram

4.4.3 Context Diagram

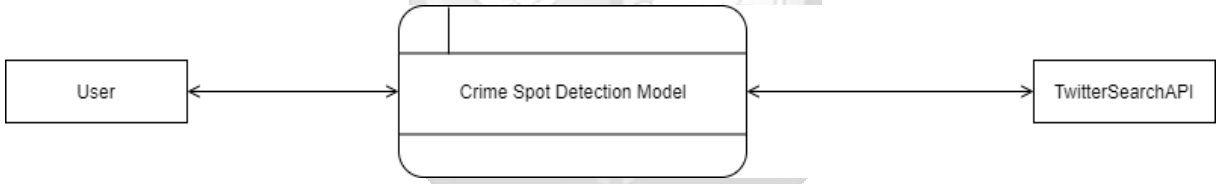


Figure 4.4 Context Diagram

4.4.4 Data Flow Diagram

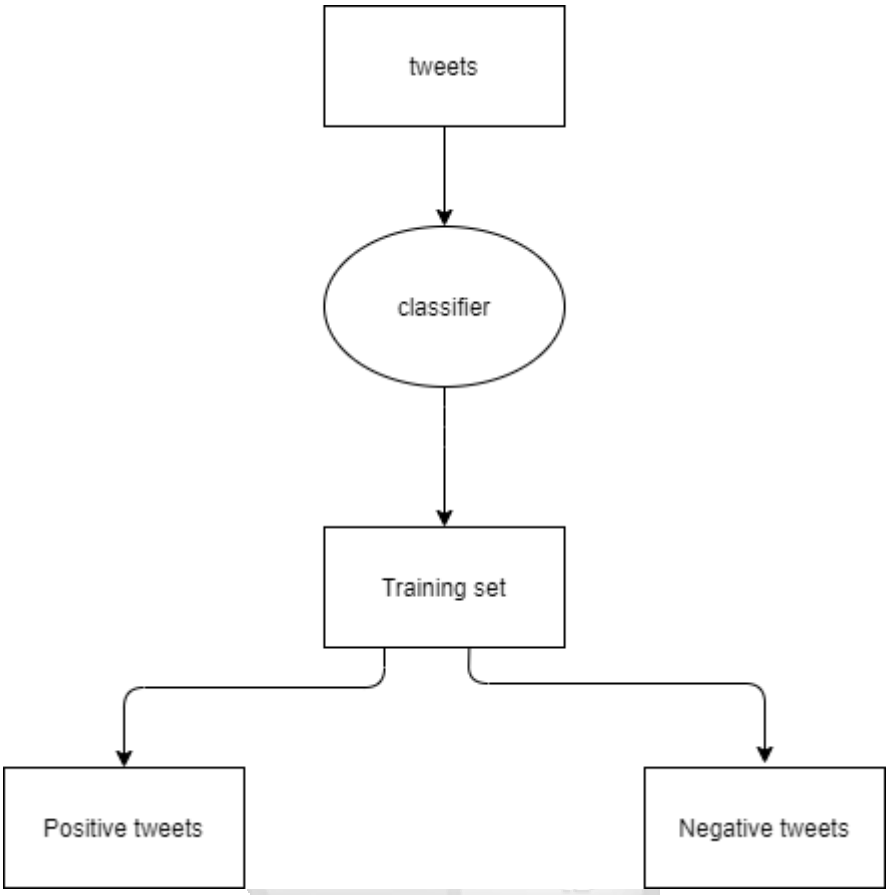


Figure 4.5 Data Flow Diagram

Chapter 5: System Implementation and Testing

5.1 Introduction

This chapter outlines how the system was implemented, tested and validated. It starts by marking out the process of building a crime hotspot detection corpus for machine learning. The pre-processing process, how the model is trained and tested against a test dataset.

5.2 Building the corpus

Twitter API, which is designed to help developers analyze conversations happening on Twitter was used to retrieve tweets including their metadata such as geo-location and timestamp.
The tweets were retrieved from sample users who had tweeted content related to crime in Nairobi. Below is the sample python code snippet used by the researcher to retrieve tweets using the Twitter API.

```
from tweepy import *

import pandas as pd
import csv
import os
import re
import string
import datetime
import preprocessor as p
from twitter.credentials import Credentials
from twitter.cleaner import Cleaner


today=datetime.date.today()
path="dataset/"+str(today)

credentials=Credentials()
cleaner=Cleaner()

auth = tweepy.OAuthHandler(consumer_key, consumer_secret)
auth.set_access_token(access_key, access_secret)

api = tweepy.API(auth,wait_on_rate_limit=True)

csvFile = open('tweets', 'a')
csvWriter = csv.writer(csvFile)

search_words = "# crime,murder,theft,hijacking,shoplifting,vandalism,carjacking,assault,arson,abduction"
new_search = search_words + " -filter:retweets"

for tweet in tweepy.Cursor(api.search,q=new_search,count=100,
```

Figure 5.1 Sample code for retrieving tweets

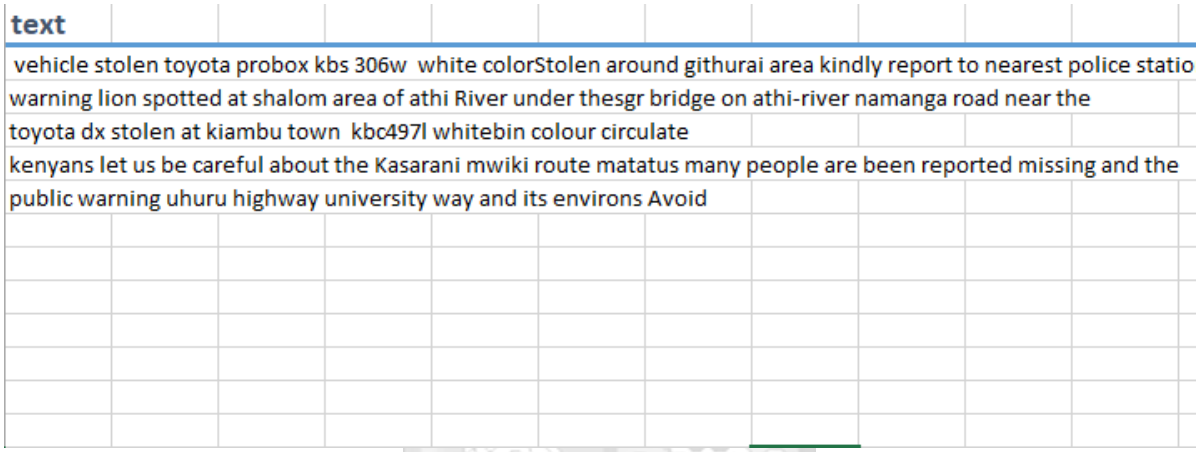
5.3 Preprocessing

Data collected using the Twitter API was found to be unstructured. Unstructured data cannot be processed or analysed by traditional data science techniques, especially machine learning.

