National College of Ireland

Crime Incidents Classification Using Supervised Machine Learning Techniques: Chicago

MSc Research Project
Data Analytics

Nelson Omonigho Edoka
X18172342

School of Computing
National College of Ireland

Supervisor: Dr. Catherine Mulwa

National College of Ireland

MSc Project Submission Sheet

School of Computing

Student Name: NELSON OMONIGHO EDOKA

Student ID: X18172342

Programme: Msc Data Analytic **Year:** 2020

Module: Research Project

Supervisor: Dr Catherine Mulwa

Submission Due Date: 23rd April 2020

Project Title: Crime Incidents Classification Using Supervised Machine Learning Techniques: Chicago

Word Count: 9169 **Page Count** 27

I hereby certify that the information contained in this (my submission) is information pertaining to research I conducted for this project. All information other than my own contribution will be fully referenced and listed in the relevant bibliography section at the rear of the project.
ALL internet material must be referenced in the bibliography section. Students are required to use the Referencing Standard specified in the report template. To use other author's written or electronic work is illegal (plagiarism) and may result in disciplinary action.

Signature:

Date: 23rd April 2020

PLEASE READ THE FOLLOWING INSTRUCTIONS AND CHECKLIST

Attach a completed copy of this sheet to each project (including multiple copies)	□
Attach a Moodle submission receipt of the online project submission, to each project (including multiple copies).	□
You must ensure that you retain a HARD COPY of the project, both for your own reference and in case a project is lost or mislaid. It is not sufficient to keep a copy on computer.	□

Assignments that are submitted to the Programme Coordinator Office must be placed into the assignment box located outside the office.

Office Use Only	
Signature:	
Date:	
Penalty Applied (if applicable):	

Crime Incidents Classification Using Supervised Machine Learning Techniques: Chicago

Nelson Omonigho Edoka
X18172342

Abstract

Crime is a difficult issue faced by most nations on the planet today. The powerlessness to control crime has prompted genuine drop down in the economy of the nation, loss of lives and property. The growing need to mitigate crimes gave rise to this research work by applying data mining techniques from the data obtained from Chicago crime porter to break down the different crimes and build up a model that was able to classify these crimes, looking at the different models as far as execution to check how well the crimes were classified, and in return help the government and law enforcement agencies get an insight of the most common type of crimes they come across daily and enable them to take careful steps to overcome this criminal activities. Logistic Regression, K Nearest Neighbors, Naïve Bayes, Decision Tree Classifier and XGBoost were the five supervised classification machine learning techniques used to handle this issue. Resampling techniques was then applied on the crime data to deal with the problem of imbalanced data. The outcomes obtained from the developed models indicated that Decision Tree Classifier, and XGBoost acquired an accuracy of 99.6% and 97.3% respectively, Logistic Regression, K Nearest Neighbors, and Naïve Bayes acquired an accuracy of 39.8%, 81.2%, and 67.5% in classifying crime incidents in Chicago.

1 Introduction

Safe guiding the lives and properties of citizens is a major responsibility of the Government in every country. Crime is an unlawful act punishable by law in a country. A country needs to have a record of low crime as it attracts investors or tourists which in turn boosts the economy of that country. The causes of Crime(s) are numerous, which makes it more difficult and complex to identify and it is the sole responsibility of the government and top security agencies to reduce the level of crime in each country and to prevent crime before it happens. Some of the causes of crimes include victims of unfair rulings and the correctional system, drugs, depression and other social and mental disorders, family condition, regionalism, Tv violence, racism, politics, poverty and overpopulation.

1.1 Research Background and Motivation

Identifying types of crimes and locations where the crime occurs has become a major problem and as such putting the lives of everyone at risk. Data mining has proven to be the most powerful tool not only in the field of IT (Information Technology) but across other fields like Life Sciences, Physical Sciences, Social Sciences, Business, Game, Engineering,

Security, etc. Data mining makes it possible to extract meaningful information from a crime dataset that will help identify various crimes in different locations. Data mining is also applied in predicting future possibilities of crime using current information Prabhjot and Kirti (2019)

Machine learning algorithms was used to develop a classification model that trained on historical data to analyse the crime rate. Linear Regression, KNN, Naïve Bayes, Random Forest, Support Vector Machine, etc. are some of the classification machine learning algorithms that have been deployed by various researchers over the years in the classification of crimes using historical dataset. One of the challenges faced by machine learning in the field of crime analysis is in the case of accuracy Kim et al. (2017). The result showed that their model had a low prediction rate that resulted in the model not performing very well as expected. So, for that reason, Extreme Gradient Boosting also known as XGBoost has been adopted in this project to further improve the performance of the model. Also, a technique called Resampling was implemented in the pre-processing stage to deal with the problem of imbalanced data.

1.2 Research Question

This research project focuses on the reduction of crime incidents using historical data obtained from the Chicago data porter to improve the performance of analyzing and classifying various crime types by using supervised machine learning models which include Logistic Regression, K Nearest Neighbors, Naïve Bayes, Decision Tree Classifier, and XGBoost and will therefore help the government and law enforcement agencies in taking preventive measures to mitigate crime. This reason gave rise to the research question.

RQ: *"To what extent can the supervised machine learning techniques (Logistic Regression, K Nearest Neighbors, Naïve Bayes, Decision Tree Classifier, and XGBoos) be enhanced to assist the government and law enforcement agencies in preventing crime occurrences in the city of Chicago?* ". The performance of the models was evaluated using evaluation metrics (accuracy, precision, recall, and f1 score).

1.3 Research Objectives and Contributions

For the project aim to be accomplished and for the research question to be answered, the following objectives were taken into serious consideration. The first objective was to critically review works done by previous researchers in handling crime, and the machine learning algorithms that were implemented, as it will play a vital role in the implementation phase. This gave rise to the second objective which involves data pre-processing on the dataset. The third objective involves the implementation of the chosen supervised learning techniques. Objective four involves comparing the performance of the implemented techniques. A comparison of the developed models with existing ones in the state of art gave rise to the fifth objective. Table 1 below gives a brief description of the set objectives required for this research work.

Table 1. Research objectives

Objectives	Description	Techniques	Evaluation Metrics
1	Critically reviewing previous work done on crime analysis by other researchers and the existing techniques (2006-2019)		
2 **2.1** **2.2** **2.3** **2.4**	Collection of crime incidents data Data pre-processing on the crime data. Carrying out exploratory analysis on the crime dataset. Performing of feature engineering and data modelling on the crime data Implementation of resampling technique on the crime data to deal with imbalance data		
3 **3.1** **3.2** **3.3** **3.4** **3.5** **3.6**	Implementation of machine learning model on the dataset Implementation and evaluation of Logistic Regression using crime data Implementation and evaluation of K Nearest Neighbour (KNN) using crime data Implementation and evaluation of Naïve Bayes using crime data Implementation and evaluation of Decision Tree Classifier using crime data Implementation and evaluation of Extreme Gradient Boosting (XGBoost) using crime data. Implementation and evaluation of the same model using fewer variables.	Logistic Regression, K Nearest Neighors, Naïve Bayes, Decision Tree and XGBoost	Accuracy, Precision, Recall and F1 Score
4	Performance comparison of the developed models (objective 3)		

5	Comparison of developed models with existing ones in the state of arts (objective 4) with existing ones		

Major Contribution: The major contribution obtained from this research work is the classification of crime incidents models with powerful evaluation metrics which will help the body of government and law enforcement agencies in identifying crime types, taking down preventive measures in dealing with the crime to ensure the safety of citizens living in that location, and will also play a vital role in the field of security.

Minor Contribution: The minor contribution of this work is the visualization of results obtained from the explanatory analysis by using plots, identification of research gaps on crime analysis by reviewing works of other researchers, and implementation of the said model using the rightful tools.

