Crime Detection and Analysis from Social Media Messages Using Machine Learning and Natural Language Processing Technique

Xolani Lombo, Olaide N. Oyelade(✉), and Absalom E. Ezugwu(✉)

School of Mathematics, Statistics, and Computer Science, University of KwaZulu-Natal,
King Edward Road, Pietermaritzburg 3201, KwaZulu-Natal, South Africa
218014246@stu.ukzn.ac.za, {oyeladeo,Ezugwua}@ukzn.ac.za

Abstract. Social media has dramatically influenced and changed the rate and the nature of crime in our society. The perpetrators cut across different age groups, social standing, and beliefs. The ability to be anonymous on social media and the lack of adequate resources to fight cybercrime are catalysts for the rise in criminal activities, especially in South Africa. We proposed a system that will analyse and detect crime in social media posts or messages. The new system can detect attacks and drug-related crime messages, hate speech, and offensive messages. Natural language processing algorithms were used for text tokenisation, stemming, and lemmatisation. Machine learning models such as support vector machines and random forest classifiers were used to classify texts. Using the support vector machine to detect crime in texts, we achieved 86% accuracy and using the random forest for crime analysis, 72% accuracy was achieved.

Keywords: Crime detection · Social media · Natural language processing · Support vector machine · Random forest

1 Introduction

Crime detection in social media messages identifies the presence of crime in social media posts or messages. Crime analysis is the study of crime and law enforcement data with other information to apprehend criminals and prevent crime [1]. According to South African Police Service (SAPS), social media crime is fast-growing because more criminals or ordinary people exploit anonymity and lack adequate resources to detect the crime. The crime that we focus on is attack and drug-related crimes, hate speech, and offensive messages.

Many people use social media, especially Twitter and Facebook, which is why it is a target for criminal activities. There are numerous reports of cyberbullying, fraud, online threats, and people getting scammed online. Law enforcement agencies are slow and sometimes handicapped to respond to cybercrime due to limited resources [2]. This study was proposed as a base for other researchers to use in real-time to monitor and detect crime in social media posts or messages, which is better than facing the consequences of the crime.

O. Gervasi et al. (Eds.): ICCSA 2022 Workshops, LNCS 13381, pp. 502–517, 2022.
https://doi.org/10.1007/978-3-031-10548-7_37

Natural Language Processing (NLP) is a branch of artificial intelligence that helps computers understand, manipulate, and interpret natural language (human language). The Term Document Inverse Document Frequency (TF-IDF) is used to prepare and represent the output data after processing the data. This gets ready the data to be passed to Machine Learning (ML) models. On the other hand, ML is a branch of artificial intelligence based on the idea that systems can learn from data, identify patterns, and make decisions with the minimal human intervention [3] The NLP was used for text pre-processing (for cleaning and formatting operations) and have also been successfully applied in email filtering [4], recommendation system [5–7], medicine [8–11] r, several ML models have been used for text classification in different crime categories [12–15] Therefore, this study employs the use of ML over deep learning considering the performance track records of ML algorithms in solving the peculiarity associated with the problem considered in the study [16].

An enhanced computational framework with increased accuracy in detecting crime in social media criminal activities is essential to apprehending criminals. A comparative study of classification algorithms known for high accuracy was first carried out in this paper to investigate the high accuracy model. The ML models, support vector machine (SVM), K-nearest neighbor (KNN), and Naïve bayesian were studied for crime detection within the binary classification paradigm. For crime analysis and multi-class classification of texts, random forest (RF) and SVM were used, respectively. As the data for crime analysis was imbalanced, we applied the RF, which is widely reported to demonstrate good performance in cases of imbalanced data. Similarly, we also used an SVM classifier to address such peculiarities for cost sensitivity in imbalanced data.

The paper aims to successfully use NLP for data pre-processing and comparatively study the classification algorithms used for text classification. The specific objectives of the study include:

- To model a framework for accurate detection and analysis of social media messages.
- To accurately detect and analyse crime from social media messages using NLP and ML algorithms, namely SVM and RF.
- To investigate the performance of similar classification algorithms, namely, K-Nearest Neighbor and Naïve Bayes, to detect and analyse crime in social media messages accurately.

The rest of the paper is organised as follows: Sect. 2 consists of a literature review that justifies the importance of the research by reviewing related research. The research methods and techniques of how the problem was addressed are discussed in Sect. 3. In Sect. 4, results and discussion describing the findings and results of the study are presented. Conclusion and future research direction are presented in Sect. 5.

