2024 8th International Conference on Computing, Communication, Control and Automation (ICCUBEA) | 979-8-3503-9177-0/24/$31.00 ©2024 IEEE | DOI: 10.1109/ICCUBEA61740.2024.10774959

Spam SMS Detection Using Natural Language Processing

Dr. Satpalsing D. Rajput
Department of Computer Engineering
Vishwakarma Institute of Technology
Pune, Maharashtra
satpalsing.rajput@vit.edu

Pratiksha Chopade
Department of Computer Engineering
Vishwakarma Institute of Technology
Pune, Maharashtra
pratiksha.chopade21@vit.edu

Atharva Chivate
Department of Computer Engineering
Vishwakarma Institute of Technology
Pune, Maharashtra
atharva.chivate21@vit.edu

Shreeshail Chitpur
Department of Computer Engineering
Vishwakarma Institute of Technology
Pune, Maharashtra
shreeshail.chitpur21@vit.edu

Isha Dashetwar
Department of Computer Engineering
Vishwakarma Institute of Technology
Pune, Maharashtra
isha.dashetwar21@vit.edu

***Abstract*— A spam SMS is an unsolicited text message dispatched to a large population, illegitimately, without the authorization of the recipients. Reports suggest that the number of Spam SMS received by citizens of any country is approximately 400 million, and this number is increasing day by day. These SMS messages may contain malicious links, causing security threats including phishing attacks and malware infections. Some spam SMS messages can involve fraudulent schemes, leading to financial losses for the receiver and can also cause privacy threats. Hence, there is an urge to detect these spam SMSs correctly and avoid the severe consequences for the recipients.**

The proposed study emphasizes the development of Natural Language Processing based Machine Learning models to identify spam SMS texts. The process involves tokenization of the SMS texts, stemming of the words, and many more abstractions of the keywords for easier classification. Firstly, the text messages are processed using NLP and then the processed data is trained on different ML algorithms. Multiple ML models are created and tested on various kinds of SMS messages for the detection of spam messages. This expert system ensures accurate and precise detection of received messages. This expert system intends to reduce the threats and safeguard the security and privacy of the consumers, by essentially recognizing and steering clear of spam SMS messages. This system aims to empower users with the tools and knowledge needed to navigate the digital landscape safely and securely.

Keywords*— (Spam SMS Detection, Extra Tree Classification, Natural Language Processing, Machine Learning, Stemming)

I. Introduction

In the contemporary era, Short Message Service (SMS) stands as a cornerstone of modern communication, seamlessly integrating into daily interactions across diverse sectors. SMS is now more than just a means of communication; it may be used for banking updates, marketing campaigns, or the distribution of agricultural information. It is a versatile instrument for commerce and connectivity. But even this widely used medium may be exploited; spam SMS is becoming more and more common, which presents serious problems for people all over the world. Unwanted messages destroy the credibility and usefulness of SMS as a communication medium by flooding inboxes with unsolicited content, which can range from promotional materials to potentially hazardous links.

The incessant barrage of spam SMS highlights the urgent need for reliable methods to discern between HAMs (harmless messages) and spam content. Because SMS is so widely used as a medium for advertising and is accessible to a wide range of users, it is a desirable target for malicious actors looking to compromise user security by infiltrating devices. In light of these difficulties, it becomes vitally necessary to create efficient systems that can determine the veracity and intention of incoming signals. This project aims to apply machine learning techniques to analyze message content in response to these imperatives. This will allow for automated classification of SMS as spam or legitimate, which will increase user confidence and security in SMS communication.

Moreover, as we explore the domain of protecting digital communication channels, the incorporation of cutting-edge technologies like Natural Language Processing (NLP) and Machine Learning becomes crucial. NLP is a branch of artificial intelligence that bridges the gap between algorithmic analysis and human communication patterns by enabling systems to understand and process human language. Through endowing systems with the ability to interpret language, natural language processing (NLP) amplifies the effectiveness of machine-learning-based decision-making procedures, permitting more sophisticated and precise detection of spam material in text messages. Therefore, this research aims to strengthen SMS communication's resilience against the ubiquitous threat of spam by utilizing the synergy of machine learning and natural language processing (NLP). This will ensure the integrity and dependability of this widely used medium in the digital age.