The remainder of the research work is organized as follows: Chapter two (2) reviews related works of literature, Chapter three (3) describes the crime incidents methodology that was deployed in this research, Chapter four (4) describes the implementation, evaluation, and result obtained from the classification models, section five (5) describes the discussion of the model, and the research work is then concluded with section six (6)

2 State of Art Review of Crime Analysis (2006-2019)

2.1 Introduction

Crime is an important issue that must be addressed before it becomes difficult to handle. There is no doubt that Chicago is among the major cities in the united states with a record of the high crime rate. Before now, researchers have contributed significantly in analysing crime classification. Some of these research works are discussed in this section. Some of the analysis includes a review of the machine learning model, review of data mining open-source software (weka), and sentiment analysis in the classification of crimes.

2.2 A Critical Review of Waikato Environment for Knowledge Analysis on Crime

Waikato Environment for Knowledge Analysis (WEKA) is an open-source machine learning software that was adopted by some researchers for analyzing various crimes. Lawrence and Natarajan (2015) focused more on the comparison between violent crime patterns from communities and crime unnormalized dataset that was obtained from the University of California-Irvine data repository and the actual statistical crime data for Mississippi. In their work, WEKA was adopted i.e. an open-source data mining software. Linear Regression, Addictive Regression, and Decision Stump algorithms were implemented to see which best produce the best result in predicting violent crime patterns. The result they got from their analysis shows Linear Regression to be very effective and accurate in predicting violent crime amongst the three algorithms that were proposed for the project, although the project

was just limited to few features like murder, rape, robbery, and assault. Emmanuel et al. (2017) performed an analysis of Business Intelligence (BI) Techniques on crime prediction. Their analysis focuses more on identifying the most accurate and effective BI that can be used in providing accurate results in crime data mining. Decision tree (J48), Naïve Bayes, Multilayer Perceptron, and Support Vector Machine were the four-classification algorithms that were proposed for the research work, and their results were compared to find which perform better for crime prediction. The classification models were generated by using an open-source data mining software called WEKA (Waikato Environment for Knowledge Analysis). The result obtained shows that the Decision Tree and Multilayer Perceptron had 100%, Naïve Bayes and Support Vector Machine had 89.7989% and 92.6724% respectively. Although both Decision Tree and Multilayer Perceptron 100% accuracy, Decision Tree had better execution time with 0.06sec why Multilayer Perceptron, SVM, Naïve Bayes had 9.26sec, 0.66sec, and 0.14sec respectively when ran on windows7 32bit. Rizwan et al. (2013) did a comparison between two classification model i.e. Naïve Bayesian and Decision Tree for crime prediction across the different states in the US using WEKA. The result obtained from their analysis showed that Decision Tree performed better than Naïve Bayesian with both having an accuracy of 83% and 70% respectively. Danison et al. (2017) used a classification algorithm (decision tree) to predict crime which has proven to be one of the machine learning algorithms for crime prediction. The model predicted crime with an accuracy of 94% which was developed in WEKA.

2.3 A Critical Review on Machine Learning Algorithm Comparison in Analyzing Crime

Researchers used various machine learning techniques in analyzing crime by comparing how they effectively performed. Kim et al. (2017) analyzed crime through machine language by using KNN and Boosted Decision Tree by measuring their accuracy. The accuracy for both algorithms was 39% and 44% respectively, with boosted decision tree having the best accuracy. The limitation of their work shows that the accuracy of the model has a low prediction rate and that it needed further analyses. Noora and Wala (2017) used both KNN and Naïve Bayes classifier for predicting crime in San Francisco. Their approach was based on comparing both classifiers i.e. the KNN and Naïve Bayes classifier. For the KNN classifier, two different techniques were deployed i.e. uniform and inverse techniques. While Gaussian, Bernoulli, and Multinomial techniques were deployed in Naïve Bayes. The result obtained shows that KNN performed poorly as it takes a longer time to execute during the classification and regression stage. The result obtained from the Naïve Bayes Gaussian was poor, and it thus indicates that the data used was not a continuous data but discrete. Naïve Bayes Bernoulli and Multinomial showed a better result amongst the proposed techniques although the data they used was applied directly on the training data set without checking for either errors or outliers. Yuanyuan (2017) proposed a system for analyzing email threats by comparing three machine algorithms i.e. Naïve Bayes, SVM, and Radom Forests. Scaling of Min and Max (0,1) was implemented to normalize all the fields. Radial Basis Function (RBF) was used to normalize the fields in SVM. The 10-fold cross-validation was employed to measure the performance of the three algorithms. The result obtained from the experiment

shows that Random Forests performed best with an accuracy of 92%, while SVM and Naïve Bayes had 90% and 84% respectively. The work was limited to smaller emails which had little impact on the training data sets.

Malathi and Santhosh (2011) tried to enhance algorithms that will aid in predicting crimes in India. MV and Apriori were the two algorithms they concentrated on that will aid in the process of fast-tracking crimes, also for filling of missing values in crime dataset. The result they got showed that the data mining tools deployed on the project will be able to identify crime patterns and future crimes in India. Tahani et al. (2015) predicted crime based on the type of crimes by using spatial and temporal criminal hotspots. Decision tree classifier and Naïve Bayesian classifier were used to predict crime types across different locations in order to help law enforcement agencies in predicting crimes in a specific location. Apriori algorithm was then implemented for them to identify the frequent crime patterns. The result they got from their analysis shows that both the Decision tree classifier and Naïve Bayesian classifier got an accuracy of 51% and 54% respectively in predicting crimes across different locations. Kiran and Kaishveen (2018) analyzed crime by using a clustering approach. Their work was based on comparing the Naïve Bayesian classifier and KNN classifier to see how well the model performed in analyzing crimes in India in terms of accuracy and execution time. Both models had an accuracy of 87% and 77%, with the execution of time 0.2seconds and 0.5seconds respectively, with Naïve Bayesian classifier performing better. Mrinalini and Shaveta (2019) used Naïve Bayes and KNN for predicting crime in India. Their aim was to compare both models in terms of accuracy to find out which perform better in predicting crime. The proposed techniques were implemented in python, both techniques had an accuracy of 77% and 96% with Naïve Bayes proving to be better although their work could not handle crime dataset with lager features. Akash et al. (2019) used a gradient boosting algorithm to predict crime and compare the result with random forest. The result showed that gradient boosting performed better in terms of accuracy, recall, precision, and f1 score compared to the random forest. Rasoul et al. (2015) analyzed crime by using the clustering and classification models. Their work focuses more on classifying crimes based on occurrence within different years. A genetic algorithm was used in other to optimized outliers in the dataset. Their work was not good enough as the proposed model did not perform very well since the number of clusters where not detected properly at the clustering phase. Keivan and Reda (2006) worked on crime prediction using a support vector machine to predict crimes via location. They concluded by saying that SVM performed better in predicting hotspot crime, and K means clustering is better when it comes to data selection.

Shao-chong et al. (2018) used a decision tree algorithm to predict offenders that indulge in crime. They also implemented other algorithms and compared the result via accuracy to see how they performed (Bayes Network, Logistic Regression, and Naïve Bayes). The decision tree performed better with an accuracy of 80% than other algorithms that were used. Renjie et al. (2010) used Bayesian learning theory to predict crime in Gansu, China. Their main work focuses more on identifying the location where crime will occur by combining it with geographical factors. Due to the limitation of crime data, the geographical features were only considered as the only that affect crime prediction which in turn caused the model to have

poor results. Shiju and Surya (2014) also analyzed crime by Naïve Bayes and Decision Tree in forecasting crime factors in India. The data used for this work was unstructured as it was obtained from web sites, blogs, etc. Naïve Bayes performed better but the accuracy was not good enough as only limited crime factors were taken into consideration. Varvara and Sergey (2018) applied three machine learning algorithms to predict the type of crimes that occurs frequently. The proposed models considered for the work were logistic regression, linear regression, and gradient boosting. Their objectives were to compare the three models to see which perform better in terms of accuracy. The gradient boosting model performed better, while linear regression predicted more negative values**.** Jazeem and John (2015) proposed a hybrid approach in predicting crime using deep learning techniques. Their analysis was based on two approaches: the first was to assist decision-makers in making the right decision in a predictive environment through visual analysis. The second approach was on sentiment analysis and topic detection using natural language processing and semantic analytics. In their work, they found out it was possible to predict crime by focusing on crime patterns and contributing factors, although their algorithm could not handle a wide variety of data causing their model to perform poorly. Ilhan Turken (2016) developed a predictive system on Hotspot crime at the University of Cincinnati. The aim of the predictive system was to decrease crimes on campus and predict crime before it happens by the means of the heat map and crime mapping which in turn plays a significant role in helping law enforcement in the United States reduce the rate of crime.

