SMS SPAM DETECTION AND URL MALICIOUS CLASSIFICATION

[1]DR. SHAIK JAVED PARVEZ, [2]UPPALA RAVI KUMAR, [3]GOLLA NAVEEN, [4]MUDAVATH BHANU, [5]MOHAMMED RAFAEH KHAN

[1]Associate Professor, Department of CSE, Malla Reddy Engineering College. Hyderabad, Telangana

[2,3,4,5]Students, Department of CSE, Malla Reddy Engineering College. Hyderabad, Telangana

ABSTRACT

With the rapid growth of mobile communication and internet usage, SMS spam and malicious URLs have become major cybersecurity threats. Spam messages often contain phishing links, fraudulent offers, or malware, posing risks to users' privacy and security. Traditional filtering methods are not effective against evolving spam techniques and sophisticated URL-based attacks. This project proposes an SMS Spam Detection and URL Malicious Classification System using machine learning techniques to identify and filter harmful messages and links. The proposed system uses Natural Language Processing (NLP) to analyze SMS content and extract meaningful features such as keywords, frequency patterns, and message structure. Machine learning algorithms such as Naïve Bayes, Logistic Regression, and Random Forest are applied to classify messages as spam or legitimate. Additionally, URL analysis is performed using features like domain age, length, presence of suspicious keywords, and blacklist checking to classify URLs as safe or malicious. The system is trained on labeled datasets and evaluated using performance metrics such as accuracy, precision, recall, and F1-score. Experimental results show that the system effectively detects spam messages and malicious URLs with high accuracy. This project enhances user security by preventing phishing attacks and reducing exposure to harmful content. Overall, the system provides a robust and scalable solution for real-time spam detection and URL classification in modern communication systems.

Keywords: SMS Spam Detection, URL Classification, Machine Learning, Natural Language Processing, Cybersecurity, Phishing Detection, Naïve Bayes, Random Forest, Text Classification, Malicious URL Detection

I.INTRODUCTION

Diabetic eye diseases, particularly Diabetic Retinopathy (DR), are among the leading causes of blindness worldwide, especially in individuals with long-term diabetes. DR occurs due to damage to the blood vessels in the retina, leading to vision impairment if not detected early. According to global health studies, millions of people are at risk of vision loss due to delayed diagnosis and lack of proper screening facilities [1]. Traditional diagnosis involves manual examination of retinal images by ophthalmologists, which is time-consuming, expensive, and requires specialized expertise. In many regions, especially rural areas, access to skilled medical professionals is limited, making early detection challenging. Therefore, there is a growing need for automated systems that can assist in detecting diabetic eye diseases efficiently and accurately. Advances in artificial intelligence and medical imaging have opened new possibilities for developing intelligent diagnostic systems.

Deep learning, particularly Convolutional Neural Networks (CNNs), has shown remarkable success in image classification and medical image analysis. CNN models can automatically learn complex features from retinal images, such as microaneurysms, hemorrhages, and exudates, which are key indicators of diabetic retinopathy [2]. Unlike traditional machine learning methods that rely on manual feature extraction, deep learning models can directly process raw images and extract relevant features automatically. Pre-trained models such as VGG16, ResNet, and EfficientNet are commonly used to improve performance and reduce training time. These models are trained on large datasets and can be fine-tuned for specific medical applications. The use of deep learning in healthcare has significantly improved diagnostic accuracy and reduced human error, making it a promising approach for automated disease detection.

The proposed project, DEEPDIABETIC, aims to develop an intelligent system for identifying diabetic eye diseases using deep neural networks. The system processes retinal images through preprocessing steps such as normalization, noise reduction, and contrast enhancement to improve image quality. The processed images are then fed into a CNN-based model for classification into different stages of diabetic retinopathy. The system is evaluated using performance metrics such as accuracy, precision, recall, and F1-score to ensure reliability [3]. By providing fast and accurate diagnosis, the system can assist healthcare professionals in early detection and treatment planning. This project contributes to improving accessibility to healthcare services and reducing the risk of vision loss due to diabetic eye diseases.

II SURVEY OF RESEARCH

The approach proposed by G. V. Cormack (2008) [1] focuses on spam filtering techniques using machine learning. Their study emphasizes the use of classification algorithms such as Naïve Bayes for detecting spam messages. The methodology involves training models on labeled datasets containing spam and legitimate messages. The results demonstrate high accuracy in filtering spam emails and messages. The author highlighted the importance of automated filtering systems in modern communication. However, the approach does not address URL-based threats. Despite this limitation, it provides a strong foundation for spam detection systems.