II. Literature Review

The study by Bollam Pragna et al. delves into the classification of spam and ham messages on mobile devices using NLP techniques. By employing methods like tokenization and stemming, they convert text data into machine-readable formats, facilitating message categorization. Various machine learning algorithms, including KNN, Decision Tree, and SVM, were evaluated on the 'SMS Spam Collection' dataset, with SVM achieving the highest accuracy of 98.49%. The research explores diverse methodologies for spam detection, ranging from machine learning to rule-based systems, offering a promising solution for efficient inbox management. Future research endeavors may further enhance performance by integrating advanced NLP techniques to better comprehend contextual nuances, thereby improving accuracy.[1] P. Gopala Krishna et al. address the contemporary issue of spam messages, particularly concerning mobile phones. They advocate for the utilization of machine learning to detect these messages, employing methods such as Naive Bayes, KNN, and logistic regression to develop a robust model. Their aim is to create a model capable of effectively distinguishing spam messages from regular ones, thereby enhancing information organization and ensuring data security. The research underscores the significance of mitigating spam emails, which constitute a substantial portion of all emails sent. Through rigorous testing of various methods, the goal is to devise strategies to combat spam emails effectively and safeguard users' communication channels.[2]

Thanniru Lakshman et al. focus on identifying spam and ham messages using supervised machine learning algorithms, particularly Random Forest Classifier and Logistic Regression. They utilize preprocessing techniques with the NLTK library, such as stemming algorithms and tokenizing, on the dataset. Results reveal that Random Forest outperforms Logistic Regression, achieving an accuracy of 97%, underscoring the importance of spam detection in mobile communication and the challenges in accurately identifying spam. The study proposes a machine learning-based approach for spam identification, aiming to enhance effectiveness and precision.[3] Luo GuangJun et al. emphasize the urgent need for accurate spam detection in mobile message communication. They propose a machine learning-based approach employing classifiers like Logistic Regression, K-nearest neighbor, and Decision Tree for message classification. Utilizing the SMS spam collection dataset, their experiments demonstrate that Logistic Regression exhibits superior classification performance compared to other methods, achieving an impressive accuracy rate of 99%. The proposed method outperforms existing techniques, showcasing its potential as an effective solution for combating SMS spam.[4]

Pumrapee Poomka et al. aim to create an innovative SMS spam detection system using NLP and Deep Learning methodologies. They introduce deep learning approaches tailored for SMS spam detection, incorporating NLP techniques for data pre-processing and model construction. The model, built upon LSTM and GRU algorithms, outperforms traditional machine learning algorithms, achieving an overall accuracy rate of 98.18%. Their research addresses the existing gap in the field by providing specialized methods for SMS spam detection, which could significantly enhance spam detection systems.[5] Tian Xia and Xuemin Chen propose a method that utilizes a discrete HMM to incorporate word order information, effectively addressing the low-term frequency issue in SMS spam detection. Their approach is language-agnostic and accurately identifies spam in both English and Chinese SMS datasets. Hidden Markov Models have been successfully applied in various NLP tasks, and their integration in spam detection showcases promising results.[6]

Sridevi Gadde et al. underscore the importance of preprocessing techniques in preparing data for spam detection, such as removing special symbols and converting text to lowercase. They compare different word embedding techniques and utilize sampling techniques like SMOTE to improve model accuracy. The LSTM model achieves the highest accuracy of 98.5% in spam detection, outperforming previous models.[7] Samadhan Nagre conducts a systematic literature review analyzing existing techniques and approaches used in SMS spam detection. The review identifies the advantages and disadvantages of different algorithms, evaluation measures, and datasets. However, it highlights the lack of focus on local content and shortened words in SMS spam detection, suggesting significant scope for future research in this area.[8]