2.4 A Critical Review on Sentiment Analysis for Analysing Crime

Sentiment analysis was also used to predict crime. Some researchers combined tweets and weather to identify crime types and the location where it will occur. Xinyu et al. (2015) proposed a model for crime prediction using twitter sentiment and weather by applying lexicon-based methods, and understanding of weather how they are categorized, combining it with kernel density estimation on historical crime and predicting using a linear model. The result obtained showed that kernel density estimation exceeded the benchmark model, and there was a correlation between crime predictors on weather and sentiment. Hardi and Patel (2017) used sentiment analysis to predict crimes by identifying the location and the type of crime by applying the lexicon-based methods. Although, the accuracy of the model wasn't accurate as there was no correlation between positive and negative opinions.

From the reviewed literature above, it is evident that researchers have used various machine learning algorithms in analysing crime by means of clustering and classification. It is visible that the decision tree had a better result compared to other models according to Emmanuel et al (2017), Rizwan et al. (2013), Danison et al (2017), and Kim et al. (2017) when predicting crime. Random forest was more effective according to Yuanyuan (2017), likewise support vector machine (SVM) according to Keivan and Reda (2006), naïve bayes performed better than other algorithms according to Tahani et al (2015), Kiran and Kaishveen (2018), Mrinalni and Shaveta (2019), Shiju and Surja (2014). The model evaluation performance that was adopted for this research will be based on precision, recall, f1 score, and accuracy which was previously adopted by Akash et al (2019).

2.5 Gaps Identified in the Research

From the reviewed literature, it is evident that some of the research work done by the previous researchers showed that the techniques deployed in crime classification were significantly poor in terms of accuracy which needed further analysis like the work done by Malathi and Santhosh (2011), Tahani et al (2015), Kim et al. (2017), Noora and Wala (2017), Kiran and Kaishveen (2018). For this reason, this research aims at paying attention closely to improve the performance of the existing machine algorithm that has been used by previous researchers in crime analysis. The data obtained from Chicago data porter (crime_2019) was trained in order to implement the choosing machine learning algorithms. These algorithms were chosen as it has proved to be more effective when it comes to analyzing crimes like classification and clustering algorithms (Emmanuel et al 2017) (Rizwan et al 2013).

2.6 Comparison and Conclusion

A comparative review of related works done by researchers in terms of criteria, algorithms used, evaluation metrics, the model with the best performance that was adopted by different authors in analyzing crimes were carried out. The results are presented in table 2 below.

Table 2: Comparative Review of Related Work

Criteria	Algorithms	Evaluation Metrics	Algorithms with better performance	Result	Authors
Crime prediction using BI techniques (supervised learning)	Decision Tree, Naïve Bayes, Multilayer Perceptron, and SVM	Accuracy, Precision and Recall	Decision Tree and Multilayer Perceptron	All metrics had 100%	Emmanuel et al (2017)
Crime prediction using classification model	Decision Tree and Naïve Bayes	Accuracy, Precision and Recall	Decision Tree	Accuracy and precision 84%	Rizwan et al (2013)
Classification algorithm for crime prediction	Decision Tree	Accuracy	Decision Tree	94%	Danison et al (2017)
Analyzing crime	KNN and Boosted Decision Tree	Accuracy	Boosted Decision Tree	44%	Kim et al. (2017)

Predictive analysis for identifying email threats	Radom Forest, SVM, and NB	Accuracy	Radom Forest	92%	Yuanyuan (2017)
Crime prediction	Decision Tree and Naïve Bayes	Accuracy	Naïve Bayes	54%	Tahani et al (2015)
Prediction of crime offenders	Decision Tree, Bayes Network, Logistic Regression, and NB	Accuracy and precision	Decision Tree	80% and 82%	Shao-chong et al (2018)
Crime Prediction	Naïve Bayes and KNN	Accuracy	Naïve Bayes	96%	Mrinalini and Shaveta (2019)

From the literature above, it is evident that researchers have used different approaches in tackling crime in terms of algorithms, models, and techniques with accuracy widely used as the most evaluation metric in evaluating the performance of the various machine learning algorithms used for crime analysis and prediction. Also, some techniques gathered from the above literature were supervised machine learning algorithms (decision tree, random forest, support vector machine, logistic regression, naïve bayes, and knn) to analyze crimes.

3 Methodology and Design Specification

This chapter explains the scientific method and architectural design that was implemented for this research work. This research work adopted the crime incident methodology for the research aim and objectives to be met and the 2-tier architectural design was also adopted for this work.

3.1 Crime Incidents Methodology Approach

This research work focuses more on enhancing the performance of supervised machine learning algorithms in classifying crime incidents that will assist law enforcement to have an insight regarding the most common types of crime and the necessary actions needed to reduce the crime rate. For this research work, the cross-industry standard process for data mining (CRISP-DM) methodology was modified and embraced for this project as it is in line with the set objectives. According to Pete et al (2000), CRISP-DM is a data mining model that shows the lifecycle of a data mining project. The CRISP-DM consists of six (phases), showing their respective tacks and the relationship amongst the tasks. The modified CRISP-DM is conferred in figure 1 below.