2 Literature Review

Many researchers have addressed text classification problems using NLP algorithms and ML models. This section will discuss techniques that have been used in literature for classifying text, for example, spam detection in messages, as many of the related research are based.

Ashok [17] looked at data mining and sentiment analysis on the online network to help detect crime patterns. Data was collected from Twitter using -related tweets that include the following keywords "gun," 'crime,' "kill," and so forth. Then data was cleaned and processed using a natural language algorithm. The research was a success as it was concluded that tweets with "very negative" sentiment were identified as contributing to crime intensity. Although this research did not strictly follow the traditional NLP methods, it was a success. Following the approach of using all standard NLP algorithms would have resulted in a more accurate study.

Sharma et al.'s [18] research focused on data pre-processing using traditional NLP algorithms in classifying emails as spam or non-spam. The steps for this study were data pre-processing, representation of data, and classification. The following steps were followed during data pre-processing: removing words lesser in length, removing alphanumeric words (words that contain both characters and numbers), removing stop words, and stemming. This research shows that pre-processing plays a crucial role in classifying texts. Another NLP algorithm, lemmatisation after stemming, could be useful. Removing words with lesser length may affect finding the pattern of spam; for example, a word like "sex" are essential in identifying spam.

Shirani-Mehr [19] used five ML models, namely SVM, RF, KNN, Multinomial NB, and Adaboost, as classifiers of SMS spam datasets for SMS Spam Detection. Their work aimed to investigate different ML algorithms for the classification of SMS spam. The only NLP algorithm that was used for data pre-processing was for tokenisation. The research was a great success because he could catch 90.62% of the Spam SMS with an accuracy of 98.57% using RF, the best in the five algorithms being investigated. The researcher caught 92.99% of Spam SMS using SVM with an accuracy of 98.86%. The problem faced by the researcher is that there are fewer datasets for SMS spam compared to email spam. Due to text messages having a small length, the number of features used for their classification is smaller than the corresponding number of emails.

Andrews et al. [20] describe an approach for detecting the presence of organised crime signals on social media. Formal concept analysis is used to group information sources according to crime type and location. NLP algorithms are used to identify, extract, and corroborate information from open web sources, identifying the early onset of organised crime. They used this to establish the idea of 'weak signals' as keywords and phrases that point to criminality.

McCord and Chuah [21] investigated spam detection using the traditional classifiers, SVM, KNN, RF, and Naïve Bayesian. They evaluate the usefulness of the features suggested by Twitter spam policies and observe spammers' behaviours in spammer detection using the mentioned traditional classifiers using the Twitter dataset they collected. Then use these features to help identify spammers. They found that the RF classifier gave the best performance, which achieved 95.7% precision. The second-best performance was the SVM model with an accuracy of 93.5%, and KNN was the third-best got an accuracy of 92.8%, which is also good. This approach of training the data was effective in detecting crime in messages.

Malmasi and Zampieri [19] examine social media methods to detect hate speech. They achieved 78% accuracy in identifying posts across three classes: hate, offensive, and no offensive. For data pre-processing, they convert all texts to lowercase, tokenise and remove URLs and emojis. The approach they used for data pre-processing was very poor. If they had used all traditional NLP methods, the accuracy might have increased. The model that these researchers used is a Support Vector Machine (SVM) classifier to perform multi-class classification in their experiment.

Using ML techniques, the theoretical study for classification by Ikonomakism et al. [22] investigated the text classification process. They mentioned that text classification includes reading the document, tokenising text, stemming, vector representation of text, deleting stop-words, feature selection, feature transformation, and learning algorithms. Lim proposed a method that improves K-Nearest Neighbor-based text classification performance by using well-estimated parameters [23].

Johnson et al. [24] use the rule simplification method that converts the decision tree into a logically equivalent and the sparsity of text data taken by a decision tree algorithm. Kim, Rim, and Yook [25] deduced that Naïve Bayes is often used in text classification applications and experiments because of its simplicity and effectiveness. Shanahan and Roma [26] mention that the SVM applied to text classification provides excellent precision but poor recall. The research's similarity is that they cover major theoretical issues to guide researchers to exciting research directions. One of the disadvantages of these kinds of research is that they are theory information that was not tested, so they might not work in other types of data or applications.