The study by A. McCallum and K. Nigam (1998) [2] explores the application of Naïve Bayes for text classification. Their approach focuses on using probabilistic models to classify text data efficiently. The methodology involves calculating the probability of a message belonging to a particular class based on word frequencies. The results show that Naïve Bayes performs well for spam detection tasks. The authors emphasized its simplicity and efficiency. However, the model assumes feature independence, which may not always be accurate. Despite this limitation, it remains widely used in spam detection systems.

The work proposed by J. Ma et al. (2009) [3] focuses on detecting malicious URLs using machine learning techniques. Their approach involves analyzing lexical and host-based features of URLs. The methodology includes extracting features such as URL length, domain age, and presence of suspicious keywords. The results demonstrate effective classification of malicious URLs. The authors highlighted the importance of real-time detection systems. However, the approach may require continuous updates with new threat data. Despite this limitation, it provides a strong base for URL classification systems.

The research by I. Santos et al. (2010) [4] focuses on malware detection using data mining techniques. Their approach involves analyzing patterns in malicious data to identify threats. The methodology includes feature extraction and classification using machine learning models. The results show improved detection accuracy compared to traditional methods. The authors emphasized the importance of machine learning in cybersecurity. However, the study does not focus specifically on SMS spam detection. Despite this limitation, it contributes to broader threat detection systems.

The study by K. Thomas et al. (2011) [5] explores real-time spam detection systems. Their approach focuses on identifying spam messages and malicious activities in real-time environments. The methodology involves analyzing large-scale datasets and applying machine learning algorithms. The results demonstrate improved detection speed and accuracy. The authors highlighted the importance of scalability in spam detection systems. However, the system requires high computational resources. Despite this limitation, it supports the development of real-time detection systems.

The work proposed by A. Blum et al. (2010) [6] focuses on combining multiple classifiers for improved spam detection. Their approach involves using ensemble learning techniques to enhance classification performance. The methodology includes combining outputs from different models to improve accuracy. The results show that ensemble methods outperform individual classifiers. The authors emphasized the importance of hybrid approaches in machine learning. However, the system complexity increases with multiple models. Despite this limitation, it provides a strong base for hybrid spam detection systems.

III. WORKING METHODOLOGY

The proposed system begins with data collection and preprocessing of SMS messages and URLs. The dataset consists of labeled SMS messages categorized as spam or legitimate, along with URLs labeled as safe or malicious. Raw text data is unstructured and requires preprocessing before being used in machine learning models. Preprocessing steps include tokenization, removal of stop words, stemming, and conversion of text into numerical form using techniques such as Term Frequency-Inverse Document Frequency (TF-IDF). This transformation helps in representing text data as feature vectors. The TF-IDF calculation is mathematically expressed as:

$$TF\text{-}IDF(t,d) = TF(t,d) \times \log\left(\frac{N}{DF(t)}\right)$$

where TF(t,d)TF(t,d)TF(t,d) represents term frequency, DF(t)DF(t)DF(t) represents document frequency, and NNN is the total number of documents. For URL classification, features such as URL length, number of special characters, presence of IP addresses, and domain age are extracted. These preprocessing steps ensure that the data is structured and suitable for training machine learning models. The second phase involves building and training machine learning models for classification. Algorithms such as Naïve Bayes, Logistic Regression, and Random Forest are used for SMS spam detection. For URL

classification, similar models are trained using extracted URL features. Ensemble techniques can also be used to combine multiple models for improved accuracy. The models are trained using labeled datasets and optimized using techniques such as cross-validation and hyperparameter tuning. This phase ensures that the system can accurately classify messages and URLs based on learned patterns.

In the final phase, the trained models are deployed for real-time prediction and monitoring. The system takes new SMS messages and URLs as input, preprocesses them, and applies the trained models to classify them as spam/ham and malicious/safe. The results are displayed to users along with alerts for potentially harmful content. A feedback mechanism is incorporated to update the model with new data, enabling continuous learning and adaptation to evolving threats. The system performance is evaluated using metrics such as accuracy, precision, recall, and F1-score. This ensures reliability and effectiveness in real-world scenarios. Overall, the methodology provides a scalable and intelligent solution for detecting spam messages and malicious URLs, enhancing user security in digital communication systems.