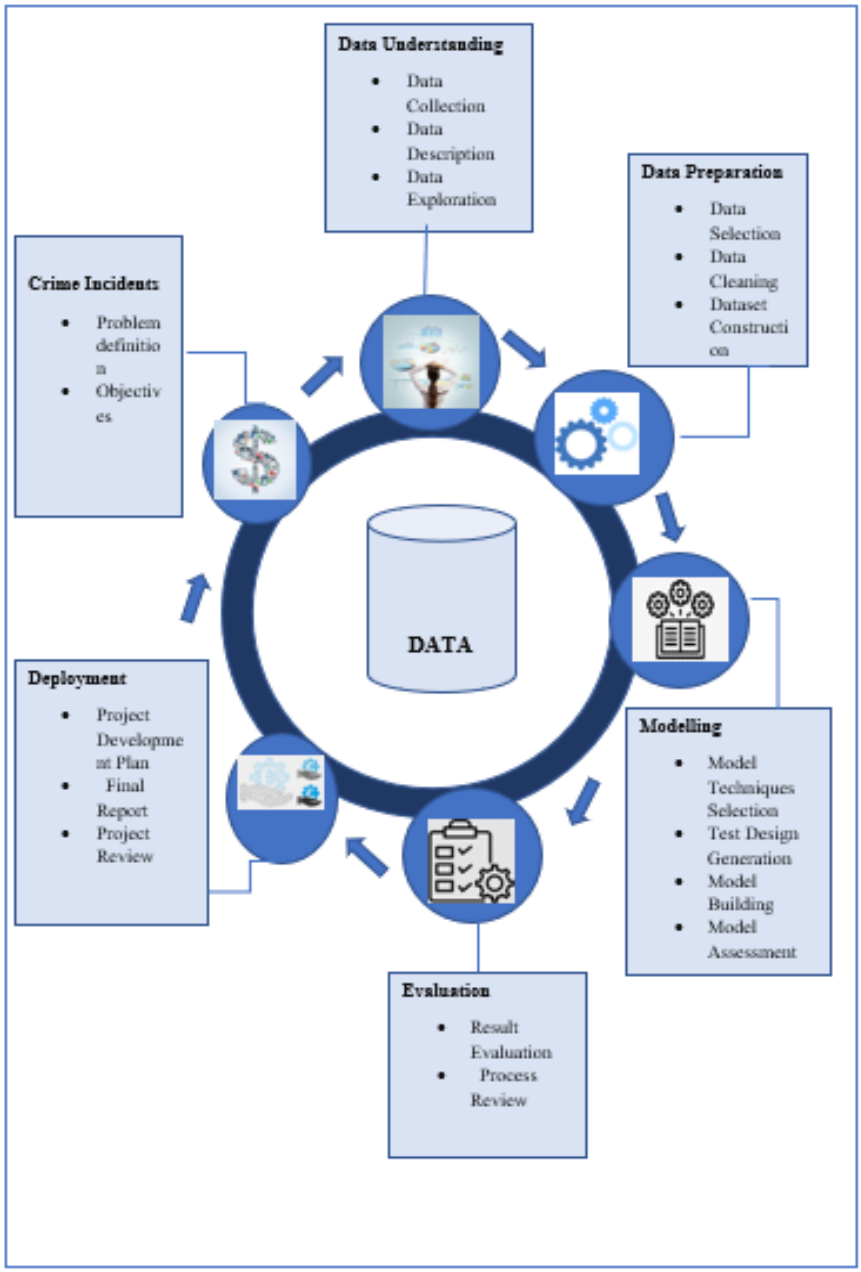

Figure 1: Crime Incidents Methodology

Crime Incidents: This section focuses on the research problem and objectives. The research objectives were identified from past work that has been done to solve the data mining problems that are associated with crime classification. This work also focuses more on using existing data mining techniques in solving the problem associated with crime incidents and measuring how accurately the techniques performed as this will help in the reduction of crime.

Data Understanding: For the set objectives to be accomplished, the need for a suitable dataset was taken into serious consideration. The historical dataset used for this work was obtained from the Chicago data portal. The dataset covered a period of one year (2019) and contained crime_2019.csv file that had 146,914 rows and 30 columns which was adequate for the research objectives. The file was then uploaded in Jupyter notebook where the cleaning

and data preparation process was carried to ensure data uniformity before performing exploratory analysis on the cleaned data.

Data Preparation: The cleaned data was then divided into two forms based on the set objectives for the research. The first form involved using all the variables obtained from the historical data to develop the machine learning models. The second form involved using important variables that were selected through Pearson correlation to develop the second machine learning models.

Modelling: In this phase, supervised classification machine learning models were implemented. Logistic Regression, K Nearest Neighbors, Naïve Bayes, Decision Tree Classifier and XGBoost were implemented as is best used for classification problems.

Evaluation: In this phase, a comparative analysis was carried out to measure the performance of the techniques implemented for this research by using precision, recall, f1 score, and their accuracies.

Deployment: This is the final stage of the project. The result obtained from the project was visualized in python using matplotlib and ggplot**.** Also, a general overview of the project was carried out to ensure the project goals were met.

3.2 Project Architectural Design Specification

For this research work, the 2-tier architecture was deployed i.e. the frontend and backend. The frontend is referred to as the presentation layer, and the backend is the business layer. Figure 2 shows the architectural design structure that was implemented for this research work.

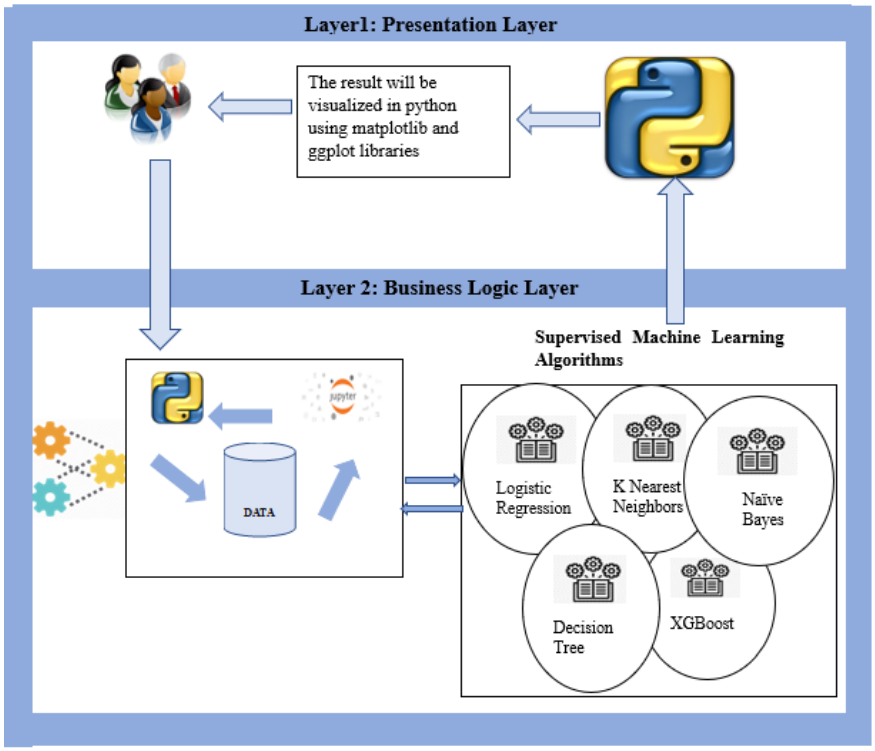

Figure 2: Design Specification for Crime Incidents Classification

The first layer which is the presentation layer shows how the result was visualized for end-users using matplotlib and ggplot which are both libraries in python. The second layer which is the business layer shows how the data was used in python and how the various machine learning techniques that were implemented for crime classification.

For this project, the crime incidents methodology was adopted from CRISP-DM and every stage of the methodology was strictly followed for the objectives to be met. Also, the 2-tier architectural design was adopted which consists of the presentation layer that shows the visualization of result for end-users, and a business logic layer which shows the back-end processes and all the techniques that were implemented for this project. The methodology adopted for this project was used as a guide during the implementation phase for crime classification using supervised machine learning techniques.

4 Implementation, Evaluation and Result of Crime Incidents Classification Models

4.1 Introduction

This section explains the various processes that were carried out in the classification of crime indents in order to the achieve the objectives of the research work. The implementation involves the extraction of meaningful features from the dataset that was used in this research work, and the steps taken were explained in the section. All the models were developed using Python code in jupyter notebook due to its flexibility and ability to handle large scale dataset. The Anaconda which is an open source distribution of the Python was used. In other for the results obtained from the implementation to the evaluated, the following evaluation matrices were adopted from the literature.

Precision: Shows how the crime classification model correctly classified the various crimes i.e. the crime model says that the various crimes belong to a class and they truly belong to that class. It is represented by the formula below:

$$\frac{TP}{TP + FP}$$

where TP or True Positive is the number of crimes that belong to a class and were classified correctly, and FP or False Positive is the number of crimes that belongs to a class and were not correctly classified.

Recall: Shows how the crime classification model classifies the various crimes correctly if they belong to the same class. It is represented by the formula below:

$$\frac{TP}{FN + TP}$$

where TP or True Positive is the number of crimes that belongs to a class and were classified correctly, and FN or False Negative is the number of crimes that were wrongly classified to belong to a class of crime.

Accuracy: It gives the total number of crimes that were correctly classified by the classification models as either true positive or true negative. It is represented by the formula below:

$$\frac{TP + TN}{TP + TN+FP+FN}$$

where TN or True Negative is the number of classified crimes that was correctly classified as not belonging to a class.

