Enhancing Security in SMS by combining NLP models using Ensemble Learning for Spam Detection with Image Steganography Integration

Aditya Kumar
Department of Networking and Communications
SRM Institute of Science and Technology
Kattankulathur, Chennai, India
ak8476@srmist.edu.in

Fancy C.
Department of Networking and Communications
SRM Institute of Science and Technology
Kattankulathur, Chennai, India
fancyc@srmist.edu.in

***Abstract*— Spam SMS messages are a prevalent problem in today's world and have become a source of annoyance for users. This research paper proposes a novel approach to detect SMS spam using Natural Language Processing (NLP) and creating multiple models and apply ensemble learning on them. The proposed method includes pre-processing the data with NLP techniques such as stopwords removal, stemming, and lemmatizing, extracting relevant features from the text data, and transforming it into numerical representations. The performance of the proposed method was evaluated on a real-world dataset and compared to traditional machine learning algorithms such as Naïve Bayes, SVM. The multiple classifiers are trained on the transformed data, and an Ensemble Learning algorithm is applied to combine their predictions to obtain a more accurate result. The resulting model KNR gave higher accuracy than traditional models. The custom model was integrated into a custom Image Steganography tool which can hide textual data into an image using 1 bit Lease Significant Bit (LSB) technique and while decrypting a hidden message, the custom model would detect if the textual data is spam or not. The purpose of the Image Steganography tool is to provide users with a secure method of communication. For example, sender can hide sensitive data they wish to send to the receiver by hiding it inside an image and the receiver decrypting it to get the original message. If a malicious actor finds out about this communication method and tried to send spam links or messages by hiding it in an image and sending it to its victim, the custom spam detection model will detect it.**

Keywords—Natural Language Processing, Ensemble Learning, Spam Detection, Image Steganography.

I. Introduction

The proliferation of mobile phones has brought about a new form of communication, Short Message Service (SMS), which has become an essential aspect of daily life. However, with the growth of SMS as a medium of communication has come the rise of spam messages, which are unsolicited and often unwanted. These spam messages not only cause inconvenience for users but can also lead to privacy concerns and financial loss .To address this problem, various machine learning algorithms have been proposed for SMS spam detection, including multiple types of Naïve Bayes. However, these methods have limitations and do not provide adequate performance in terms of accuracy and speed. To overcome these limitations, this research paper proposes a new method for SMS spam detection using NLP and creating a new model using Ensemble methods. NLP techniques will be used to pre-process the data and extract relevant features, while Ensemble Learning will be used to combine the predictions of multiple classifiers to produce a more accurate result. The custom model will be integrated into an Image Steganography tool. While revealing the hidden message in the image, the custom ensemble model will process the hidden text and detect if the message is spam or legitimate. This will provide a layer of security in SMS applications as if a malicious actor discovers this communication technique and tries sending a spam text, the custom model will detect it.

II. Related Work

Rule Based filtering was the earliest phase of spam detection [1]. The rules were typically based on specific keywords, phrases, or patterns that were commonly found in spam messages. This approach was effective in the early days of spam, when most spam messages followed a predictable pattern and used similar language and formatting. Some focused on specific words [2] while some focused on If-Then patterns [3]. However, it was limited by its inability to keep up with the rapidly changing nature of spam messages. Spammers quickly learned to adapt to rule-based filters by changing the wording, formatting, and delivery method of their messages, making it difficult for filters to keep up. The next technique, Context-based filtering takes into account the context in which a message is received and analyzed to determine its likelihood of being spam. This approach considers a range of factors beyond just the content of the message itself, such as the sender's reputation, the content of previous messages from the same sender, and the behavior of the recipient in responding to similar messages [4]. Some used relation of rates of one type of word to another [5] and some used Bayesian Filtering [6]. Building upon Context based filtering, Feature Extraction would train the models based on the features of messages from the dataset such as length of words, the occurrence of specific words, and so on [7]. Gini Index and Information Gain was another method [8] while cleaning the dataset and focusing on less features proved viable as well [9] The hybrid approach to spam detection were combinations of rule-based filters and machine learning algorithms such as NB and SVM. The rule-based filter may be used to quickly and accurately identify messages that match certain predefined criteria, such as messages containing specific keywords or originating from known spam domains [10]. The machine learning algorithm can then be used to classify the remaining messages based on their content and other features. They were also implemented with feature extraction technique [11] and with coupling supervised and unsupervised learning [12].