Machine learning methods have been well harnessed to solve the challenge of crime detection. Studies in [16] applied the combination of ML and computer vision algorithms to improve the accuracy and detailing of crime prediction. The approach depends on data sourced from cameras and microphones in public spaces so that algorithmic solutions based on their hybrid model are then applied to detect crime using a social security number database. Similarly, neural networks have also been applied to crime detection using geo-spatiality to support the detection process [27]. A study in [28] combined the big data technique with an optimised ML algorithm to extract key features suggesting the occurrence of crime, the location, and potential hotspots. Other related works are those in [29] and [30].

Based on the literature reviewed, all traditional NLP algorithms were used to process the data in this research. The SVM and RF classifier were used for crime detection and analysis.

3 Research Methodology

Based on the ideas and approach of the previous research in text classification using NLP algorithms. In this paper, the techniques used for accurate crime detection include pre-processing with traditional NLP algorithms, comparing different ML models, and using the most suitable models that produce accurate results. This section explains in detail the methodology and techniques used to implement the proposed crime detection concepts. More so, Fig. 1 shows the summary of the steps we took.

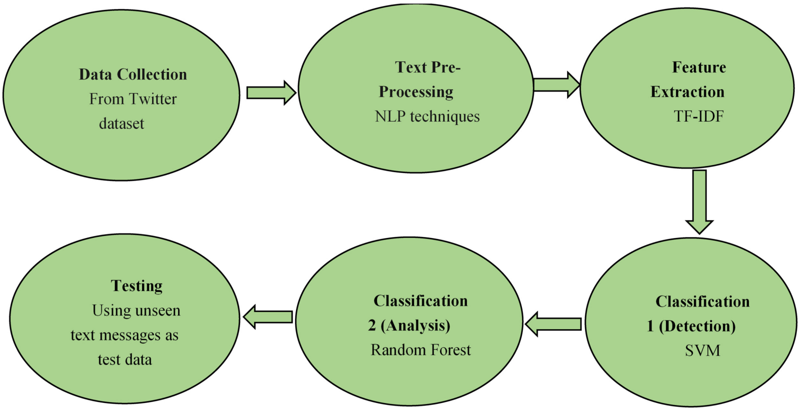


Fig. 1. Summary of the approach

3.1 Dataset

Twitter Spam dataset was used for crime detection. Twitter spam is unwanted content manifesting in many ways, including profanity, insults, hate speech, malicious, link and fraudulent reviews. By closely inspecting the data/tweets in this dataset, we saw that this data is crime-related. Twitter hate speech dataset was used for crime analysis. This data was used to create two more other classes for Attack and Drug- "crime," "gun," and more were used to group attack-related tweets. Keywords such as "drug," "cocaine," "overdose" were used to group drug-related crime messages. Table 1 and Table 2 show the distribution of the texts across different classes. The analysis dataset indicates the imbalance between the classes, but the ML algorithm works well with imbalanced data due to the classifiers applied in this study. Table 3 and Table 4 show the example of the records of the dataset. Moreover, several studies have demonstrated outstanding performances with Twitter datasets when used with ML algorithms [31, 32].

Table 1. Crime twitter dataset (detection dataset)

Class	Texts
Crime	5804
Quality	5983
Total	11787

3.2 Text Pre-processing

Natural Language Learning techniques were used for text cleaning. Text message data contains much noise and no text data like emoji and numbers. The process of data cleaning will include converting emojis to text, converting all characters to lower cases, converting numbers to words, removing punctuations, mentions, white spaces, stop words,

Table 2. Twitter hate speech dataset (analysis dataset)

Class	Texts
Hate speech	5097
Offensive	3245
Attack	2456
Drug	348
Normal	6979
Total	20122

Table 3. Sample of detection dataset

Tweet	Type
'Gun for hire': how Jeff Sessions used his prosecuting power to target Democrats	Crime
I posted a new photo to Facebook http://fb.me/2Be7LiyuJ	Quality

Table 4. Sample of analysis dataset

Tweet	**Label**
I don't think I'm getting my baby them white 9 he has two white j and Nikes not even touched	Normal
@Libre, I am a bit confused coz Chinese ppl cannot access Twitter than how this Ching Chong using it. I think he Pakistani 😂😂😂	Hate speech
All my exes were cute, but they were hoes I guess I only attract fly looking thots 🤔	Offensive
Overthink can kill yourself too <happy>	Attack
common blacked on get em high	Drug

and replacing URLs with the word “URL”. The NLP techniques such as tokenisation, stemming, and word lemmatisation are used to reduce words to their most basic form. Figure 2 shows the summary of data pre-processing.