F1 Score: Returns the proportion of precision and recall. It is represented by the formula below: 2 * (Precision * Recall/ Precision + Recall). This section also describes the comparison of the various crime incidents classification models that were deployed based on the evaluation metrics that were adopted from previous research work.

4.2 Data Pre-processing and Exploratory Data Analysis of Crime Incidents Classification

The dataset used for this research was obtained from Chicago [1]data portal and it covers a period for the year 2019. It was downloaded in csv format with a size of 37. 6MB.The "Panda" library in Python was used to import the downloaded data into Jupyter in other to get the overall summary of the data that contained 146,913 rows and 30 columns. A python function was called upon to replace strings that have space with an underscore, the isnull().sum() function was used to check for the presence of missing values in the imported data. 2160 rows were removed from 146912 rows that were initially loaded into Jupyter after handling the missing values. 144752 rows were realized after the removal of the missing values which was still enough to perform the experiment. For the model to perform better, the Pandas.to_datetime() function was implemented as the date was originally in string in the crime data frame, this leads to additional columns in the crime data frame (day, month, time, and date of crime), also irrelevant column like ID and Case_Number were dropped. The function pd.factorize was used to encode variables that were categorical into numerical values that were present in the crime data as some of the selected supervised machine learning algorithms that were deployed for this research work performs better on numerical data.

In other for the imported dataset to be understood, exploratory data analysis was carried out. Two (2) inquiries were set aside and to respond to each inquires, the data were transformed in the pre-processing phase. These inquiries were very vita as it aids in a better understanding of the processed data, and they are as follows

1. What is the percentage of crime arrest in the city?
2. What is the percentage of domestic crime violence?

For question 1 to be answered, the Arrest column was plotted against the target variable which is the primary_type. Figure 3 below shows that the percentage of arrest in the city of Chicago was 51.8% true and about 48.2% of the crime committed were not punishable.

[1] Chicago crime:https://data.cityofchicago.org/Public-Safety/Crimes-2019/w98m-zvie

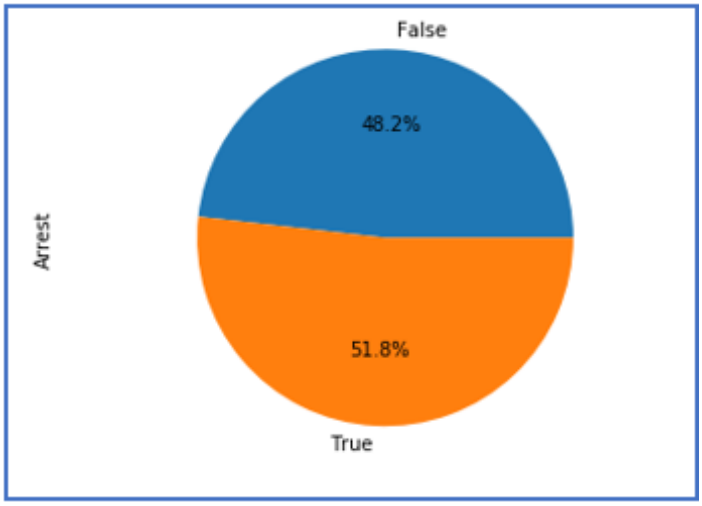

Figure 3: Percentage of Crime Arrest

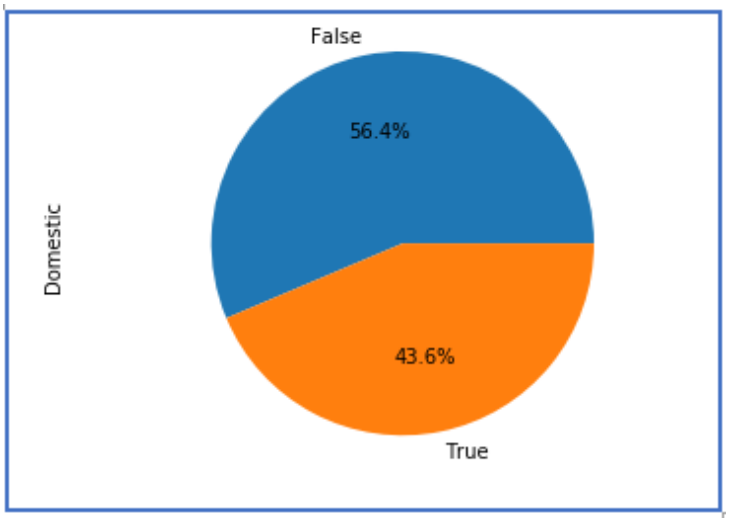

Figure 4: Percentage of Domestic Crime

4.4 Feature Engineering and Data Modelling

The exploratory analysis carried out on the target variable (primary_type) showed that it was a multiclass classification problem with over 20 crimes committed in the city of Chicago. Some of the crimes committed had lower values and this led to the mapping of some of the crimes into a similar group which in turn reduced the number of crimes into 8 classes. The same technique was also applied to the locations that are prompt to crime. The result also showed that over 30 locations were affected by crime, and as such similar location was mapped into one group which resulted in the location having 8 classes too. To avoid overfitting in the model, model_selection function was imported to split the data into train and test set, 70% was for the training set to build the crime classification model and 30% was for the test set to test the performance of the crime classification model. For a better analysis, the resampling technique was then applied for data balancing as the data were imbalanced after mapping the target variable into similar group.

Experiment one was done to answer the research question that was mentioned in section 1.2. The logistic regression model was trained for the model to correctly classify the various crimes with the help of an inbuilt function called sklearn.linear_model. K nearest neighbour, naïve bayes, decision tree classifier, and xgboost libraries were trained also to classify crimes. The second experiment was done to enhance the model performance by using feature importance and correlation with a heatmap to get the best features in the crime dataset. Four (4) variables were identified as the best amongst the variables that were initially used to train six algorithms in the first experiment. The four (4) important variables were used to retrain the same algorithms that were implemented in experiment one. The new variables were then imported into a new data frame by using the pd.DataFrame function. Grid search which is a hyperparameter tuning technique was implemented in other to improve the performance of the five algorithms that were trained in the second experiment.

4.5 Implementation, Evaluation and Results of Logistic Regression

Logistic regression is a supervised machine learning classification algorithm that has been used widely in solving classification problems although it can also be used for predictively or regression problems. Logistic regression can either be

- Binomial Classification Problem: This is when target variable has just two categories e.g. ‘yes’ and ‘no’, ‘male’ and ‘female’, ‘married’ and ’single’, etc
- Multinomial or Multiclass Classification Problem: This is when target variable has more than two categories e.g. ‘Crime A’ vs ‘Crime B’ vs ‘Crime C’ vs ‘Crime D’ vs ‘Crime E’, etc.
- Ordinal Classification Problem: This deals with classification problems that are in order e.g. ‘Tall’, ’Taller’,’ Tallest’ and as such can be represented as 0,1,2 or 1,2,3.

For this research work, the target variable is multinomial i.e. “THEFT”, “NON_CRIMINAL_ ASSUALT”, “CRIMINAL_OFFENSE”, “OTHER_OFFENSE”, “NARCOTIC_OFFENSE”, “WEAPONS_OFFENSE”,“SEXUAL_OFFENSE”, “HUMAN_TRAFFICKING_OFFENSE” which was coded from 0 to 7 as the default numbering in python starts from 0.

4.5.1 Implementation

For experiments 1 and 2, a library in python called sklearn was used in the implementation of the logistic regression model after balancing the data with RESAMPLING. The LogisticRegression() function was used to implement the model, it was then fit into the trained data. The fit model was then tested on the test set in to check the performance of the developed model. The same procedure was also carried out in experiment two but this time, grid search was introduced for hyperparameter tuning to optimize the performance of the model by introducing the function GridSearchCV. The parameters chosen for the hyperparameter tuning was setting Penalty to be l1 and l2, C to be 1,10,100,1000, cross-validation of 10 folds, and n_jobs to be -1 due to the large volume of the dataset to enable it to run faster.