A. S. Sodiya et al. critically review existing SMS spam filtering approaches, identifying their limitations such as adaptability to spammers' concept drift and computational robustness. They construct a taxonomy for existing SMS spam filtering techniques and recommend the use of adaptive and collaborative SMS spam filtering systems.[9] Soumyabrata Saha et al. present a study on SMS spam detection using machine learning techniques. Their model achieves promising results in classifying SMS messages as spam or non-spam, highlighting the efficacy of machine learning in combating spam messages and improving user experience in mobile communication channels.[10]

III. Methodology

Accurate identification of spam text messages helps shield users from phishing and scams, protecting their safety and privacy. It was imperative to design a system that could clearly identify spam text messages based on the research done on the current systems. All SMS texts are filtered by this expert system, which also abstracts keywords that are employed in the detection process.

A. About the Dataset

The SMS Spam Collection Dataset is the dataset that is utilized to identify spam texts. The study has two independent variables: text messages, categorized as spam and non-spam, and a dependent variable that indicates if a message is considered spam or ham. A text message that is categorized as "ham" is an ordinary, non-spam communication.

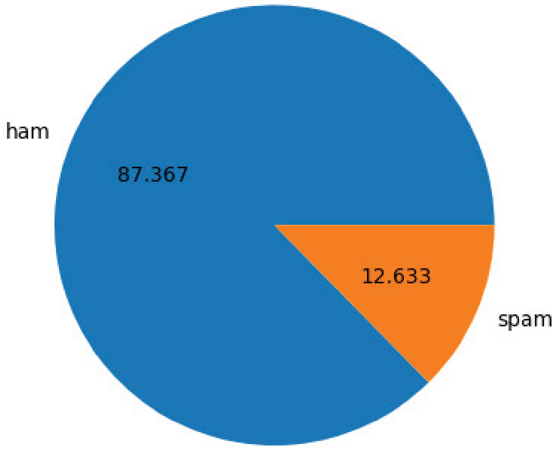


Fig. 1. Pie chart of the dataset indicating its dependent variables

The total distribution of dependent variables in the dataset is shown in Fig. 1. However, pre-processing is necessary due to several inconsistencies in the dataset to effectively isolate meaningful text and apply machine learning techniques.

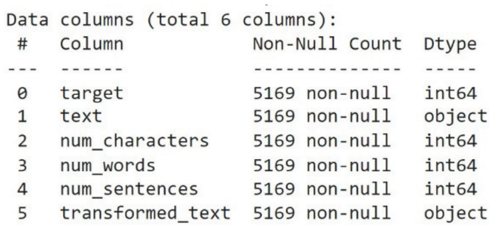
B. Data Preprocessing:

Firstly, the dataset had three unnamed columns, which were dropped from the dataset. Initially, the labels were of string datatype, which are converted to integer datatype. This type of conversion is necessary to enhance the performance of the entire system. The conversion is done through binary encoding, a combination of hash encoding and one-hot encoding. This generates a new binary dependent variable, labeled as ‘target’. Target has binary values – 0 (ham or non-spam messages) and 1 (spam messages). Additionally, the number of sentences, words, and characters present in the dataset are computed.

The dataset is further processed using the Natural Language Toolkit. This involves, in total, five steps.

1. Conversion of sentences or the input text to lowercase.
2. Tokenization: It is the process of transforming a sequence of text into smaller parts, known as tokens. This is done using the inbuilt ‘word_tokenize()’ function in the NLP library.
3. Removing special characters (all characters other than alphanumeric) from the input string.
4. Removing Stop words and punctuation: Stop words are certain commonly used words in a language. For example, “the,” “is,” “and,” etc. are stop words. All these stop words are removed from the text messages.
5. Stemming: Finally, the text is stemmed, which is the process of reducing a word to its root form or word stem. For example, stemming will reduce the words “running”, “runner” and “runs” to their word stem “run”.