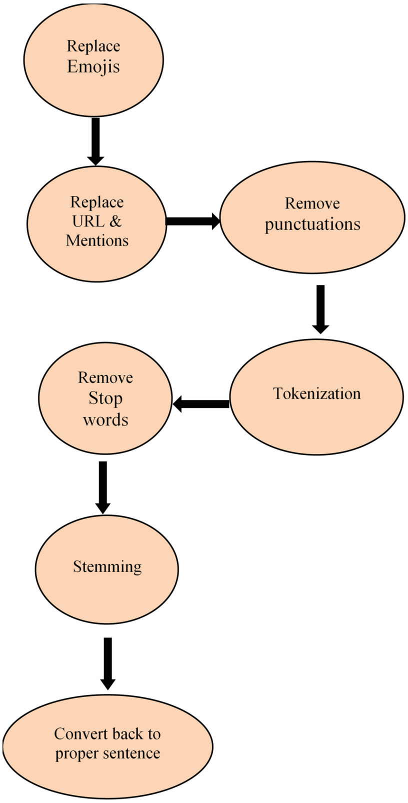


Fig. 2. Data pre-processing steps

3.2.1 Replacing Emojis with Text

We decided not to remove the emojis but to replace them with text because they have meaning and will support detecting the crime.

Example 1: Input: “Take some 💊 they will make you high”
Output: “Take some pill they will make you high.”
Example 2: Input: “Did you 🔥 him?”
Output: “Did you fire him?”

If emojis were removed, sentences one and two would lose meaning, resulting in the training model losing the crime pattern in them. This will result in models yielding negative reports.

3.2.2 Replacing URL and User Mentions

Every URL is replaced with the word "URL", that is because we want to keep track of the text that contains a URL in it so that if a model detects a crime and it's not hate speech, offensive, attack, and drug-related, we can know that that text contains a malicious link. Every mention such as (@XolaniLombo are not removed but replaced with the word "USER" so that the training model can find the pattern that describes certain words said or posted by certain users. This step also includes converting all numbers to word form as they can be used to detect the crime pattern when we know the quantity.

Example 1: Input: "Go to http://MaliciousWebsite.com for"
Output: "Go to URL"
Example 2: Input: "I hate @JamesCordan you should die"
Output: "I hate USER you should die."
Example 3: Input: "We need to be 4 to successfully steal the money"
Output: "We need to be four to successfully steal the money."

Examples 1 and 2 may lose the crime pattern if the URL and the user mention are removed. Although the numbers are not very important in identifying the crime pattern, knowing the quantity (number) associated with certain words can help detect the crime, especially in attack and drug-related texts.

3.2.3 Removing Punctuations

Our focus is only on the texts (words), so punctuations (including hashtags) are unnecessary. That is why we removed all the punctuations in the texts in this step.

Example 1: Input: "How many grams do you want???"
Output: "How many grams do you want"
Example 2: Input: "Can you deliver weed, crack and cocaine, please?"
Output: "Can you deliver weed crack and cocaine please"

This step ensures the characters are shown clearly for the next step, such as tokenisation for easy splitting of the words.

3.2.4 Tokenisation

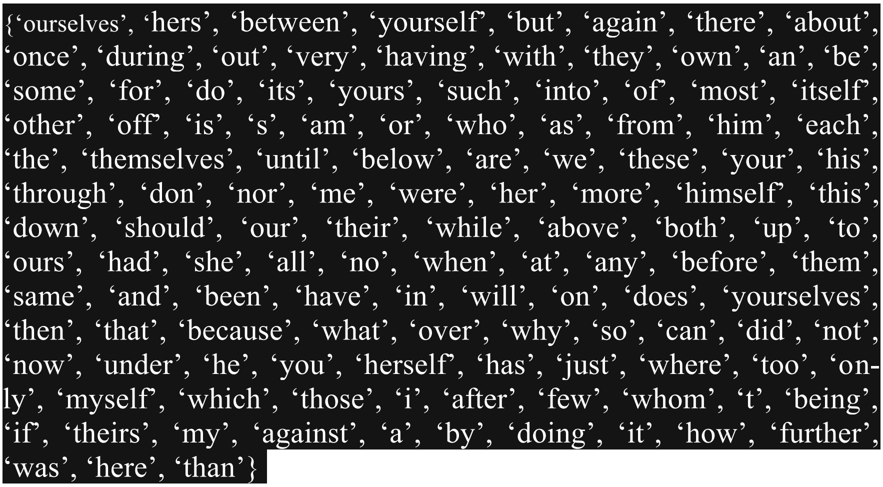
Tokenisation is the essential basic step of NLP, where you identify words that constitute a string of characters. This is important because the meaning of the text could be interpreted by analysing the terms present in the text. This step includes converting all the words to lower cases.