This therefore made it paramount to first preprocess that data before training it. Figure 5.2 below shows a snippet of the raw data fetched using Twitter API.

Figure 5.2 Raw JSON data from Twitter search API

For the purposes of building a corpus that fits this study, username, datetime, retweets, text and geolocation was concetrated on. During preprocessing, tweets were tabulated into a structure that contained username, datetime, retweets, text and geolocation. Figure 5.3 below shows a sample of preprocessed tweets.



text
vehicle stolen toyota probox kbs 306w white colorStolen around githurai area kindly report to nearest police statio
warning lion spotted at shalom area of athi River under thesgr bridge on athi-river namanga road near the
toyota dx stolen at kiambu town kbc497l whitebin colour circulate
kenyans let us be careful about the Kasarani mwiki route matatus many people are been reported missing and the
public warning uhuru highway university way and its environs Avoid

Figure 5.3 Sample preprocessed tweets

Data collected had to be cleaned. There were phrases that were not needed in the interest of this study. These phrases included hashtags, retweets and external URLs. With the use of regular expressions, patterns that matched the unwanted phrases were stripped out. Figure 5.4 below shows a snippet of the code that was used to clean out the unwanted terms.

```
import re
import pandas as pd
import numpy as np
import string
import nltk
from nltk.stem.porter import *
import warnings
from datetime import datetime as dt

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

tweets = pd.read_csv(r'tweets.csv')
df = pd.DataFrame(tweets, columns = ['date','text'])

df['date'] = pd.to_datetime(df['date']).dt.date #changing date to datetime format from time-series

#removing pattern from tweets

def remove_pattern(input_txt, pattern):
    r = re.findall(pattern, input_txt)
    for i in r:
        input_txt = re.sub(i, '', input_txt)
    return input_txt

# remove twitter handles (@user)
tweets['clean_tweet'] = np.vectorize(remove_pattern)(tweets['text'], "@[\w]*")
#remove urls
tweets['clean_tweet'] = np.vectorize(remove_pattern)(tweets['text'], "https?://[A-Za-z./]*")

## remove special characters, numbers, punctuations
tweets['clean_tweet'] = tweets['clean_tweet'].str.replace("[^a-zA-Z#]", " ")
#
tweets['clean_tweet'] = tweets['clean_tweet'].apply(lambda x: ' '.join([w for w in x.split() if len(w)>2]))
```

Figure 5.4 Sample code for cleaning tweets

Additionally, conversion of all tweets to lower case and the removal of punctuation marks was done. Figure 5.5 below shows a sample of the clean tweets.

text
vehicle stolen toyota probox kbs 306w white colorStolen around githurai area kindly report to nearest police statio
warning lion spotted at shalom area of athi River under thesgr bridge on athi-river namanga road near the
toyota dx stolen at kiambu town kbc497l whitebin colour circulate
kenyans let us be careful about the Kasarani mwiki route matatus many people are been reported missing and the
public warning uhuru highway university way and its environs Avoid

Figure 5.5 Sample clean tweets

The tweets were grouped into three main categories. Negative, Positve and Neutral. The table below shows how the tweets were labelled.

Table 5.1 Labelling Of Tweets

Negative	A location is not a crime hotspot
Positive	A location is a crime hotspot
Neutral	Algorithm could not determine whether a location was negative or positive

In order to carry out supervised machine learning, the tweets were labelled as 1 for crime hotspot, 0 for secure locations and -1 for neutral locations. Figure 5.6 below shows the labelled tweets.

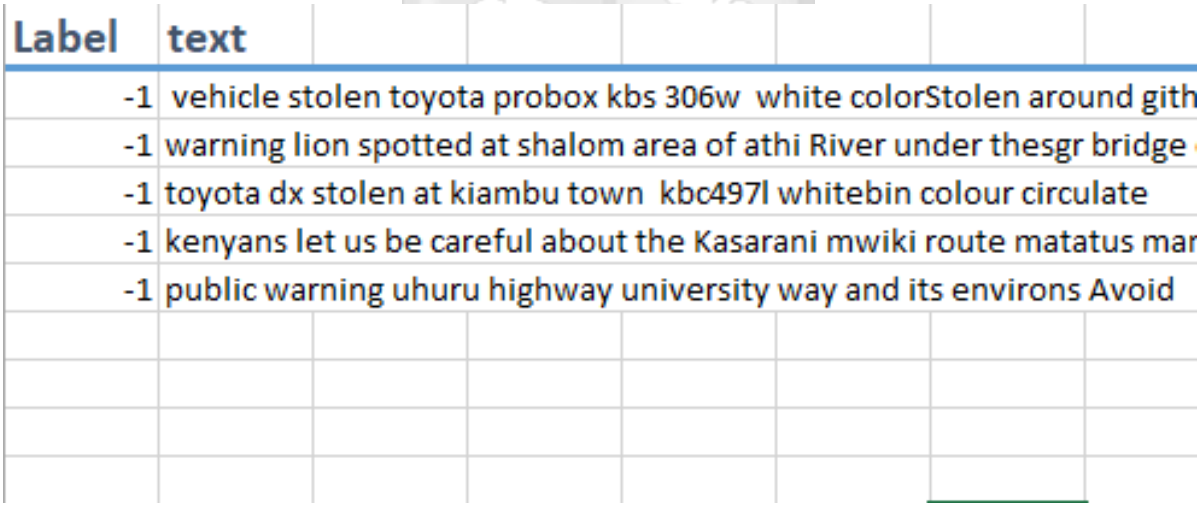


Figure 5.6 Labelled tweets

5.4 Determining positive and negative tweets

This study fetched tweets that were associated with crime and mapped their geolocation as well. As much as this technique fetched a reasonable large dataset for analysis, it contained quite a huge chunk of tweets that could not be categorized as either positive or negative. Some tweets were misrepresented as users did not intent to mean that there was the presence of crime in an area.