III. Methodology

A. *Architecture of the proposed methodology*

The dataset would be first cleaned up and preprocessed. Exploratory Data Analysis would give insight on the dataset. Based of EDA, features will be extracted so that they can act as more data points for the AI models. The text messages will be vectorized and algorithmic models will be created and their performance will be evaluated based on accuracy and precision. Best performing models will be selected to create a custom model using Ensemble Learning. An Image Steganography Tool will be created which will use 1-bit LSB technique for hiding text data in images. The obtained model will be implemented with the Steganography tool and will detect for spam while decrypting hidden messages from images. The architecture of the proposed methodology is shown in Figure 3.1.1.

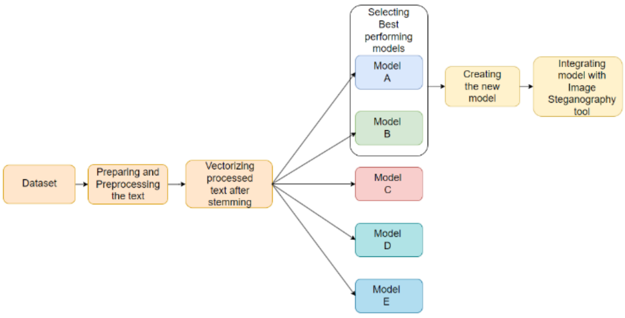

Dataset
Preparing and Preprocessing the text
Vectorizing processed text after stemming
Selecting Best performing models
Model A
Model B
Model C
Model D
Model E
Creating the new model
Integrating model with Image Steganography tool

Figure 3.1.1

B. *Exploratory Data Analysis*

The dataset was taken from UCI Machine Learning from Kaggle.com and it consisted of 5574 text messages marked as either spam or ham i.e. legitimate. 418 duplicates were found and after cleaning the dataset, 5156 remained in which 12.66% messages were spam and the remaining 87.34% were ham as shown in Figure 3.2.1.

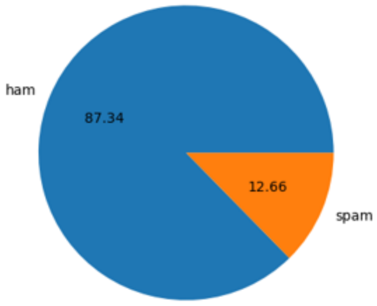


Figure 3.2.1

Features were extracted such as Count of Number of Characters, Count of Number of Words and Count of Number of Sentences in Figure 3.2.2, Figure 3.2.3 and Figure 3.2.4 respectively. Legitimate labelled data is shown in blue and spam labelled data is shown in red.