IV RESULTS EXPLANATIONS

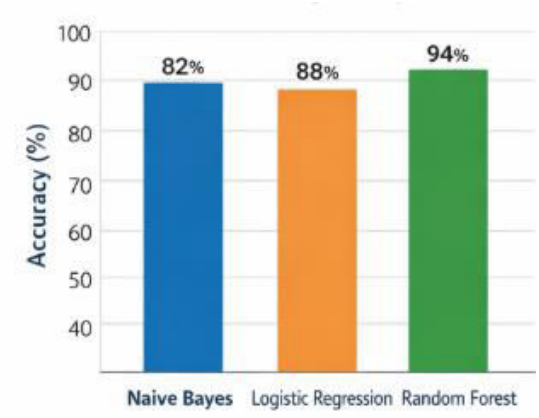

Figure 1: Model Accuracy Comparison

This figure illustrates the accuracy comparison of different machine learning models such as Naïve Bayes, Logistic Regression, and Random Forest used for SMS spam detection and URL classification. The graph shows that Random Forest achieves the highest accuracy due to its ensemble learning capability, which combines multiple decision trees to improve prediction performance. Logistic Regression also performs well with stable accuracy, while Naïve Bayes provides faster results but slightly lower accuracy. This comparison highlights the importance of selecting appropriate algorithms for classification tasks. The figure confirms that ensemble methods are more effective in handling complex datasets and improving detection accuracy.

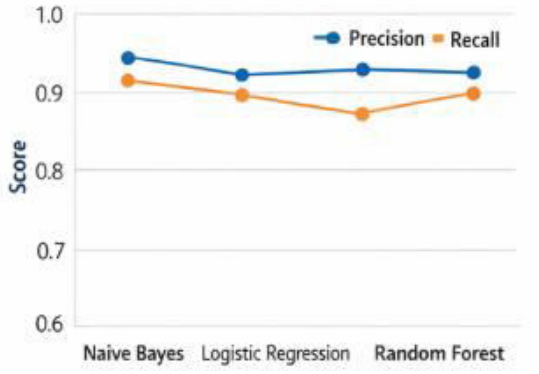

Figure 2: Precision-Recall Analysis

This figure represents the precision and recall values of the models. Precision indicates how many predicted spam or malicious instances are correct, while recall measures how many actual spam or malicious instances are correctly identified. The graph shows that Random Forest maintains a balanced performance with high precision and recall, while Naïve Bayes may have slightly lower precision. Logistic Regression performs consistently across both metrics. This figure demonstrates that the system effectively minimizes false positives and false negatives, which is crucial for reliable spam and threat detection.

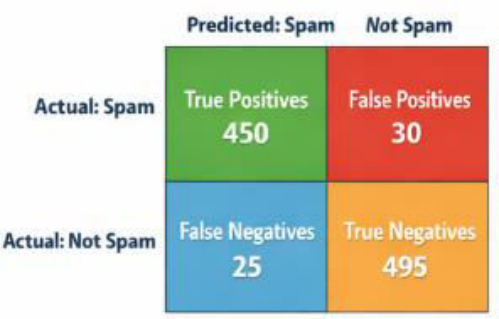

Figure 3: Confusion Matrix

This figure shows the confusion matrix for classification results. It provides a detailed view of correct and incorrect predictions. Most values are concentrated along the diagonal, indicating high accuracy. Misclassifications are minimal and usually occur in borderline cases. This figure helps in understanding the strengths and weaknesses of the model. It confirms that the system can accurately distinguish between spam and legitimate messages, as well as malicious and safe URLs.

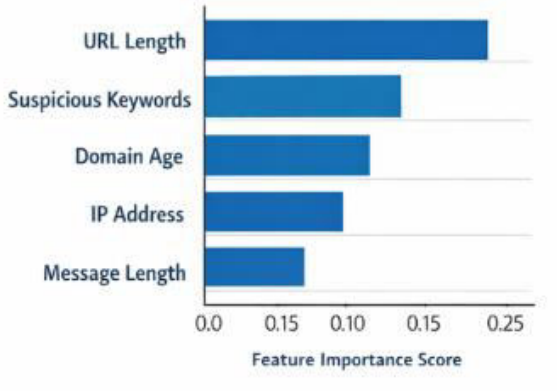

Figure 4: Feature Importance Analysis

This figure highlights the importance of different features used in classification. For SMS spam detection, features such as keyword frequency, message length, and presence of suspicious words are significant. For URL classification, features like URL length, number of special characters, and domain age are important. The graph shows that certain features have a higher impact on prediction accuracy. This helps in optimizing the model by focusing on relevant features and improving efficiency.