4.5.2 Evaluation and Result

The sklern.metrics is a library in python that was used to evaluate the accuracy, precision, recall and f1 score for the model. The result obtained from the first experiment shows that logistic regression had an accuracy of 0.181 i.e. the model's overall performance in classifying crimes was 18.1%. The precision, recall and f1 score for the model were 16.2%,18.1%, and 15.6% respectively. This result shows that the performance of the model was significantly poor in classifying crimes. For experiment 2, there was a significant increase in the performance of the model as the accuracy, precision, recall, and f1 score had an improved value i.e. 39.8%, 42.1%, 39.9%, and 38.2% respectively.

4.6 Implementation, Evaluation and Results of K Nearest Neighbors

K Nearest Neighbors or KNN is another algorithm that belongs to the family of a supervised machine learning algorithm. It is also implemented for solving both classification and regression problems. KNN is easier to use, has quick execution time although the quality of the data defines the model accuracy, and the k value (nearest neighbour) must be optimal.

4.6.1 Implementation

For experiments 1 and 2, a library in python called sklearn was used in the implementation of the knn model, the data was standardized as knn works on metrics distance before applying RESAMPLING to have a balanced data. The KNeighborsClassifier() function was used to implement the model, it was then fit into the trained the data. The fit model was then tested on the test set to check the performance of the developed model. No parameters were used for the first experiment. The same procedure was also carried out in experiment two but this time, grid search was introduced for hyperparameter tuning to optimize the performance of the model by introducing the function GridSearchCV. The parameters chosen for the hyperparameter tuning was setting n_neighbors=5, metric='minkowski', p=2, n_jobs=-1. The execution time for knn was high after the implementation of grid search as it took long hours to obtain the result.

4.6.2 Evaluation and Result

The sklern.metrics is a library in python that was used to evaluate the accuracy, precision, recall and f1 score for the model. The result obtained from the first experiment shows that k nearest neigbors had an accuracy of 0.812 i.e. the model overall performance in classifying crimes was 81.2%. The precision, recall and f1 score for the model were 80.5%, 81.3%, and 80.7% respectively. This result shows that the performance of the model was significantly higher than the logistic regression in classifying crimes. Both experiment one and two had the same results with regards to accuracy, precision, recall and f1 score due to the balance data that was used to perform the experiment.

4.7 Implementation, Evaluation and Results of Naïve Bayes

Naïve Bayes is a supervised learning model that uses the principle of Naïve Bayes Theorem to get the probability of samples to be in a certain class i.e. every sample in each class

contribute independently and then the classification of the sample is determined by the probability and output the category with the highest probability sample.

4.7.1 Implementation

For experiments 1 and 2, a library in python called sklearn was used in the implementation of the naïve bayes model after balancing the data with RESAMPLING. The GaussianNB() function was used to implement the model, it was then fit into the trained the data. The fit model was then tested on the test set in other to check the performance of the developed model. No parameters were used because the GaussianNB() function has no inbuild parameters causing the model to execute faster than any other model used for this research work. The same procedure was also carried out in experiment two. GridSearchCV was not applied to this algorithm because it has no inbuild parameters.

4.7.2 Evaluation and Result

The sklern.metrics is a library in python that was used to evaluate the accuracy, precision, recall and f1 score for the model. The result obtained from the first experiment shows that naïve bayes had an accuracy of 0.524% i.e. the model overall performance in classifying crimes was 52.4%. The precision, recall and f1 score for the model were 50.7%, 52.4%, and 49.3% respectively. This result shows that the performance of the model was significantly higher than logistic regression but was outperformed by knn in classifying crimes. For experiment 2, there was a significant increase in the performance of the model as the accuracy, precision, recall, and f1 score had an improved value i.e. 67.5%, 68.8%, 67.5%, and 64.6% respectively.

4.8 Implementation, Evaluation and Results of Decision Tress Classification

A decision tree classifier is a supervised machine learning algorithm that is used for both classification and regression problems. The decision tree classifier acts like a flowchart or a tree-like structure that breaks down data into smaller subsets and by so doing, the decision tree is gradually created. The execution time is faster, and easy to understand the output than other models but is often prone to overfitting i.e. a model that learns too much on the training set of data.

4.8.1 Implementation

For experiments 1 and 2, a library in python called sklearn was used in the implementation of the decision tree classifier model after balancing the data with RESAMPLING. The DecisionTreeClassifier() function was used to implement the model, it was then fit into the trained data. The fit model was then tested on the test set in other to check the performance of the developed model. The parameters used for the first experiment was setting criterion to be 'entropy' and random_state to be 0. The same procedure was also carried out in experiment two but this time, grid search was introduced for hyperparameter tuning to optimize the

performance of the model by introducing the function GridSearchCV. The parameters chosen for the hyperparameter tuning was setting max_features to be auto, sqrt, and log2, min_samples_leaf to be 1-11, min_samples_split to be 2-15, random_state to be 0, the criterion to be gini and entropy, max_depth to be 2,5 and 10, cross-validation of 10 folds, and n_jobs to be -1 due to the large volume of the dataset to enable it to run faster but it took longer time to execute i.e. a couple of hours.

4.8.2 Evaluation and Result

The sklern.metrics is a library in python that was used to evaluate the accuracy, precision, recall and f1 score for the model. The result obtained from the first experiment shows that the model overfit due to the balanced of the data which resulted in all the metrics to have same results (100%). This result shows that the performance of the model was significantly higher than logistic regression, k nearest neigbors classifier, and naïve bayes. For experiment two, there was a slight dropdown in the performance of the model as the accuracy, precision, recall and f1 score had value of 99.9%, 99.6%, 99.6%, and 99.6% respectively.

4.9 Implementation, Evaluation and Results of XGBoost

Extreme Gradient Boosting popularly known as XGBoost is a machine learning algorithm that uses gradient boosted decision trees for the purpose of speed and the optimization of model performance. XGBoost package was download and installed in python through anaconda command prompt as the package was not installed initially in python.

4.9.1 Implementation

For experiments 1 and 2, a library in python called sklearn was used in the implementation of the XGBoost model after balancing the data with RESAMPLING. The XGBClassifier () function was used to implement the model, it was then fit into the trained the data. The fit model was tested on the test set in other to check the performance of the developed model. The parameters used for the first experiment were setting max_depth to be 3, n_estimators to be 100 as the dataset used for the model was quite large and learning_rate to be 0.03. The same procedure was also carried out in experiment two but this time, grid search was not applied to this algorithm because it took long hours to execute.

4.9.2 Evaluation and Result

The sklern.metrics is a library in python that was used to evaluate the accuracy, precision, recall and f1 score for the model. For experiment one, all the evaluation metrics had the same result with regards to accuracy, precision, recall and f1 score (99.6%). For experiment two, all the evaluation metrics had the same result with regards to accuracy, recall and f1 score (97.3%), and precision had 97.4%. Although, the hyperparameter tuning was not implemented in XGBoost as the computation took longer time to process without getting the desired result. Table 8 and 9 below gives an overview of all the results obtained from both experiments.

Table 3: Results for experiment one using all variables

Model	Accuracy	Precision	Recall	F1 Score
Logistic Regression	0.181	0.162	0.181	0.156
K Nearest Neighbors	0.812	0.805	0.813	0.807
Naïve Bayes	0.524	0.507	0.524	0.493
DT Classifier	1	1	1	1
XGBoost	0.996	0.996	0.996	0.996

Table 4: Results for experiment two using important variables

Model	Accuracy	Precision	Recall	F1 Score
Logistic Regression	0.398	0.421	0.399	0.382
K Nearest Neighbors	0.812	0.805	0.813	0.807
Naïve Bayes	0.675	0.688	0.675	0.646
DT Classifier	0.996	0.996	0.996	0.996
XGBoost	0.973	0.974	0.973	0.973

4.10 Comparison of Developed Models

The precisions and accuarcies of the five developed models (Logistic Regression, K Nearest Neighbors, Naïve Bayes, Decision Tree Classifier and Extreme Gradient Boosting or XGBoost on crime incidents classification is presented in figure 5 below with decision tree and xgboost having a precision score of 99.6% and 97.4% respectively i.e. the classification model correctly classifies the various crimes with a score of 99.6%.