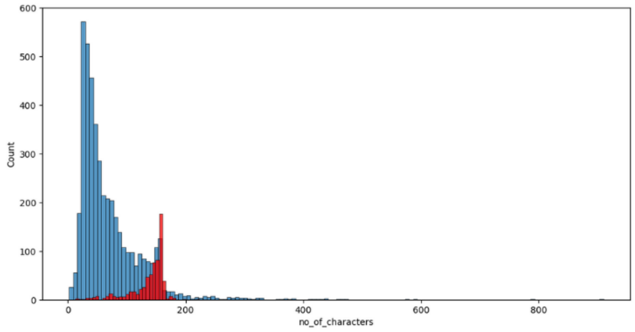


Figure 3.2.2

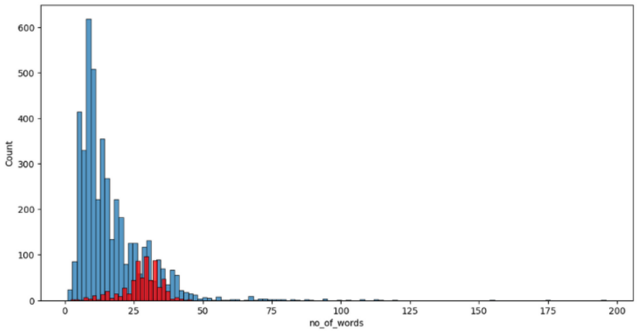


Figure 3.2.3

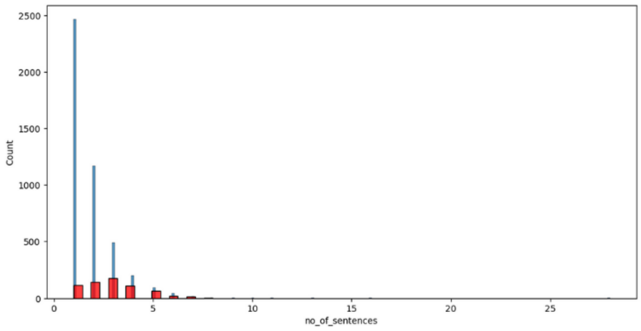


Figure 3.2.3

The above graphs shows that legitimate messages are shorter than spam messages. This data was extracted as separate fields and would affect the performance of the models.

C. *Data Preprocessing*

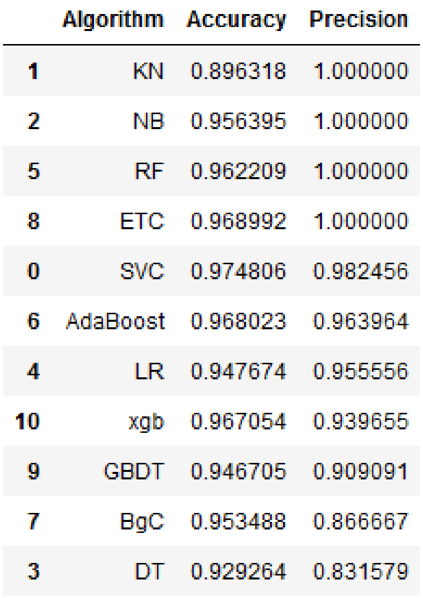
The text data was converted to lowercase to remove different tokenization of same words with initial letter capitalized. It was then tokenized and all special characters were removed. Stopwords were removed to remove unnecessary words from the dataset and the remaining unique words were then stemmed into their base word. The preprocessed data was then vectorized into numbers using Term Inverse – Document Inverse Frequency algorithm (TF-IDF) and was scaled between 0 and 1 for calculation of probability in AI models.

D. *Model Building*

The algorithms chosen to build AI models were chosen on the basis of their proved results. Along with those algorithms, some existing ensemble algorithms were also chosen to compare the performances of both types of algorithms based on their accuracy and precision. The chosen algorithms are as follows:

- Support Vector Machines (SVC)
- K Nearest Neighbors (KN)
- Naïve Bayes (NB)
- Decision Trees (DT)
- Logistic Regression (LR)
- AdaBoost Classifier (AdaBoost)
- Bagging Classifier (BgC)
- Extra Trees Classifier (ETC)
- Gradient Boosting Classifier (GBDT)
- Extreme Gradient Boosting Classifier (xGB)

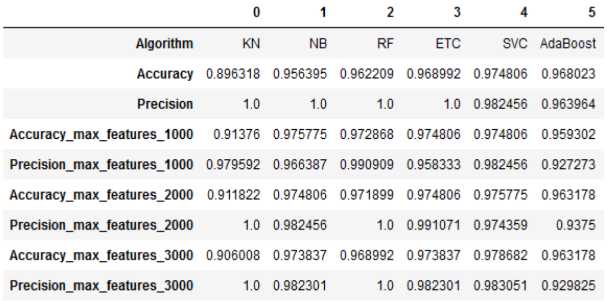
After training the models it was found that the best performing models were K Nearest Neighbors, Multinomial Naïve Bayes and Random Forest classifiers as shown in Figure 3.4.1.



	Algorithm	Accuracy	Precision
1	KN	0.896318	1.000000
2	NB	0.956395	1.000000
5	RF	0.962209	1.000000
8	ETC	0.968992	1.000000
0	SVC	0.974806	0.982456
6	AdaBoost	0.968023	0.963964
4	LR	0.947674	0.955556
10	xgb	0.967054	0.939655
9	GBDT	0.946705	0.909091
7	BgC	0.953488	0.866667
3	DT	0.929264	0.831579

Figure 3.4.1

The following models were optimized by changing the limit of max features i.e. number of unique words taken into consideration in model building from the corpus. After optimization, the results were as shown in Figure 3.4.2.