Example 1: Input: "This is fun"
Output: ['this', 'is', 'fun']
Example 2: Input: "Drugs are bad for your health"
Output: ['Drugs', 'are', 'bad', 'for;', 'your', 'health']

This makes it easy for the following steps to perform their function as words are separated nicely.

3.2.5 Removing Stop Words

One of the essential steps in NLP is to filter out useless data. In NLP, useless words (data) are referred to as stop words. Figure 3 shows an example of stop words. This list can be modified by adding words of your choice in English.



{‘ourselves’, ‘hers’, ‘between’, ‘yourself’, ‘but’, ‘again’, ‘there’, ‘about’, ‘once’, ‘during’, ‘out’, ‘very’, ‘having’, ‘with’, ‘they’, ‘own’, ‘an’, ‘be’, ‘some’, ‘for’, ‘do’, ‘its’, ‘yours’, ‘such’, ‘into’, ‘of’, ‘most’, ‘itself’, ‘other’, ‘off’, ‘is’, ‘s’, ‘am’, ‘or’, ‘who’, ‘as’, ‘from’, ‘him’, ‘each’, ‘the’, ‘themselves’, ‘until’, ‘below’, ‘are’, ‘we’, ‘these’, ‘your’, ‘his’, ‘through’, ‘don’, ‘nor’, ‘me’, ‘were’, ‘her’, ‘more’, ‘himself’, ‘this’, ‘down’, ‘should’, ‘our’, ‘their’, ‘while’, ‘above’, ‘both’, ‘up’, ‘to’, ‘ours’, ‘had’, ‘she’, ‘all’, ‘no’, ‘when’, ‘at’, ‘any’, ‘before’, ‘them’, ‘same’, ‘and’, ‘been’, ‘have’, ‘in’, ‘will’, ‘on’, ‘does’, ‘yourselves’, ‘then’, ‘that’, ‘because’, ‘what’, ‘over’, ‘why’, ‘so’, ‘can’, ‘did’, ‘not’, ‘now’, ‘under’, ‘he’, ‘you’, ‘herself’, ‘has’, ‘just’, ‘where’, ‘too’, ‘only’, ‘myself’, ‘which’, ‘those’, ‘i’, ‘after’, ‘few’, ‘whom’, ‘t’, ‘being’, ‘if’, ‘theirs’, ‘my’, ‘against’, ‘a’, ‘by’, ‘doing’, ‘it’, ‘how’, ‘further’, ‘was’, ‘here’, ‘than’}

Fig. 3. Example of stop words

Example 1: Input: [‘this’, should’, ‘stay’, ‘between’, ‘us’]
Output: [‘stay’, ‘us’]
Example 2: Input: [‘he’, ‘killed’, ‘her’, ‘not’, ‘me’]
Output: [‘killed’]

Removing stop words helps remove all the unwanted words because they mostly include pronouns, prepositions, and conjunctions that are unimportant in this scope.

3.2.6 Stemming

Stemming removes the suffix from a word and reduces it to its root word. For example, “Killing” is a word and its suffix “ing”, if “ing” is removed from “Killing”, then we get the root word or base word which is “Kill”. This step helps, so the word like “Killed” and “Killing” be considered as one-word “Kill” to narrow the unique words our model will process. Porter Stemmer algorithm was used for stemming because it is fast, most common, and is a gentle stemmer (Tables 5 and 6).

Example 1: Input: [‘bombing’, ‘execution’, terrorist]
Output: [‘bomb’, ‘execut’, ‘terrorist’]
Porter’s algorithm consists of five phases of word reductions, applied sequentially.

Table 5. Rules of porter stemmer

Rule	Example
SSSES -> SS	Caresses -> Caress
IES -> I	Bullies -> Bulli
SS -> SS	Caress -> Caress
SS ->	Pills -> Pill

Table 6. Few examples of words with their stem

Words	Stem
Killing	Kill
Ponies	Poni
Caress	Caress
Feed	Fe
Sing	Sing
Liked	Like
Laundering	Launder
Troubling	Troubl

3.2.7 Lemmatisation

Lemmatisation is not that different from stemming. In both lemmatisation and stemming, we try to reduce the word to its root. The root in the stemming process is called a stem, and in the lemmatisation process is called a lemma. The difference is that in stemming, a part of the word at the tail end is removed to arrive at the stem of the word, with no understanding of the meaning. In lemmatisation, the algorithm has this knowledge. It is like the algorithm referring to a dictionary to understand the word’s meaning before reducing it to its root word (lemma). The lemmatisation algorithm knows that the word better is derived from the word good, and hence the lemma of better is good. We decided to use both algorithms for better results, first applying the stemming operation followed by lemmatization (Table 7).