To further filter only negative tweets or positive ones for analysis, VADAR (Valence Aware Dictonary and sEntiment Reasoner), a lexicon and rule-based sentiment analysis tool which is specifically attuned to sentiments expressed in social media was used. VADER is fully open sourced and is issued under MIT License. VADER was the prefered tool because it does not require any training data, it deeply understands a text's sentiment even if the text contains capital letters, punctuations, conjuctions or even slang. VADER (Venkateswarlu Bonta, n.d., 2019) has an accuracy of 77%, higher than other other lexicon sentiment analysis tools such as Textblob which is 74% and NLTK which is 62%. The table below depicts the performance of VADER compared to other lexicon sentiment analysis tools.

Table 5.2 Lexicon-Classification Performance

A. Lexicon-Classification Performance

TABLE III PERFORMANCE OF LEXICON SENTIMENT ANALYSIS TOOLS

Lexicon	Classification Accuracy metrics			
	Precision%	Recall%	F1 score%	Accuracy%
VADER	78.46	85.0	81.60	77.0
Textblob	76.92	81.96	79.37	74.0
NLTK	60	55.0	57	62.0

The code snippet below shows how VADER was used in this study. For example, the text "Be on the lookout, cctv captures man getting robbed in Kilimani" scores a positive value of 1.

```
# function to print sentiments
# of the sentence.
def sentiment_scores(sentence):

    # Create a SentimentIntensityAnalyzer object.
    sid_obj = SentimentIntensityAnalyzer()

    # polarity_scores method of SentimentIntensityAnalyzer
    # oject gives a sentiment dictionary.
    # which contains pos, neg, neu, and compound scores.
    sentiment_dict = sid_obj.polarity_scores(sentence)

    print("Overall sentiment dictionary is : ", sentiment_dict)
    print("sentence was rated as ", sentiment_dict['neg']*100, "% Negative")
    print("sentence was rated as ", sentiment_dict['neu']*100, "% Neutral")
    print("sentence was rated as ", sentiment_dict['pos']*100, "% Positive")

    print("Sentence Overall Rated As", end = " ")

    # decide sentiment as positive, negative and neutral
    if sentiment_dict['compound'] >= 0.05 :
        print("Positive")

    elif sentiment_dict['compound'] <= - 0.05 :
        print("Negative")

    else :
        print("Neutral")
```

Figure 5.7 VADAR Sentiment Analyser code snippet

5.5 Identifying a Location's General Score

To calculate the general score of a certain location, the total number of positive tweets was divided by the total number of negative tweets and a location will either be scored as a hot spot or secure.

$$\text{General Score} = \frac{\text{Positive Tweets}}{\text{Negative Tweets}}$$

Equation 5.1 General Location Score

5.6 Training the SVM model

Support Vector Machine otherwise known as the SVM was first proposed by Vapnik and has swiftly gained traction since. It has lured quite a high level of interest in the machine learning world. A number of recent studies have reported that Support Vector Machines by and large are capable of delivering higher classification accuracy compared to other data classification algorithms. They have been utilized in a wide range of real world problems such as , hand-written digit recognition, text categorization, image classification and object detection, tone recognition, micro-array gene expression data analysis and data classification. Various studies have also shown that SVM is consistently superior to other supervised learning methods (Srivastava & Bhambhu, 2010).

The main intention of the support vector machine is to establish a hyperplane in an N-dimensional space (N- the number of features) that clearly classifies the data points.

Figure 5.8 below depicts how there are multiple hyperplanes that could be selected to separate the two classes of data points.

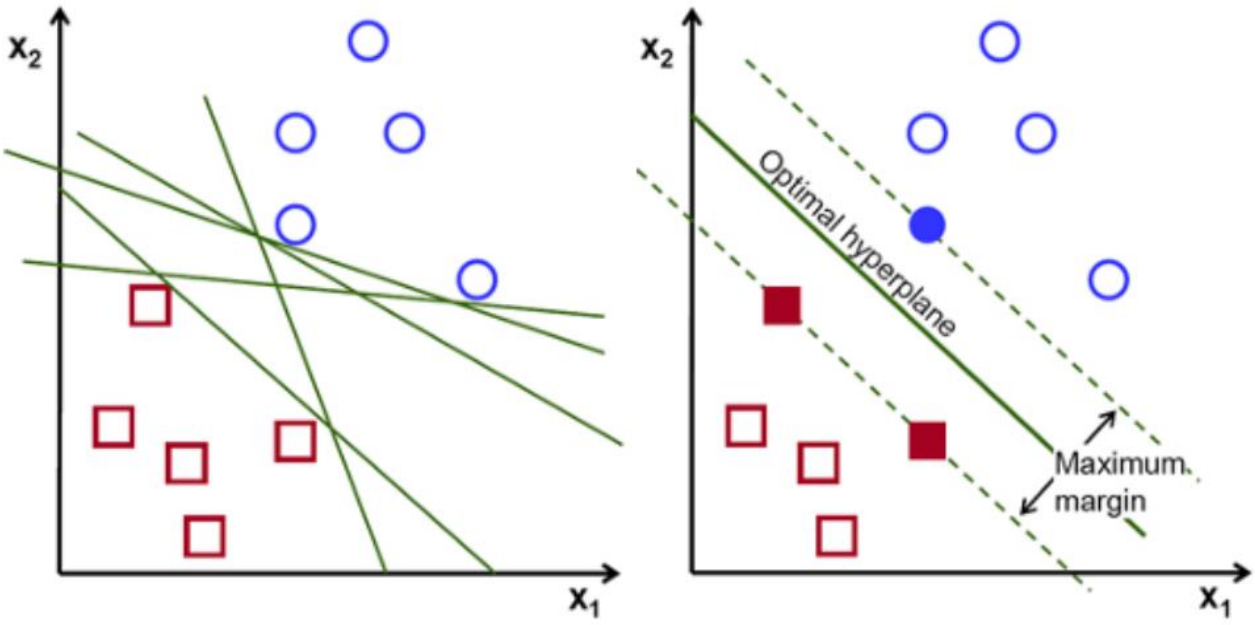


Figure 5.8 Possible Hyperplanes

After preprocessing and analysing the tweets, the training of the model then took place. Before the training of the model, a Comma Seperated Value (CSV) file that had the preprocessed and labelled tweets was fed into **pandas**. Pandas is a Python library of rich data structures and tools that is used for working with structured data sets. Pandas is widely used in vast fields including social sciences, finance and statistics. This Python library issues integrated, intuitive routines for performing multiple data sets analysis and manipulations. Pandas is set to be the foundational layer for the future of statistical computing in Python. It acts as an important addition to the already existing scientific Python stack while implementing and improving upon the kinds of data manipulation tools found in other statistical programming languages, for example R (McKinney, n.d., 2011).