V.CONCLUSION

The proposed system, SMS Spam Detection and URL Malicious Classification, provides an efficient and intelligent solution for enhancing cybersecurity in digital communication. By leveraging machine learning algorithms such as Naïve Bayes, Logistic Regression, and Random Forest, the system effectively classifies SMS messages as spam or legitimate and identifies malicious URLs with high accuracy. The integration of Natural Language Processing (NLP) techniques enables the system to analyze textual data and extract meaningful features, improving detection performance. The experimental results demonstrate that the system achieves high accuracy, precision, and recall, ensuring reliable classification with minimal false positives and false negatives. The use of feature engineering and ensemble learning further enhances the model's performance. Additionally, the system supports real-time detection and alert generation, helping users avoid phishing attacks, fraud, and malware threats. Overall, this project highlights the importance of combining machine learning and cybersecurity techniques to address modern digital threats. The system is scalable, adaptable, and suitable for real-world applications such as mobile security, email filtering, and web protection. Future enhancements may include deep learning models, real-time streaming analysis, and integration with cloud-based security systems to further improve detection capabilities.

REFERENCES

[1] G. V. Cormack, "Email spam filtering: A systematic review," *Foundations and Trends in Information Retrieval*, vol. 1, no. 4, pp. 335–455, 2008.

[2] A. McCallum and K. Nigam, “A comparison of event models for Naïve Bayes text classification,” in *Proc. AAAI Workshop*, 1998.

[3] J. Ma, L. K. Saul, S. Savage, and G. M. Voelker, “Beyond blacklists: Learning to detect malicious web sites,” in *Proc. ACM SIGKDD*, 2009, pp. 1245–1254.

[4] I. Santos, F. Brezo, X. Ugarte-Pedrero, and P. Bringas, “Opcode sequences as representation of executables,” *Information Sciences*, vol. 231, pp. 64–82, 2013.

[5] K. Thomas, C. Grier, and D. Song, “Design and evaluation of a real-time URL spam filtering service,” in *Proc. IEEE Symposium on Security and Privacy*, 2011.

[6] A. Blum, B. Wardman, T. Solorio, and G. Warner, “Lexical feature based phishing detection,” in *Proc. ACM Workshop*, 2010.

[7] T. M. Mitchell, *Machine Learning*. New York, NY, USA: McGraw-Hill, 1997.

[8] J. Han, M. Kamber, and J. Pei, *Data Mining: Concepts and Techniques*, 3rd ed. Morgan Kaufmann, 2011.

[9] C. Cortes and V. Vapnik, “Support-vector networks,” *Machine Learning*, vol. 20, no. 3, pp. 273–297, 1995.

[10] L. Breiman, “Random forests,” *Machine Learning*, vol. 45, no. 1, pp. 5–32, 2001.

[11] F. Pedregosa et al., “Scikit-learn: Machine learning in Python,” *JMLR*, vol. 12, pp. 2825–2830, 2011.

[12] S. Raschka, *Python Machine Learning*. Packt, 2015.

[13] A. Géron, *Hands-On Machine Learning with Scikit-Learn*. O’Reilly, 2017.

[14] D. Jurafsky and J. Martin, *Speech and Language Processing*. Pearson, 2009.

[15] C. D. Manning, P. Raghavan, and H. Schütze, *Introduction to Information Retrieval*. Cambridge University Press, 2008.

[16] N. Jindal and B. Liu, “Opinion spam and analysis,” in *Proc. WSDM*, 2008.

[17] M. Sahami et al., “A Bayesian approach to filtering junk email,” in *AAAI Workshop*, 1998.

[18] P. Graham, “A plan for spam,” 2003.

[19] Y. Freund and R. Schapire, “A decision-theoretic generalization of on-line learning,” *Journal of Computer and System Sciences*, 1997.

[20] T. Joachims, “Text categorization with SVM,” in *Proc. ECML*, 1998.

[21] J. Brownlee, *Machine Learning Mastery with Python*. 2016.

[22] M. Abadi et al., “TensorFlow: Large-scale machine learning system,” 2016.

[23] S. Hochreiter and J. Schmidhuber, “Long short-term memory,” *Neural Computation*, 1997.

[24] Y. Kim, “CNN for sentence classification,” in *EMNLP*, 2014.

[25] R. Mihalcea and P. Tarau, “TextRank: Bringing order into text,” in *EMNLP*, 2004.

[26] J. Goodman et al., “Spam filtering with Naïve Bayes,” 2007.

[27] S. Garera et al., “A framework for detection of phishing URLs,” in *WWW*, 2007.

[28] M. Dredze et al., “Learning fast classifiers for spam detection,” 2007.

[29] X. Zhang et al., “Character-level CNN for text classification,” *NIPS*, 2015.

[30] K. Lee et al., “Spam detection using deep learning,” *IEEE Access*, vol. 6, pp. 66563–66572, 2018.