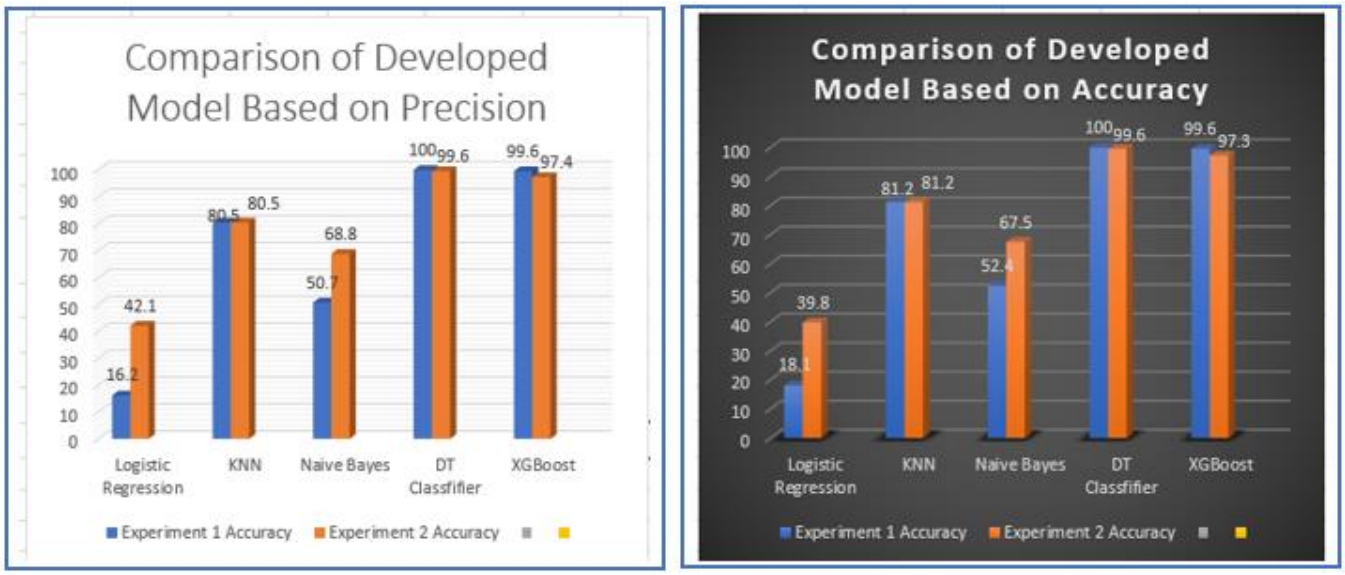

Figure 5: Models comparison based on Precision and Accuracy

4.11 Comparison of Developed Models with Existing Models

Table 5 below shows the comparison between the developed crime incidents classification model and existing models gathered from the literature view as discussed in section 2.6.

Table 5: Comparison of crime classification with existing models

Criteria	Models	Accuracy	Authors
Crime prediction using classification model	Decision Tree	84%	Rizwan et al (2013)
Crime Prediction	Naïve Bayes	54%	Tahani et al (2015)
Crime Prediction using BI techniques	Decision Tree and Multilayer Perception	100%	Emmanuel et al (2017)
Crime Analysis	Boosted Decision Tree	48%	Kim et al. (2017)
Prediction of crime offenders	Decision Tree	80%	Shao-chong et al (2018)
Crime Prediction	Naïve Bayes	96%	Mrinalini and Shaveta (2019)
Crime Classification	**XGBoost**	**97%**	**Nelson Edoka (2020)**

4.12 Conclusion

Conclusively, the results obtained from the implemented models for this research work has completely answered the research question that was outlined in section 1.2. The collection of crime data, the performance of the data pre-processing, the exploratory data analysis, the feature engineering that was carried out on the crime data, and the implementation of resampling technique on the crime data to deal with imbalanced data completely satisfy Objective 2 in section 1.3 (Table 1). The implementation of the supervised machine learning models for crime classification (Logistic Regression, K Nearest Neighbors, Naïve Bayes, Decision Tree Classifier and XGBoost) satisfies Objective 3 in section 1.3 (Table 1). Lastly, the comparison of the developed model as shown in Figure 5 above and comparing also the developed models with existing models obtained from the literature review as shown in Table 5 satisfies the Objectives of 4 and 5 as shown in section 1.3 (Table 1).

5 Discussion

For the research questions to be answered as set in section 1.2, two experiments were conducted. The first experiment focuses on using the entire variables obtained from the crime data to classify crime types. Due to the main aim of the project to have a better performance of the models, the second experiment was conducted but this time using important variables that lead to the implementation of feature selection as discussed in section 4.4. The result obtained from the first experiment showed that logistic regression performed poorly having an accuracy of 18.1%, precision of 16.2%, recall of 18.1% and 15.6% for f1 score. Naïve bayes and knn performed logistic regression both having accuracies of 52.4% and 81.2%, precisions of 50.7% and 80.5%, recalls of 52.4% and 81.3%, f1 scores of 49.3% and 80.7% respectively. Decision tree and XGBoost performed better than the other models with an accuracy of 100% and 99.6%, precisions of 100% and 99.6%, recalls of 100% and 99.6%, f1 scores of 100% and 99.6% respectively. The result obtained from decision tree classifier shows that is does not work very well for balanced data as it is prone to overfitting. In terms of execution time, naïve bayes took less time to execute compared to other models that was used for crime classification for this research work.

To enhance the model performance, the need for experiment two came into play by using important variables. The result obtained from the experiment shows a significant increase in performance than experiment one. The accuracies of logistic regression and naïve bayes increased (39.8% and 67.5%) with precision of 42.1% and 68.8% respectively. Xgboost, knn and decision tree had accuracies of 97.3%,81.2%, and 99.6% respectively. Another noticeable increase was in the case of precision, recall and f1 score, all the models performed better as shown in table 4 above compared to experiment one as shown in table 3 above, which means that using important variables in classifying crimes was able to enhance the performance of the supervised machine learning techniques. Going back to the gaps identified from the previous works in section 2.5, the developed models from experiment two had improved accuracies compared to the work done by Malathi and Dr.Santhosh (2011), Tahani et al (2015), Kim et al. (2017), Noora and Wala (2017), Dr.J.Kiran and Kaishveen (2018) whose accuracies were poor.

6 Conclusion and Future Work

For this research work, feature engineering was applied by mapping all the crime types into similar groups for better analysis. For better performance of the models, grid search which is a hyperparameter tuning was applied, RESAMPLING was then applied for the balancing of the crime data before the implementation of the five supervised machine learning algorithms that were used to classify crime incidents. Logistic regression, k nearest neighbors, naïve bayes, decision tree classifier, and xgboost were all implemented and evaluated to find out which performs better in crime classification. All the models performed very well with improved accuracy, precision, recall, and f1 score after conducting the second experiment but it turns out to be that xgboost performed better than other models having accuracy and precision of 97.3% respectively.

Conclusively, machine learning algorithms have made it easier to classify crimes on historical data. The result obtained from experiment two agrees with the research question stating that if the supervised machine learning techniques can be enhanced to perform better in classifying crime incidents to assist government and law enforcement agencies in the prevention of crime occurrences by using the important features as deployed in the second experiment. This research work can further be optimized by using an embedded method to avoid the problem of overfitting as seen in the decision tree classifier. Analysis can also be carried out on the factors that lead to the contribution of crimes using machine learning techniques to enable the government and law enforcement agencies to mitigate crimes and to ensure the safety of every individual as this research work pays more attention in the classification of crime.