After the preprocessing of the text messages, a new column is generated which stores the transformed text of all the input values, which are preprocessed and ready for further implementation. For example, this preprocessing transforms the text message “I HAVE A DATE ON SUNDAY WITH WILL” to “date sunday”. This text can easily be processed and machine learning algorithms can be implemented with more precise and better performance. The dataset has now been modified and appears as shown below in Fig. 2.



```
Data columns (total 6 columns):
 #   Column            Non-Null Count  Dtype 
---  ------            --------------  ----- 
 0   target            5169 non-null   int64 
 1   text              5169 non-null   object
 2   num_characters    5169 non-null   int64 
 3   num_words         5169 non-null   int64 
 4   num_sentences     5169 non-null   int64 
 5   transformed_text  5169 non-null   object
```

Fig. 2. Modified dataset with new target variable.

C. *Algorithms Used:*

1. Logistic Regression: Logistic Regression is employed when the outcome variable is binary, such as true or false, spam or not-spam, 0 or 1, etc., rather than predicting a continuous variable like size. Instead of attempting to fit a linear model to the data, logistic regression fits a sigmoidal logistic function.

2. Support Vector Machine: A Support Vector Machine (SVM) is a supervised learning method utilized primarily for non-linear classification tasks. It operates by identifying the optimal hyperplane that maximizes the margin between the nearest data points of different classes. The dimensionality of the input data determines whether this hyperplane appears as a line in a two-dimensional space or a plane in a higher-dimensional space. By maximizing the margin between points, SVM can effectively discern the most suitable decision boundary between classes, even when multiple hyperplanes could potentially separate them.

3. Random Forest Classification: Random Forest is a classifier ensemble employing numerous decision tree models, suitable for classification and regression assignments. It introduces extra randomness during tree growth, unlike merely seeking the most significant feature for node division, by exploring the optimal feature within a random subset. This approach fosters extensive diversity, typically enhancing model performance.

4. Extra Trees Classification: Extra Trees, abbreviated from extremely randomized trees, is a supervised ensemble learning technique employing decision trees. It's incorporated within the Train Using AutoML tool. Serving as an upgraded variant of random forest, it employs a random sampling technique without replacement for each tree, generating distinct datasets. Rather than computing locally optimal values using Gini or entropy for data partitioning, this algorithm randomly chooses split values. This approach fosters diversified and uncorrelated trees.

5. Gradient Boosting Classification: Gradient boosting is an algorithmic technique that leverages multiple weaker models to construct a significantly more robust and accurate final model. It operates on the principle of ensemble learning, enabling the optimization of various differentiable loss functions and progressively building an additive model in a forward, iterative fashion.

6. AdaBoost Classification: Adaptive Boosting, or AdaBoost for short, uses a series of weak models after assigning equal weights to each training observation. It emphasizes error correction by giving misclassified observations more weight. ADA Boost is an algorithm that improves the accuracy of misclassified discoveries and overall iteration outcomes by integrating the effects of decision boundaries gained from several iterations and combining results from several weak models.

7. XGBoost Classification: The extreme Gradient Boosting algorithm iteratively constructs an ensemble of models, with each subsequent model emphasizing the errors of its predecessors. Employing decision trees as base learners, it aggregates numerous weak learners to create a powerful learner. The ultimate prediction is

determined by a weighted sum of all tree predictions. Its purpose is to mitigate bias and prevent underfitting.

D. Implementation

After the preprocessing of the dataset, the dataset is modeled, trained, and tested on altogether seven machine learning algorithms. The main objective of the system was to build machine learning models that can precisely identify the SMS message as spam or not spam.

This process was initiated with a basic ML algorithm, Logistic Regression. As the dataset is comparatively smaller, the solver function is set to 'liblinear.' The regularization penalty is also used to perform the process of feature selection very effectively. Secondly, a Support Vector Machine was implemented. The kernel type used was sigmoid because the data was non-linear binary classification. Gamma, the single training example, was set to 1, indicating the influence of all the parameters was relatively high.

For the implementation of the Random Forest Classifier, the parameter for the number of decision trees requirements for model training, 'n_estimators' is set to 50. Additionally, a specific random state is also assigned to ensure the reproducibility of the outputs. Extra Trees Classifier, which is an extended version of Random Forest is also deployed. This algorithm assigns arbitrary inception values to all the input parameters, unlike the random forest.