	0	1	2	3	4	5
Algorithm	KN	NB	RF	ETC	SVC	AdaBoost
Accuracy	0.896318	0.956395	0.962209	0.968992	0.974806	0.968023
Precision	1.0	1.0	1.0	1.0	0.982456	0.963964
Accuracy_max_features_1000	0.91376	0.975775	0.972868	0.974806	0.974806	0.959302
Precision_max_features_1000	0.979592	0.966387	0.990909	0.958333	0.982456	0.927273
Accuracy_max_features_2000	0.911822	0.974806	0.971899	0.974806	0.975775	0.963178
Precision_max_features_2000	1.0	0.982456	1.0	0.991071	0.974359	0.9375
Accuracy_max_features_3000	0.906008	0.973837	0.968992	0.973837	0.978682	0.963178
Precision_max_features_3000	1.0	0.982301	1.0	0.982301	0.983051	0.929825

Figure 3.4.2

E. Custom Ensemble Model

The models chosen for Custom Ensemble Model were K Nearest Neighbors with max features set at 1000, Multinomial Naïve Bayes with max features set at 2000 and Random Forest with max features set at 3000. Naïve Bayes model was chosen as base estimator and Stacking technique was used to create the custom model dubbed KNR and it resulted in 97.9% accuracy and 95.57% precision.

F. Custom Image Steganography Tool

The custom Image Steganography tool was created using 1 bit Least Significant Bit technique for hiding text in images. It has Graphical User Interface with a display for selected image preview and one for writing and displaying the text message to be hidden/revealed respectively. With four buttons for opening an image, saving an image after hiding text, hiding the text and revealing the text. The custom ensemble model is integrated with the show button such that when the hidden text is decrypted, it can detect if the said text is spam or not as shown in Figure 3.6.1.

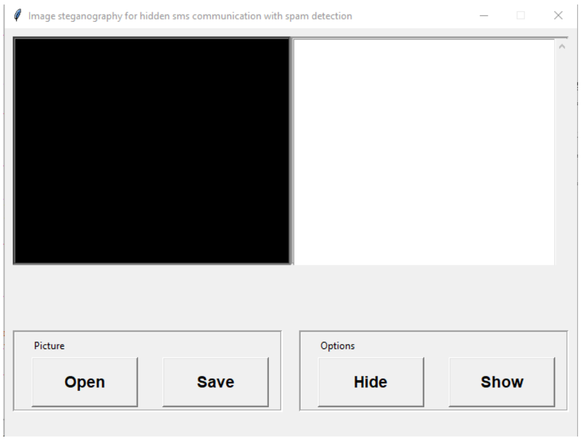


Image steganography for hidden sms communication with spam detection
Picture
Open
Save
Options
Hide
Show

Figure 3.6.1

Figure 3.6.2 and Figure 3.6.3 show some examples of text detected as legitimate and spam respectively.

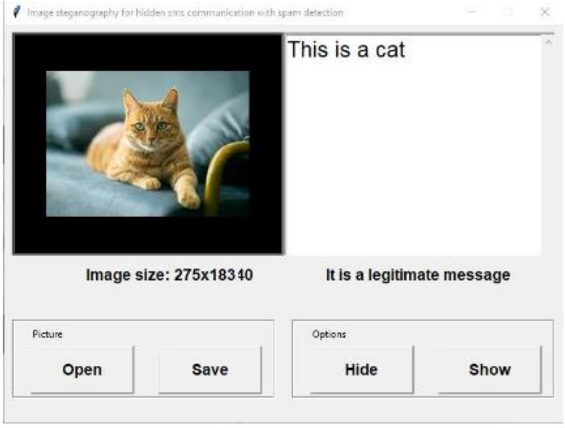


Image steganography for hidden sms communication with spam detection

This is a cat

Image size: 275x18340

It is a legitimate message

Picture
Open
Save
Options
Hide
Show

Figure 3.6.2

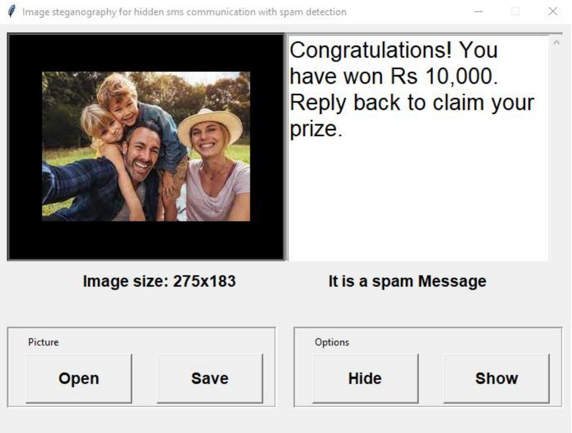


Image steganography for hidden sms communication with spam detection
Picture
Open
Save
Options
Hide
Show

Congratulations! You have won Rs 10,000. Reply back to claim your prize.