Table 7. Few examples of words with their lemma

Words	Lemma
Has	Have
Helped	Help
Better	Good
Am	Be
Relatives	Relative
Study	Study
Studying	Study
Studies	Study

3.2.8 Converting Word List to Sentence

This is the final step of data pre-processing, where the output of the above steps is converted to a proper sentence to prepare for the next step.

Example: Input: [‘student’, ‘study, ‘smart’]
Output: “student study smart”

3.3 Feature Extraction

Machine learning models deal with numbers while we are dealing with text. So, we need to transform and represent the text to numbers, known as text vectorisation, for the classification model to learn. The commonly used algorithm TF-IDF (Term Frequency-Inverse Document Frequency) is used for this study. This statistical measure evaluates how relevant a word is to a document in a collection of documents. The highest-scoring word of a document is most relevant to that document (considered keywords for that document). This is done by multiplying two metrics: how many times a word appears in a document and the inverse document frequency of words across a set of documents. Figure 4 shows the crime word cloud (Table 8).

TF-IDF of a term word in a text:

tf-idf (word, text) = tf (word, text)*idf (word)
tf (word, text) = (frequency of term word in text/number of words in the text)
idf (word) = log [(n+1)/(df(word)+1)] + 1
(n = total number of texts, df(word) = number of texts where the word occurs)

For an example of TF-IDF calculations, suppose you have:

Text A: “killing humans is fun to psychopaths.”
Text B: “killing other humans is bad.”

Results features for the above examples would be:

Fig. 4. Crime word cloud

Table 8. Example of TF-IDF calculations

Word	tf		idf	tf * idf	
	A	B		A	B
Killing	1/6	1/5	Log (3/3) + 1	0.167	0.200
Humans	1/6	1/5	Log (3/3) + 1	0.167	0.200
Is	1/6	1/5	Log (3/3) + 1	0.167	0.200
Fun	1/6	0/5	Log (3/2) + 1	0.196	0
To	1/6	0/5	Log (3/2) + 1	0.196	0
Psychopaths	1/6	0/5	Log (3/2) + 1	0.196	0
Other	0/6	1/5	Log (3/2) + 1	0	0.235
Bad	0/6	1/5	Log (3/2) + 1	0	0.235

[[0.167, 0.167, 0.167, 0.196, 0.196, 0.196, 0.000, 0.000]
[0.200, 0.200, 0.200, 0.000, 0.000, 0.000, 0.235, 0.235]]

4 Results and Discussion

The next step after TF-IDF was to train the models for crime detection to achieve binary classification and crime analysis using multi-class classification. If the accuracy of a model falls within the range of 0%–69%, then such model is considered as bad, 70%–79% good, 80% to 89% excellent and 90%–100% overfitting. For crime detection in text SVM, Naïve Bayesian and KNN classification algorithms were used. Notably, 70% (8251 texts) of data was used to train the models and 30% (3536 texts) for testing. Table 9 shows the classification results obtained.

Table 9. Classification results for crime detection in texts

Model	Accuracy	Precision	Recall
SVM	85.69%	85.85%	85.46%
Naïve Bayesian	60.02%	56.59%	85.92%
KNN	50.21%	73.33%	06.00%

The results show that SVM achieved higher accuracy than other models, which yielded 85.69% accuracy. This is regarded as an excellent accuracy as it shows that the model is not overfitting. Therefore, this model will be used to detect crime in texts. As the dataset for crime analysis in the text was imbalanced, two ML algorithms that are good with imbalance datasets were investigated, RF and Cost-Sensitive SVM for imbalanced data. So, both models ensure all the classes get equal chances of getting to a training model. In this case, 70% (14085 texts) of training data was used to train the models and 30% (6037 texts) for testing. Table 10 shows the classification results obtained.