Python also contains another vital library; Scikit-learn. Scikit-learn is a machine learning package in Python that is generally employed in data science (Hao and Ho, 2019). It contains implementations of a comprehensive list of machine learning methods under unified data and modelling procedure conventions. This in turn makes it a convenient toolkit for behaviour and educational scientist. Generally plain text cannot be analysed by machine learning algorithms. Scikit-learn is used to remove punctuation marks and perform feature extraction.

This study implemented Scikit-learn. CountVectorizer, a module in Scikit-learn library was used to convert the training data into a matrix that contained token counts. TfIdfTransformer transformed the counts using tf-idf weighting. Vectorization, transformation and specification of the classifier into one pipeline was handled by Scikit-learn. Figure 5.9 below shows how SVM was implemented. The function inputs data D with label x, it then learns and returns the SVM model.

```
def createSVM(D,x):
    svm_clf=Pipeline([('vect',CountVectorizer()),('tfidf',TfidfTransformer()),
    ('svm',SVC(kernel="linear",C=1))])
    svm_clf=svm_clf.fit(D,x)
    return svm_clf
```

Figure 5.9 Implementation of SVM Model

5.7 Testing the model

The training dataset which was 30 percent of the labelled tweets was used to train the model. The training data set was run via the learnt model for prediction. A confusion matrix was used to examine the performance of the model.

Table 5.3 below displays the output of the confusion matrix.

Table 5.3 Output from the confusion matrix

	Actual -1	Actual 1
Predicted -1	2800	431
Predicted 1	388	3100

The values for the true negatives, true positives, false negatives and false positives were derived from the confusion matrix. Table 5.4 below shows the values from the confusion matrix.

Table 5.4 Values from the confusion matrix

True Positives (TP)	3100

False Positives (FN)	431
True Negatives (TN)	2800
False Negatives (FN)	388

The metrics' recall, precision and F-measure was calculated. The results were as follows.

5.7.1 Precision

Precision is calculated as the number of true positives divided by the total number of true positives plus false positives.

$$\text{Precision} = \frac{\text{TruePositives}}{(\text{TruePositives} + \text{FalsePositives})}$$

Equation 5.2 Precision

$$\text{Therefore, Precision} = \frac{3100}{(3100+431)} = 0.88$$

Precision can also be calculated in Python using the precision_score() scikit-learn function.

5.7.2 Recall

Rerecall is calculated as the number of true positives divided by the total number of true positives plus false negatives.

$$\text{Recall} = \frac{\text{TruePositives}}{(\text{TruePositives} + \text{FalseNegatives})}$$

Equation 5.3 Recall

$$\text{Therefore, Recall} = \frac{3100}{(3100+388)} = 0.89$$

In Python recall score can be calculated using the recall_score() scikit-learn function.

5.7.3 F-measure

F-Measure provides a way to combine both precision and recall into a single measure that captures both properties.

F-measure is calculated using equation 4 below.

$$\text{F-Measure} = \frac{(2 * \text{Precision} * \text{Recall})}{(\text{Precision} + \text{Recall})}$$

Equation 2.4 F-measure

Therefore, F-Measure = $\frac{(2*0.88*0.89)}{(0.88+0.89)}$

$= \frac{1.57}{1.77}$

$= 0.89$

The F-measure score can be calculated using the f1_score() scikit-learn function in Python.

5.8 ROC Curve for the SVM model

A Receiver Operating Characteristic (ROC) curve which shows the SVM Classifier performance was plotted. Figure 5.10 below shows how the SVM classifier performed.