Furthermore, all the ensemble learning techniques, Gradient Boosting Classifier, AdaBoost Classifier, and XGBoost Classifier are built on the same lines, retaining the 'n_estimators' values to 50 and 'random_state' seed value to 2. However, these algorithms differ in the process and modeling, and training. Gradient Boosting builds a strong ML model by gradually enumerating weak learners, and decision trees, to enhance the accuracy and precision. Adaboost Classifier trains the weak learners by adjusting the weights of the misclassified input strings and transforming them into strong learners. XGBoost is an adaptable machine learning algorithm that continuously performs modeling of datasets to enhance scalability and performance.

The above seven models were built and further experimented on the testing dataset. The results obtained are very impressive and are discussed succeeding section.

IV. Results and Discussion

Spam SMS detection was implemented successfully using the Natural Language Processing and Machine Learning libraries. The deployed models were tested on various unknown variables and their accuracy and precision values were computed. Table 1. displays the results of all seven ML models along with their evaluation parameters.

The accuracy values conclude that SVM and Extra Trees Classifier are the most optimal algorithms for spam SMS detection. However, the SMS Spam Collection dataset is partially imbalanced, and thus merely contemplating accuracy can be misleading. Hence, the precision values should be also considered for the performance analysis.

TABLE I. Accuracy and Precision Values of the Implemented Algorithms

Algorithm	Accuracy	Precision
Logistic Regression	0.95	0.97
Support Vector Machine	0.97	0.97
Random Forest Classification	0.96	0.98
Extra Trees Classification	0.97	0.96
Gradient Boosting Classification	0.94	0.91
AdaBoost Classification	0.96	0.92
XGBoost Classification:	0.96	0.92

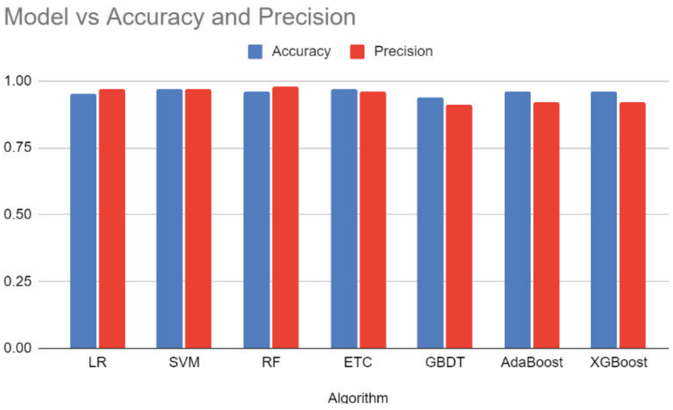


Fig. 3. Graphical representation of accuracy and precision values for comparative analysis.

Therefore, keeping the accuracy and precision scores into consideration, it can be concluded that the Extra Trees Classifier produces effective and unambiguous classification. Extra Trees, also called Extremely Randomized Trees, model on unpruned decision trees and choose features at every split in the decision tree, thus ensuring efficiency computationally and overcoming overfitting, unlike the traditional decision trees.

V. Future Scope

In the future, the main motive is to enhance the spam detection system further. One way is by broadening our dataset to include more types of messages, such as those in different languages and from various messaging apps. This will help our system recognize spam in diverse situations. Additionally, further investigation features like web links, files, and phone numbers within messages to improve our system's accuracy in identifying spam. Understanding sarcasm and context is also vital for better spam detection, so there is a need to teach the expert system to recognize these nuances. By implementing these improvements, it can be possible to make mobile phones safer from spam messages and other potential risks, like smishing attacks. The commitment is to refine the existing system to achieve even higher levels of accuracy, utilizing advanced techniques in machine learning. Ultimately, the goal is to create a spam detection system that is highly effective across different languages, messaging platforms, and communication styles, thereby enhancing overall cybersecurity for mobile device users.