Table 10. Classification results for crime analysis in texts

Model	Accuracy	Precision	Recall
RF	72.16%	72.16%	72.16%
Cost-Sensitive SVM	69.33%	69.33%	69.33%

The results show that RF achieved higher accuracy than Cost-Sensitive SVM, achieving 72.16%, which is regarded as good accuracy. Therefore, this model was used to analyse texts for the crime. The main challenge encountered in this study is that there is no social media (Twitter) dataset that has its texts classified as a crime. We analysed the data and noticed that data labels as spam are all crime-related to circumvent this. However, this approach can lead to other messages not being detected as a crime. To prevent this, we ensured that the models missed no message.

As the Attack and Drug classes were self-generated based on keywords, not all the texts that are labelled as Attack or Drug may have anything to do with these classes. Other drug or attack-related texts cannot be labelled as drug or attack because the keywords used did not pick them. The same thing in the detection dataset, as the owner regards the label as spam, not crime. Due to a thorough analysis of the dataset, we noted that the labels classified as spam are crime-related. It can happen that other texts that are labelled as spam are not strictly crime-related. The implemented system ensures that it checks for crime-related keywords if they are missed by the models (classified as normal while there is a crime) to detect and analyse crime in text messages accurately.

The implemented system ensures that it checks for crime-related keywords if they are missed by the models (classified as normal while there is a crime) to detect and analyse crime in text messages accurately.

5 Conclusion and Future Work

This paper applied NLP techniques to detect and analyse crime in social media messages. The crime that this paper focused on is hate speech, offensive (lite crime), attack, and drug-related crimes. We used two datasets to detect crime in texts (binary classification) and analyse crime (multi-class classification). We investigated three ML models for crime detection in the text: SVM, Naïve Bayesian, and KNN. It was found that the SVM is the most accurate achieving 85.69% accuracy. In addition, we investigated two good models with imbalanced data for crime analysis in text, RF and Cost-Sensitive SVM classifier. More so, it was also discovered that the RF achieved an accuracy of 72.16%.

This research focused more on normal English text messages. However, if non-English or slang words are absent in a text message, this may negatively impact the learning algorithm. In future work, we would like to investigate the use of NLP in non-English languages and internet slang. An error analysis could also help better understand the challenges of this task, and this could be used to provide insights into the work of NLP algorithms.

References

1. Boba, R.: Introductory guide to crime analysis and mapping. Community Oriented Policing Services, USA (2001)
2. Dlamini, S., Mbambo, C.: Understanding policing of cybe-rcrime in South Africa: the phenomena, challenges and effective responses. Cogent Soc. Sci. **5**(1), 1675404 (2019)
3. SAS: SAS: Machine Learning: What it is and why it matters. https://www.sas.com/en_us/insights/analytics/machine-learning.html. Accessed 27 Apr 2021
4. Salloum, S., Gaber, T., Vadera, S., Shaalan, K.: Phishing email detection using natural language processing techniques: a literature survey. Procedia Comput. Sci. **189**, 19–28 (2021)
5. Guo, W., et al.: Deep natural language processing for search and recommender systems. In: Conference: the 25th ACM SIGKDD International Conference (2019)
6. Chavare, S.R., Awati, C.J., Shirgave, S.K.: Smart recommender system using deep learning. In: 2021 6th International Conference on Inventive Computation Technologies (ICICT) (2021)
7. Chakraoui, M., Elkalay, A., Mouhni, N.: Recommender system for information retrieval using natural language querying interface based in bibliographic research for Naïve users. Int. J. Intell. Sci. **12**(1), 9–20 (2022)
8. Olaide, O., Kana, A.D.: OWL formalization of cases: an improved case-based reasoning in diagnosing and treatment of breast cancer. Int. J. Inf. Secur. Priv. Digit. Forensics (IJIS) **3**(2), 92–105 (2019)
9. Oyelade, O.N., Ezugwu, A.E.: COVID19: a natural language processing and ontology oriented temporal case-based framework for early detection and diagnosis of novel coronavirus. Preprints (2020)