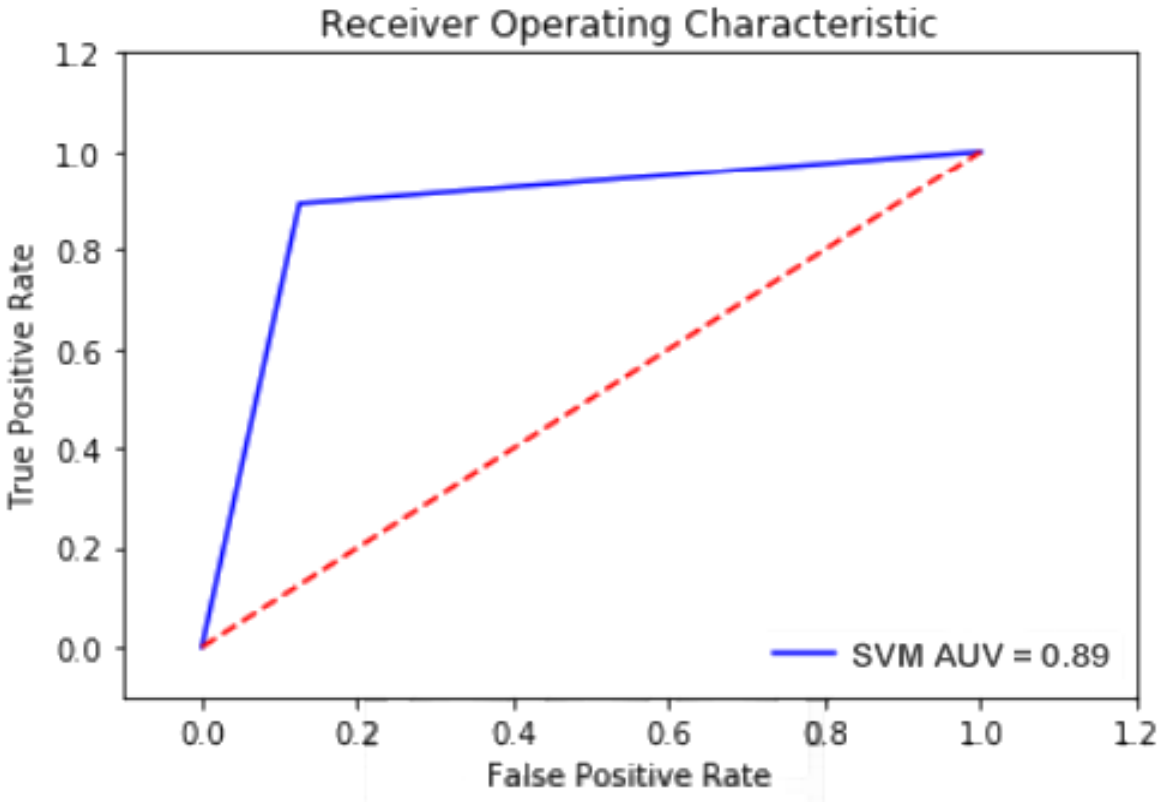


Figure 5.10 ROC Curve for the SVM Classifier

5.9 Using the model to detect crime hotspots

For the model to detect a crime hotspot, the user has to choose a location. The location will be used to extract tweets through the Twitter Search API. The tweets will then be passed through the built model to do a crime hotspot detection.

Figure 5.11 shows the home screen that welcomes the user. The user is prompted to choose a location.

Figure 5.11 Usalama home screen

The chosen location will then provide a key phrase which the model will use to fetch the tweets through the Twitter search API. The fetched tweets will then undergo preprocessing. the fetched tweets will be tabulated into a structure that contained username, datetime, retweets, text and geolocation. Unwanted phrases will then be cleaned. These phrases include hashtags,

retweets and external URLs. The tweets will all be converted to lower case and all the punction marks will be removed as well.

Using the build model, the tweets will then be categorized into negative, positve or neutral. Based on the number of positive or negative tweets, a location will then be labelled as either safe or a crime hot spot.

Chapter 6 : Discussions

This chapter examines the results of the study in respect to the set objectives. The main purpose of this research was to develop a prediction model that detects crime hotspots in Nairobi from Twitter. An SVM model was developed using unigram and tf-idf weighting. The SVM model's performance was tested using the test data set. In order to certify the researcher's procedure in detecting crime hotspots in Nairobi using Twitter, several experiments were done using different classifiers. Based on the experiment, the indication was that SVM achieved the best performance in analysing sentiments.

6.1 Sentiment analysis experiments

6.1.1 Using different classifiers

The main objective of performing this tests was to do a comparison between the performance of the SVM model compared to Naive Bayes, K-nearest neighbour and Random forest machine learning algorithms.

Table 6.1 below shows the results achieved by Naive Bayes.

Table 6.1 Naive Bayes Results

	Precision	Recall	F-Score
0	0.85	0.89	0.86
1	0.87	0.82	0.85
avg/total	0.86	0.86	0.86

A Receiver Operating Characteristic (ROC) curve which shows the Naïve Bayes Classifier performance was plotted. Figure 6.1 below shows how the Naïve Bayes classifier performed.

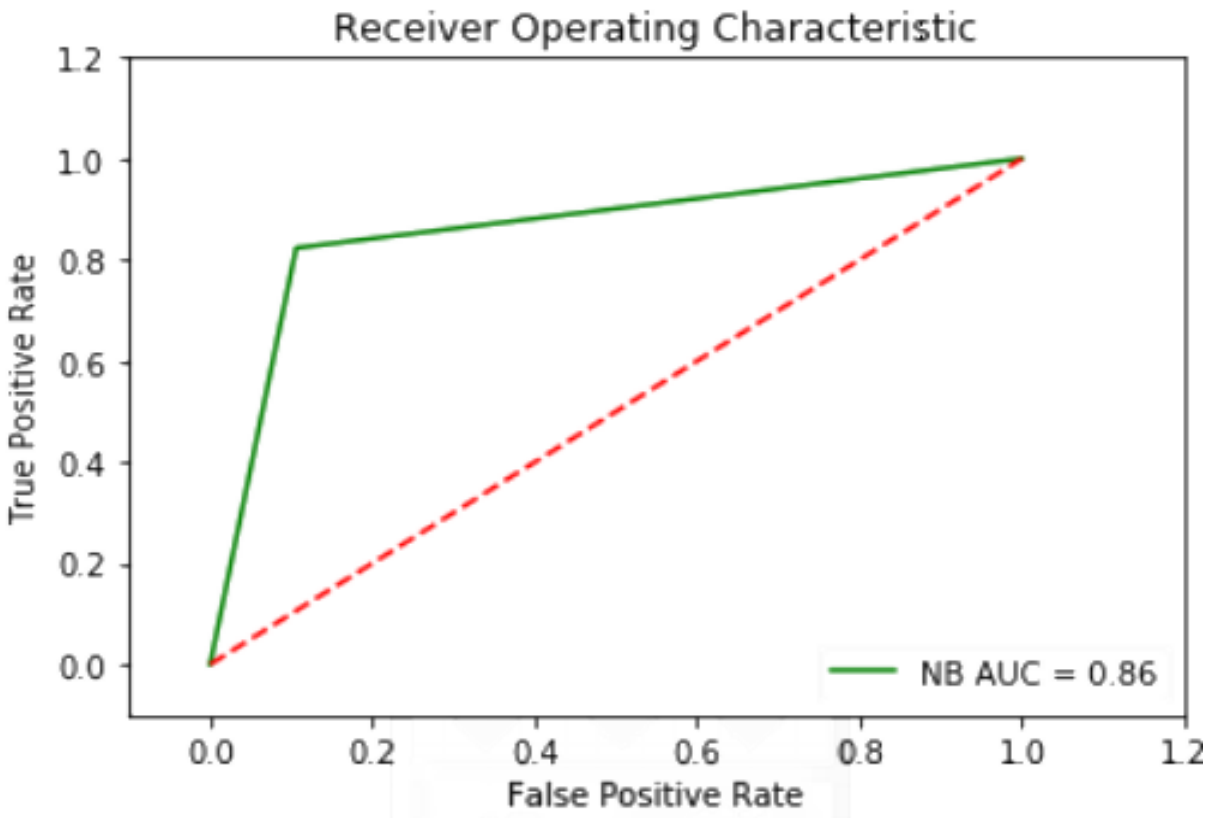


Figure 6.1 ROC Curve for the Naive Bayes Classifier

Table 6.2 below shows the results achieved by Random Forest.