Image size: 275x183

It is a spam Message

Figure 3.6.3.

IV. Conclusion and Future Improvements

The custom ensemble model KNR was created using Multinomial Naïve Bayes, K Nearest Neighbors and Random Forest classifiers with using Stacking Ensemble technique has 97.9% accuracy and 95.57% precision dubbed as KNR model. The models individually had 100% precision but then it would imply that its completely detecting the test spam texts from the dataset but it is possible that the model can label even legitimate messages as spam so having some leeway is preferable. As for the accuracy of the individual models with respect to the custom model, the custom model gives 7% higher accuracy than KN, 0.5% higher than NB and 1% higher accuracy than RF models when max features is set to 3000. As for the Image Steganography tool, the tool accepts .jpg and .png format images and is able to hide text successfully without any loss of the hidden text, and when a steganographed image is decrypted, the original message is the same as the hidden message and the spam detector module of the tool is able to successfully label if the text is spam or legitimate and this tool is lightweight and easy to use and is compatible with proprietary and private messaging applications. This satisfies the overall objective of the research.

This paper endeavors to provide a comparative study between different types of algorithmic models for SMS Spam Detection and aims to contribute in additional methods of security by introducing Image Steganography that can be integrated in existing SMS applications on mobile devices. 1 bit LSB technique was chosen for this tool as there was no limit in dimensions that could be used in the tool but it is possible to use other types of techniques such as Spread Spectrum as long as the compression of image in transit does not get corrupted.

References

[1] Jain, A. K., & Gupta, B. B. (2018). Rule-based framework for detection of smishing messages in mobile environment. Procedia Computer Science, 125, 617-623.

[2] Xia, T. (2020). A constant time complexity spam detection algorithm for boosting throughput on rule-based filtering systems. IEEE Access, 8, 82653-82661.

[3] Foozy, M., Feresa, C., Ahmad, R., & Abdollah, M. F. (2014). A practical rule based technique by splitting SMS phishing from SMS spam for better accuracy in mobile device. International Review on Computers and Software, 9(10), 1776-1782

[4] Karami, A., & Zhou, L. (2014). Improving static SMS spam detection by using new content-based features.

[5] Gómez Hidalgo, J. M., Bringas, G. C., Sánz, E. P., & García, F. C. (2006, October). Content based SMS spam filtering. In Proceedings of the 2006 ACM symposium on Document engineering (pp. 107-114).

[6] Roy, P. K., Singh, J. P., & Banerjee, S. (2020). Deep learning to filter SMS Spam. Future Generation Computer Systems, 102, 524-533.53

[7] Sharaff, A. (2019). Spam detection in SMS based on feature selection Techniques. In Emerging Technologies in Data Mining and Information Security: Proceedings of IEMIS 2018, Volume 2 (pp. 555-563). Springer Singapore.

[8] Uysal, A. K., Gunal, S., Ergin, S., & Gunal, E. S. (2013). The impact of feature extraction and selection on SMS spam filtering. Elektronika ir Elektrotechnika, 19(5), 67-72.

[9] Zainal, K., & Jali, M. Z. (2016). A review of feature extraction optimization in SMS spam messages classification. In Soft Computing in Data Science: Second International Conference, SCDS 2016, Kuala Lumpur, Malaysia, September 21-22, 2016, Proceedings 2 (pp. 158-170). Springer Singapore.

[10] Senthil Murugan, N., & Usha Devi, G. (2018). Detecting streaming of Twitter spam using hybrid method. Wireless Personal Communications, 103, 1353-1374.

[11] Ghourabi, A., Mahmood, M. A., & Alzubi, Q. M. (2020). A hybrid CNN- LSTM model for SMS spam detection in arabic and english messages. Future Internet, 12(9), 156.

[12] Baaqeel, H., & Zagrouba, R. (2020, November). Hybrid SMS spam filtering system using machine learning techniques. In 2020 21st International Arab Conference on Information Technology (ACIT) (pp. 1-8). IEEE.