10. Oyelade, A.O.S.J.S.A.O.N.: Patient symptoms elicitation process for breast cancer medical expert systems: a semantic web and natural language parsing approach. Future Comput. Inform. J. **3**(1), 72–81 (2018)
11. Oyelade, O.N., Ezugwu, A.E.: A case-based reasoning framework for early detection and diagnosis of novel coronavirus. Inform. Med. Unlocked **20**, 100395 (2020)
12. Osorio, J., Beltran, A.: Enhancing the detection of criminal organisations in mexico using ML and NLP. In: 2020 International Joint Conference on Neural Networks (IJCNN) (2020)
13. Meira, J., Carneiro, J., Bolón-Canedo, V., Alonso-Betanzos, A., Novais, P., Marreiros, G.: Anomaly detection on natural language processing to improve predictions on tourist preferences. Electronics **11**(5), 779 (2022)
14. Zhang, T., Schoene, A.M., Ji, S., Ananiadou, S.: Natural language processing applied to mental illness detection: a narrative review. NPJ Digital Med. **5**(46) (2022)
15. Wang, M., Xu, L., Guo, L.: Anomaly detection of system logs based on natural language processing and deep learning. In: 2018 4th International Conference on Frontiers of Signal Processing (ICFSP) (2018)
16. Shah, N., Bhagat, N., Shah, M.: Crime forecasting: a machine learning and computer vision approach to crime prediction and prevention. Vis. Comput. Ind. Biomed. Art **4**(1), 1–14 (2021)
17. Bolla, R.A.: Crime Pattern Detection Using Online Social Media. Missouri University of Science and Technology (2014)
18. Sharma, A., Jain, R.: Data pre-processing in spam detection. IJSTE Int. J. Sci. Technol. Eng. **1**(11) (2015)
19. Shirani-Mehr, H.: SMS spam detection using machine learning approach, Stanford University (2013)
20. Malmasi, S., Zampieri, M.: Detecting hate speech in social media, arXiv preprint arXiv:1712.06427 (2017)
21. Andrews, S., Brewster, B., Day, T.: Organised crime and social media: a system for detecting, corroborating and visualising weak signals of organised crime online. Secur. Inform. **7**(1), 1–21 (2018)
22. Ikonomakis, E., Kotsiantis, S., Tampakas, V.: Text classification using machine learning techniques. WSEAS Trans. Comput. **4**(8), 966–974 (2005)
23. Lim, H.S.: Improving KNN based text classification with well estimated parameters. In: Pal, N.R., Kasabov, N., Mudi, R.K., Pal, S., Parui, S.K. (eds.) ICONIP 2004. LNCS, vol. 3316, pp. 516–523. Springer, Heidelberg (2004). https://doi.org/10.1007/978-3-540-30499-9_79
24. Johnson, D.E., Oles, F.J., Zhang, T., Goetz, T.: A decision-tree-based symbolic rule induction system for text categorization. IBM Syst. J. **41**(3), 428–437 (2002)
25. Kim, S.-B., Rim, H.-C., Yook, D., Lim, H.-S.: Effective methods for improving naive bayes text classifiers. In: Ishizuka, M., Sattar, A. (eds.) PRICAI 2002. LNCS (LNAI), vol. 2417, pp. 414–423. Springer, Heidelberg (2002). https://doi.org/10.1007/3-540-45683-X_45
26. Shanahan, J.G., Roma, N.: Improving SVM text classification performance through threshold adjustment. In: Lavrač, N., Gamberger, D., Blockeel, H., Todorovski, L. (eds.) ECML 2003. LNCS (LNAI), vol. 2837, pp. 361–372. Springer, Heidelberg (2003). https://doi.org/10.1007/978-3-540-39857-8_33
27. Walczak, S.: Predicting crime and other uses of neural networks in police decision making. Front. Psychol. **12** (2021)
28. Palanivinayagam, A., Gopal, S.S., Bhattacharya, S., Anumbe, N., Ibeke, E., Biamba, C.: An optimised machine learning and big data approach to crime detection. Wirel. Commun. Mob. Comput. **2021** (2021)
29. Bharati, A., Sarvanaguru, R.A.K.: Crime prediction and analysis using machine learning. Int. Res. J. Eng. Technol. (2018)

30. Navalgund, U.V., Priyadharshini, K.: Crime intention detection system using deep learning. In: 2018 International Conference on Circuits and Systems in Digital Enterprise Technology (ICCSDET) (2018)
31. Rodrigues, A.P., Fernandes, R., Shetty, A., Lakshmanna, K., Shafi, R.M.: Real-time twitter spam detection and sentiment analysis using machine learning and deep learning techniques. Comput. Intell. Neurosci. (2022)
32. Yadav, N., Kudale, O., Gupta, S., Rao, A., Shitole, A.: Twitter sentiment analysis using supervised machine learning. In: Hemanth, J., Bestak, R., Chen, J.I.Z. (eds.) Intelligent Data Communication Technologies and Internet of Things. Springer, Singapore (2020). https://doi.org/10.1007/978-981-15-9509-7_51