Table 6.2 Random Forest Results

	Precision	Recall	F-Score
0	0.72	0.83	0.77
1	0.79	0.66	0.72
avg/total	0.75	0.75	0.75

A Receiver Operating Characteristic (ROC) curve which shows the Random Forest Classifier performance was plotted. Figure 6.3 below shows how the Random Forest classifier performed.

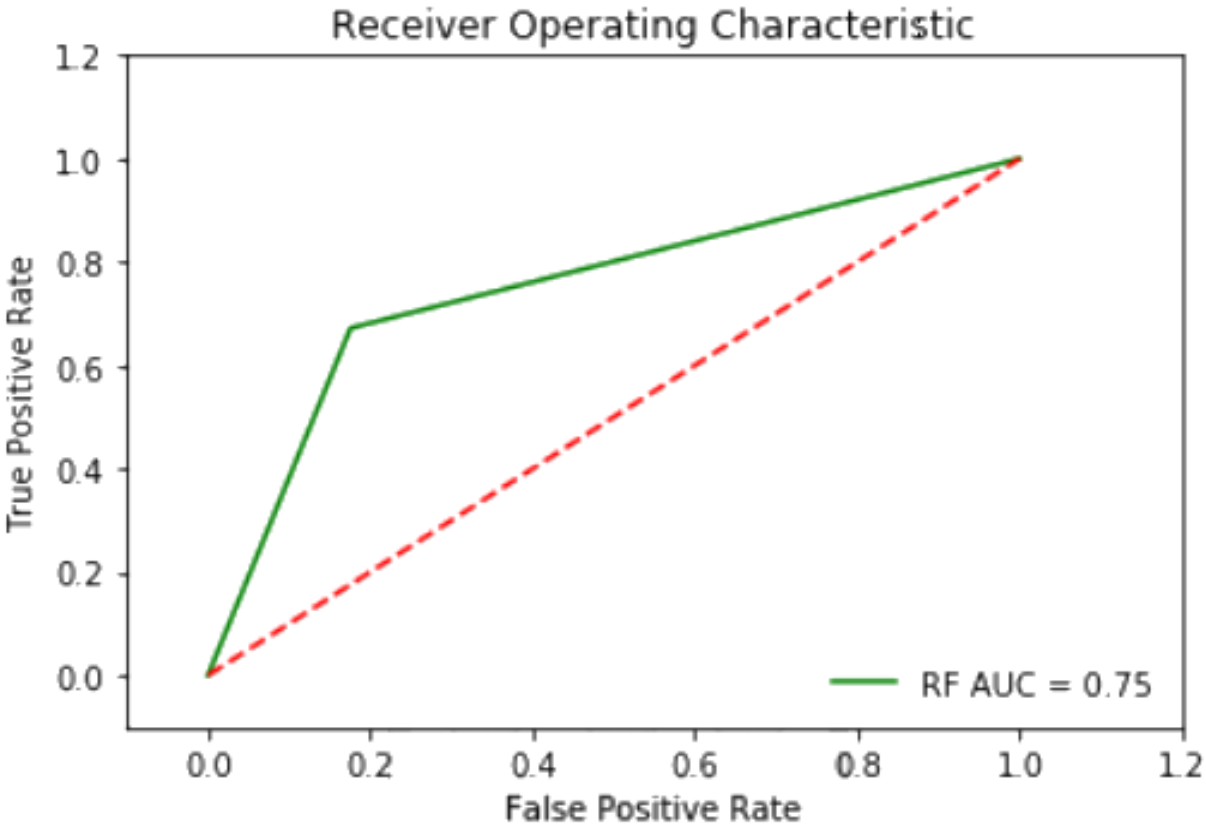


Figure 6.3 ROC Curve for the Random Forests Classifier

Table 6.3 below shows the results achieved by K-nearest neighbour.

Table 6.3 K-nearest neighbour Results

	Precision	Recall	F-Score
0	0.74	0.76	0.75
1	0.74	0.73	0.74
avg/total	0.74	0.74	0.74

A Receiver Operating Characteristic (ROC) curve which shows the K-nearest neighbour Classifier performance was plotted. Figure 6.4 below shows how the K-nearest neighbour classifier performed.