VI. Conclusion

In this research, various machine-learning algorithms were investigated for SMS spam detection. The data preprocessing stage involved converting text to lowercase, tokenization, removing stop words and special characters, and stemming. This improved the data's suitability for machine learning models. Subsequently, seven machine learning algorithms were implemented and evaluated, ranging from Logistic Regression to ensemble methods like Gradient Boosting and XGBoost. While all models exhibited commendable performance, particularly in terms of accuracy, the Extra Trees Classifier emerged as the most effective solution for spam SMS detection. This algorithm's strength lies in its use of unpruned decision trees and random feature selection at each split, leading to better computational efficiency and overcoming overfitting issues compared to traditional decision trees.

Future work could involve testing the Extra Trees Classifier on more datasets to ensure its generalizability. Additionally, exploring hyperparameter tuning for the Extra Trees Classifier and potentially investigating deep learning architectures could lead to further performance improvements. In conclusion, our study contributes significantly to the field of SMS spam detection by providing a comprehensive framework that combines preprocessing techniques with a diverse range of machine-learning algorithms. This study paves the path for future research and development, especially in tackling issues like dataset imbalance and scalability, which will ultimately lead to the development of spam filtering systems that are more successful in practical applications.

References

[1] Bollam Pragna and M. Rama Bai, "Spam Detection using NLP Techniques," *International Journal of Recent Technology and Engineering (IJRTE)*, vol. 8, no. 2S11, September 2019, pp. 2424-2426.

[2] Yerakaraju, S., Krishna, P. G., & Raju, N. V. G. (2022). Spam Detection Using Machine Learning. Dogo Rangsang Research Journal UGC Care Group I Journal, Volume 09(Issue 01), ISSN: 2347-7180.

[3] Thanniru Lakshman, Singarapu Sanjay Kumar, Ulligaddala Satish Kumar, Yenikepalli Sri Sekhar, Yellamati Suresh. "SMS spam detection in Machine Learning using Natural Language Processing." International Journal of Advance Research, Ideas and Innovations in Technology, ISSN: 2454-132X, Volume 9, Issue 5 (V9I5-1190), Impact Factor: 6.078. Available online at: https://www.ijariit.com

[4] L. GuangJun, S. Nazir, H. U. Khan, and A. Ul Haq, "Spam Detection Approach for Secure Mobile Message Communication Using Machine Learning Algorithms," Security and Communication Networks, vol. 2020, Article ID 8873639, 6 pages, July 9, 2020. [Online]. Available: https://www.example.com.

[5] Poomka, Pumrapee & Pongsena, Wattana & Kerdprasop, Nittaya & Kerdprasop, Kittisak. (2019). SMS Spam Detection Based on Long Short-Term Memory and Gated Recurrent Unit. International Journal of Future Computer and Communication. 8. 11-15. 10.18178/ijfcc.2019.8.1.532.

[6] Xia, Tian, and Xuemin Chen. 2020. "A Discrete Hidden Markov Model for SMS Spam Detection" Applied Sciences 10, no. 14: 5011. https://doi.org/10.3390/app10145011

[7] S. Gadde, A. Lakshmanarao and S. Satyanarayana, "SMS Spam Detection using Machine Learning and Deep Learning Techniques," 2021 7th International Conference on Advanced Computing and Communication Systems (ICACCS), Coimbatore, India, 2021, pp. 358-362, doi: 10.1109/ICACCS51430.2021.9441783.

[8] Nagare, Samadhan. (2021). Mobile SMS Spam Detection using Machine Learning Techniques.. 7. 331-334.

[9] Abayomi-Alli, Olusola & Onashoga, Saidat & Sodiya, Adesina Simon & Ojo, Da & Ng,. (2015). A CRITICAL ANALYSIS OF EXISTING SMS SPAM FILTERING APPROACHES.

[10] Gupta, Suparna & Saha, Soumyabrata & Das, Suman Kumar. (2021). SMS Spam Detection Using Machine Learning. Journal of Physics: Conference Series. 1797. 012017. 10.1088/1742-6596/1797/1/012017.