Acknowledgement

I want to use this medium to thank my supervisor Dr. Catherine Mulwa for her words of encouragement and supervision throughout this research work. My sincere appreciation also goes to my lovely family for their support also. I would also want to acknowledge my beloved friends Akeem Lekan Ayantola and Kenneth Anuforo for their contributions towards this research work and above all give thanks to God almighty for making this work to be accomplished.

References

Akash, Z., Gaurav, D. and Avinash, S. (2019) 'Identifying Email Threats Using Predictive Analysis'. *ICAESMT,* pp.1-7

Danison, T., Ivan, N., Elisha, O. and Emmanuel, A. (2017) 'Crime Prediction Using Decision Tree (J48) Classification Algorithm'. *IJCIT,* 6(3), pp.188-195

Dataschool (2015) *Comparing Supervised Learning Algorithms.* Available at: https://www.dataschool.io/comparing-supervised-learning-algorithms/ [Accessed 3rd April 2020]

Emmanuel, A., Ivan, N., Elisha, O. and Ruth, W. (2017) 'A Performance Analysis of Business Intelligence Techniques on Crime Prediction'. *IJCIT,* 6(2), pp.84-90

Geeksforgeeks (2016) *Understanding Logistic Regression.* Available at: https://www.geeksforgeeks.org/understanding-logistic-regression/ [Accessed 11th March 2020]

Hardi, P. and Ripal, P. (2017) 'Enhance Algorithm to Predict A Crime Using Data Mining'. *JETIR,* 4(4), pp.257-259

Ilhan, T. (2016) 'Crime Analysis Through Machine Learning'. Available at: https://www.academia.edu/33324653/Hotspot_Crime_Prediction?auto=download [Accessed 20th March 2020]

Keivan, K. and Rada, A. (2006) 'Crime Hot-Spots Prediction Using Support Vector Machine'. *IEEE,* pp.952-959

Kim, S., Param, J., Parminder, K. and Pooya, T. (2017) 'Crime Analysis Through Machine Learning'. Available at: https://www.researchgate.net/publication/330475412_Crime_Analysis_Through_Machine_Learning [Accessed 10th March 2020]

Kiran, J. and Kaishveen, K. (2018) 'Prediction Analysis of Crime in India Using a Hybrid Clustering Approach'. *IEEE,* pp.520-523

Lawrence, M. and Natarajan, M. (2015) 'Using Machine Learning Algorithms to Analyze Crime Data'. *MLAIJ,* 2(1), pp.1-12

Machine Learning Mastery (2017) *A Gentle Introduction to XGBoost for Applied Machine Learning.* Available at: https://machinelearningmastery.com/gentle-introduction-xgboost-applied-machine-learning/ [Accessed 12th March 2020]

Malathi, A. and Santhosh, S. (2011) ‘An Enhanced Algorithm to Predict a Future Crime Using Data Mining’. *IJCA*, 21(1), pp.1-6

Medium (2017) *Why, How and When to Scale your Features.* Available at: https://medium.com/greyatom/why-how-and-when-to-scale-your-features-4b30ab09db5e [Accessed 12th March 2020]

Medium (2018) *Decision Tree Explained Easily.* Available at: https://medium.com/@chiragsehra42/decision-trees-explained-easily-28f23241248 [Accessed 12th March 2020]

Medium (2018) *Easy and quick explanation: Naive Bayes algorithm.* Available at: https://medium.com/@montjoile/easy-and-quick-explanation-naive-bayes-algorithm-99cb5f3f4e9c [Accessed 12th March 2020]

Medium (2018) *K Neighbors Classifier with GridSearchCV Basics.* Available at: https://medium.com/@erikgreenj/k-neighbors-classifier-with-gridsearchcv-basics-3c445ddeb657 [Accessed 12th March 2020]

Medium (2019) *K-Nearest Neighbors (KNN) Algorithm for Machine Learning.* Available at: https://medium.com/capital-one-tech/k-nearest-neighbors-knn-algorithm-for-machine-learning-e883219c8f26 [Accessed 12th March 2020]

Medium (2019) *Tips and Tricks for Multi-Class Classification.* Available at: https://medium.com/@b.terryjack/tips-and-tricks-for-multi-class-classification-c184ae1c8ffc [Accessed 15th March 2020]

Mrinalini, J. and Shaveta, K. (2019) ‘Naïve Bayes Approach for the Crime Prediction in Data Mining’. *IJCA,* 178(14), pp.33-37

Noora, A. and Wala, A. (2017) ‘KNN Classifier and Naïve Bayes Classifier for Crime Prediction in San Francisco Context’. *IJDMS,* 9(4), pp.1-19

Pete, C., Julian, C., Randy, K., Thomas, K., Thomas, R., Colin, S. and Rudiger, W. (2000) *‘CRISP-DM 1.0’.* 1st edn. Available at: http://www.statoo.com/CRISP-DM.pdf [Accessed 15th March 2020]

Pete, C., Julian, C., Randy, K., Thomas, K., Thomas, R., Colin, S. and Rudiger, W. (2000) *‘CRISP-DM 1.0’.* 1st edn. Available at: http://www.statoo.com/CRISP-DM.pdf [Accessed 15th March 2020]

Prabhjot, K. and Kirti, J. (2019) ‘Crime Prediction Analysis: A Review’. *IJARCET,* 8(1), pp.14-17

Pyactlearn (2016) *Multi-Class Performance Metrics.* Available at: https://pyactlearn.readthedocs.io/en/latest/performance/multiclass.html [Accessed 9th March 2020]

Rasoul, K., Siamak, M. and Amin, K. (2015) 'Analysis and Prediction of Crimes by Clustering and Classification'. *IJARAI*, 4(8), pp.11-17

Renjie, L., Xueyao, W., Lun., L. and Zengchange, Q. (2010) 'A Novel Serial Crime Prediction Model Based on Bayesian Learning Theory'. *IEEE,* pp.1757-1762

Rizwan, I., Masrah, M., Aida, M., Payam, P. and Nasim, K. (2015) 'An Experimental Study of Classification Algorithms for Crime Prediction'. *INDJST*, 6(3), pp.4220-4225

Shap-Chong, S., Peng, C., Peng-Hui., Y., Chao, H. and Hong-xia, M. (2018) 'The Prediction of Offender Identity Using Decision-Making Tree Algorithm'. *IEEE,* pp.405-409

Shiju, S. and Surya, G. (2014) 'Crime Analysis and Prediction Using Data Mining'. *IEEE,* pp.406-412

Shiju, S. and Surya, G. (2015) 'Hybrid Approach to Crime Prediction using Deep learning'. *IEEE,* pp.1701-1710

Tahani, A., Rsha, M. and Elizabeth, L. (2015) 'Crime Prediction Based on Crime Type and Using Spatial and Temporal Criminal Hotspots'. *IJDKP*, 5(4), pp.1-19

Topyaps (2019) *Top Ten Causes of Crime.* Available at: https://topyaps.com/top-10-causes-of-crime/ [Accessed 20th February 2020]

Towardsdatascience (2018) *Feature Selection Techniques in Machine Learning with Python.* Available at: https://towardsdatascience.com/feature-selection-techniques-in-machine-learning-with-python-f24e7da3f36e [Accessed 25th March 2020]

Varvara, I. and Sergey, I. (2018) 'Crime Rate Prediction in The Urban Environment Using Social Factors'. *ELSEVIER*, 136, pp.472-478

Xinyu, C., Youngwoon, C. and Suk, J. (2015) 'Crime Prediction Using Twitter Sentiment and Weather'. *IEEE,* pp.63-68

Yuanyuan, Z. (2017) 'Identifying Email Threats Using Predictive Analysis'. *IEEE,* pp.1-2