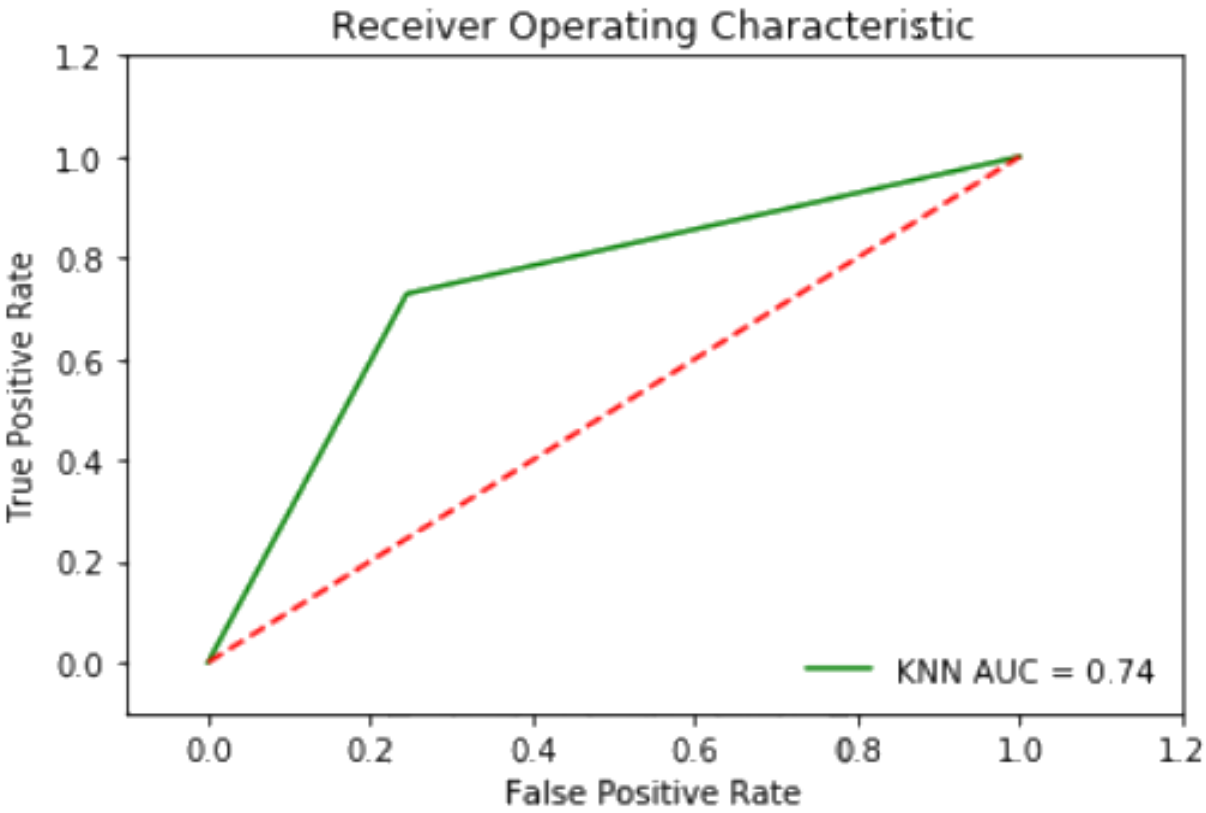


Figure 6.4 ROC Curve for the KNN Classifier

As depicted in table 6.4 below, it is clear that the linear SVM model achieved better results compared to the other classifiers (Naive Bayes, K-Nearest Neighbour and Random Forest).

Table 6.4 Performance of Different Classifiers

	Accuracy	Precision	Recall	F-Score
Linea SVM	0.88	0.88	0.88	0.88
K-Nearest neighbour	0.74	0.74	0.74	0.74
Naïve Bayes	0.86	0.86	0.86	0.86
Random Forest	0.75	0.75	0.75	0.75

6.1.2 Experiment 1: SVM with different feature types

The objective of this test was to establish the outcome of using different feature types and weighting schemes on the SVM model. Unigrams, bigrams and trigrams where the feature types considered in the experiment. The feature weighting schemes reviewed were tf-idf and basic count. Bigram feature gave the best performance when the classifier was using the bigram feature.

Table 6.5 below shows the results for the feature types.

Table 6.5 SVM Performance

	Accuracy	Precision	Recall	F-Score
Unigram	0.887	0.89	0.89	0.88
Bigram	0.894	0.89	0.89	0.74
Trigram	0.871	0.88	0.88	0.86

6.1.3 Experiment 2: K-nearest neighbour with different feature type

A similar experiment to experiment 1 was done to examine the performance of K-nearest neighbour classifier on the data set. Unigrams, bigrams and trigrams where the feature types considered in the experiment. The feature weighting schemes reviewed were tf-idf and basic count. Unigram and bigram features gave the best performance for the K-nearest neighbour classifier.

Table 6.6 below shows results for K-nearest neighbour's performance

Table 6.6 K-nearest neighbour's Performance

	Accuracy	Precision	Recall	F-Score
Unigram	0.74	0.74	0.74	0.74
Bigram	0.74	0.74	0.72	0.72
Trigram	0.73	0.73	0.70	0.70

6.1.4 Experiment 3: Random Forest with different feature type

Another experiment, similar to the one done to determine the performance of K-nearest neighbour classifier on the data set was done. This time, the performance of Random Forest was tested. Unigrams, bigrams and trigrams where the feature types considered in the experiment. The feature weighting schemes reviewed were tf-idf and basic count. Unigram and bigram features gave the best performance for the Random Forest classifier.

Table 6.7 below shows results for Random Forest' performance

Table 3.7 Random Forest Performance

	Accuracy	Precision	Recall	F-Score
Unigram	0.74	0.74	0.73	0.74
Bigram	0.73	0.73	0.73	0.72
Trigram	0.73	0.74	0.73	0.73

6.1.5 Experiment 4: Naive Bayes with different feature type

An experiment similar to the one done for Random Forest and K-nearest neighbour was conducted for Naïve Bayes. Unigrams, bigrams and trigrams where the feature types considered in the experiment. The feature weighting schemes reviewed were tf-idf and basic count. Unigram and bigram features gave the best performance for the Random Forest classifier.

Table 6.8 below shows results for Naïve Bayes' performance

Table 6.8 Naïve Bayes Performance

	Accuracy	Precision	Recall	F-Score
Unigram	0.873	0.85	0.88	0.85
Bigram	0.883	0.89	0.87	0.87
Trigram	0.872	0.88	0.88	0.88

6.2 Discussions

Based on the results outlined from the experiments, it was found that the best way to create a model for detecting crime hotspots using Twitter is the use of an SVM machine learning algorithm with bigram features weighted using tf-idf.

Chapter 7: Conclusion and Recommendation

7.1 Conclusion

This research purposed to develop a tool to detect crime hotspots in Nairobi using data from Twitter with the aid of machine learning techniques. In order to carry out the set objectives for this study, profound research on other similar studies was done. Literature review for different relevant studies was also done. Stakeholders were interviewed to establish the current challenge of the lack of a proper tool that both the public and the police could use to curb crime in Nairobi. Additional literature on several machine learning techniques was reviewed. This was done with the sole purpose of understanding the application of various machine learning techniques in the detection of crime hot spots and how these techniques could be used in this study.

With the use of the Twitter API, tweets related to crime in Nairobi were fetched based on location. These tweets were subjected through text preprocessing and were labelled as positive denoted as 1 or negative denoted as 0 or neutral denoted as -1. The large collected text data set was then divided into two. The training data set which was used to train the SVM model and the test data set which was use to determine the performance of the SVM model.

In order to ascertain whether the researcher's approach delivered the best results, a sequence of experiments was conducted. The SVM model was tested against other machine learning classifiers. The machine learning classifiers were Naive Bayes, Random Forest and K-nearest Neighbour. It was then established that the SVM model with bigram features performed with the highest degree of accuracy. It had an accuracy score of 88 percent. The built tool was then subjected to classify other similar tweets that were retrieved from Twitter through the use of the Twitter search API.

7.2 Recommendations

This study clearly depicted that the SVM model can be used to detect crime hotspots in Nairobi using Twitter as an addition to the current methods that the law enforcement agencies are using in Kenya. This would provide a tool that runs on mobile which will help both the public and the police in detecting crime hot spots which in turn will help reduce crime. The researcher acknowledges the fact that, better results would have been arrived at if a larger dataset had been used.

7.3 Further work

One tweet consists of 280 characters as per Twitter's limitation. This therefore affects users' choice of words when tweeting. Users are forced to shorten their words by the use of abbreviations and in some cases use informal language specific to Twitter. This therefore hinders the classification of tweets. Future research could lean towards cleaning of tweets before classification can be performed.

Crime in Nairobi can be reported on Twitter through various languages. These languages include English, Swahili and Sheng. This research only focused on tweets written in English and Swahili. This therefore means that it is necessary to develop a model that performs classifications for tweets that are written in other languages apart from English which is available in the lexical resources.

Non textual tweet content such as videos, images and emoticons that may contain crime information were not considered by this study. Images, videos and emoticons should be considered in upcoming studies.

References

Akinsola, J.E.T., 2017. Supervised Machine Learning Algorithms: Classification and Comparison. Int. J. Comput. Trends Technol. IJCTT 48, 128–138. https://doi.org/10.14445/22312803/IJCTT-V48P126

Artificial Intelligence Market Size & Share Report, 2020-2027 [WWW Document], n.d. URL https://www.grandviewresearch.com/industry-analysis/artificial-intelligence-ai-market (accessed 7.17.20).

Bolla, R., 2014. Crime pattern detection using online social media. Masters Theses.

Carleo, G., Cirac, I., Cranmer, K., Daudet, L., Schuld, M., Tishby, N., Vogt-Maranto, L., Zdeborová, L., 2019. Machine learning and the physical sciences. Rev. Mod. Phys. 91, 045002. https://doi.org/10.1103/RevModPhys.91.045002

Chen, X., Cho, Y., Jang, S.Y., 2015. Crime prediction using Twitter sentiment and weather, in: 2015 Systems and Information Engineering Design Symposium. Presented at the 2015 Systems and Information Engineering Design Symposium, pp. 63–68. https://doi.org/10.1109/SIEDS.2015.7117012

Chirawichitchai, N., 2013. Sentiment classification by a hybrid method of greedy search and multinomial naïve bayes algorithm, in: 2013 Eleventh International Conference on ICT and Knowledge Engineering. Presented at the 2013 Eleventh International Conference on ICT and Knowledge Engineering, pp. 1–4. https://doi.org/10.1109/ICTKE.2013.6756285

Dinisman, T., Moroz, A., 2017. Understanding victims of crime: The impact of the crime and support needs. https://doi.org/10.13140/RG.2.2.17335.73124

Effectiveness of the Nyumba Kumi community policing initiative in Kenya | Wangari Maathai Institute for Peace and Environmental Studies [WWW Document], n.d. URL https://wmi.uonbi.ac.ke/news/phosfluore (accessed 7.5.20).

Gibbs, T., 2020. Seeking economic cyber security: a Middle Eastern example. J. Money Laund. Control 23, 493–507. https://doi.org/10.1108/JMLC-09-2019-0076

Hao, J., Ho, T.K., 2019. Machine Learning Made Easy: A Review of Scikit-learn Package in Python Programming Language. J. Educ. Behav. Stat. 44, 348–361. https://doi.org/10.3102/1076998619832248

Li, G., Zhang, J., 2018. Music personalized recommendation system based on improved KNN algorithm, in: 2018 IEEE 3rd Advanced Information Technology, Electronic and Automation Control Conference (IAEAC). Presented at the 2018 IEEE 3rd Advanced Information Technology, Electronic and Automation Control Conference (IAEAC), pp. 777–781. https://doi.org/10.1109/IAEAC.2018.8577483

Mazzonello, V., Gaglio, S., Augello, A., Pilato, G., 2013. A Study on Classification Methods Applied to Sentiment Analysis. https://doi.org/10.1109/ICSC.2013.82

McKinney, W., n.d. pandas: a Foundational Python Library for Data Analysis and Statistics 9.

Muiya, B.M., 2014. The Nature, Challenges and Consequences of Urban Youth Unemployment: A Case of Nairobi City, Kenya. Univers. J. Educ. Res. 2, 495–503.

Musoi, K., Muthama, T., Kibor, J., Kitiku, J., 2014. A study of crime in urban slums in Kenya: the case of Kibra, Bondeni, Manyatta and Mishomoroni slums. Security Research and Information Centre (SRIC), Nairobi, Kenya.

Oduor, C., Acosta, F., Makhanu, E., 2014a. The adoption of mobile technology as a tool for situational crime prevention in Kenya, in: 2014 IST-Africa Conference Proceedings. Presented at the 2014 IST-Africa Conference Proceedings, pp. 1–7. https://doi.org/10.1109/ISTAFRICA.2014.6880669

Oduor, C., Acosta, F., Makhanu, E., 2014b. The adoption of mobile technology as a tool for situational crime prevention in Kenya, in: 2014 IST-Africa Conference Proceedings. Presented at the 2014 IST-Africa Conference Proceedings, pp. 1–7. https://doi.org/10.1109/ISTAFRICA.2014.6880669

Predicting Crime Using Twitter and Kernel Density Estimation | Request PDF [WWW Document], n.d. URL https://www.researchgate.net/publication/260314336_Predicting_Crime_Using_Twitter_and_Kernel_Density_Estimation (accessed 10.18.20).

ResearchGate Link, n.d.

Rodríguez-Pose, A., Cols, G., 2017. The determinants of foreign direct investment in sub-Saharan Africa: What role for governance? Reg. Sci. Policy Pract. 9, 63–81. https://doi.org/10.1111/rsp3.12093

Samoei, P.C., 2018. Role of Information Communication and Technology in Enhancing Security in Urban Areas in Kenya: A Literature Based Review. J. Inf. Technol. 2, 17–27.

Sileyew, K.J., 2019. Research Design and Methodology. Cyberspace. https://doi.org/10.5772/intechopen.85731

Srivastava, D., Bhambhu, L., 2010. Data classification using support vector machine. J. Theor. Appl. Inf. Technol. 12, 1–7.

University of Salzburg, 2015. Spatiotemporal Interaction of Urban Crime in Nairobi, Kenya, in: GI_Forum 2014 – Geospatial Innovation for Society. Presented at the GI_Forum 2014 - Geospatial Innovation for Society, Austrian Academy of Sciences Press, Salzburg, pp. 175–186. https://doi.org/10.1553/giscience2014s175