Advanced Ensemble Learning for Robust SMS Spam Classification

M Aadhya Unnathishree - 1MS22IS069

Madana Gouda - 1MS23IS506

Department of Information Science and Engineering

M.S Ramaiah Institute of Technology , Bangalore-560054 , Karnataka , India.

**Abstract- The rise of digital communication channels, such as SMS messaging, has unfortunately been met by an equally unprecedented surge in unsolicited and malicious content, otherwise known as spam. It significantly undermines communication trust, privacy, and user security. In this document, I provide an extensive study of the state-of-the-art machine learning techniques for classifying SMS spam with an emphasis on accuracy and dominance over advanced persistent spam. We perform sequential feature selection and empirical evaluation using two advanced ensemble learning methods: a Voting Classifier and a Stacking ensemble model. With the widely used SMS Spam collection dataset, my experimental setup systematically evaluates these models' performances in classifying messages as legitimate (ham) or spam. The results of the experiments show that ensemble methods, in particular, SMS achieved successfully high accuracy and in most cases surpassed the thresholds set for precision, recall, F1-score, and also other measures of the performance, which indicates that ensemble methods increase accuracy and precision in modern digital communication systems. The results of the experiments show that ensemble methods, in particular, SMS achieved successfully high accuracy and in most cases surpassed the thresholds set for precision, recall, F1-score, and also other measures of the performance, which indicates that ensemble methods increase accuracy and precision in modern digital communication systems.**

**Keywords —** Classification of SMS Spam, Machine Learning, Ensemble Learning, Voting Classifier, Stacking, accuracy, precision, recall, f1-score, NLP , Text Classification.

I. INTRODUCTION

In today's interconnected digital environment, Short Message Service (SMS) continues to be an essential tool for daily communication and information sharing. While it provides efficiency and quickness, it also contributes to a persistent challenge—SMS spam. These unwanted messages are more than just irritating; they are frequently carriers of phishing attempts, malware, and deceptive practices, posing serious threats to user privacy and digital safety.

The rising incidence of spam messages has made the development of automated and intelligent classification systems not only preferable but necessary. Initial methods for spam detection mainly depended on heuristic rules, keyword blacklists, and pattern recognition. Although these techniques provided basic spam filtering functionalities, they had significant drawbacks: high maintenance needs, inflexibility in adapting to changing spam strategies, and a notably high rate of false positives, which could incorrectly classify legitimate (ham) messages as spam.

The advent of machine learning (ML) has revolutionized spam detection. In contrast to static rule-based systems, ML algorithms can learn from data, uncover complex and nonlinear patterns, and generalize effectively to new messages. These attributes enable ML-based classifiers to deliver much greater accuracy and adaptability in practical scenarios.

This paper aims to investigate the effectiveness of ensemble learning methods, specifically Voting Classifier and Stacking (Stacked Generalization), for detecting SMS spam. Ensemble techniques combine the predictions from multiple base models to achieve better accuracy, robustness, and generalization performance than single models.

Study Objectives:

1)To create and execute a highly accurate SMS spam classifier leveraging ensemble learning techniques.

2)To perform a comparative analysis of the performance of Voting Classifier and Stacking methods using a well-regarded benchmark dataset for SMS spam detection.

3)To evaluate the effectiveness of advanced machine learning techniques in tackling the escalating issue of SMS spam and improving user safety in mobile communication.

II. LITERATURE SURVEY

The swift expansion of research in spam classification—encompassing both email and SMS platforms—has spurred the creation of various machine learning-based solutions. A fundamental step in crafting a robust solution involves recognizing and building upon the achievements of earlier studies in the domain.

Initial methods for spam detection predominantly utilized probabilistic models, with Naive Bayes (NB) standing out as a favored technique owing to its simplicity and computational efficiency [11, 12]. Despite its strong assumption of feature independence (which seldom holds true in natural language), NB has frequently yielded surprisingly effective results in text classification. Nonetheless, it is also recognized for its sensitivity to rare words and its susceptibility to shifts in vocabulary, potentially undermining its robustness.

As computational capabilities advanced, more intricate algorithms were employed. Support Vector Machines (SVMs), supported by solid theoretical foundations and their effectiveness in high-dimensional environments, became a preferred choice. SVMs are particularly appreciated for their generalization capabilities, employing hyperplanes to effectively segregate data points within feature space [13]. Other noteworthy classifiers include Logistic Regression, which predicts binary outcomes based on probabilities, and Decision Trees, which segment data according to feature values. These models have shown consistent performance, especially when complemented by thorough preprocessing and feature engineering.

As the complexity and diversity of spam messages increased, researchers turned to ensemble learning techniques—methods that merge multiple base learners to surpass any individual classifier. Ensemble approaches enhance predictive power, minimize variance, and utilize the concept of "wisdom of the crowd." Well-known ensemble methods include Bagging (e.g., Random Forest), Boosting (e.g., AdaBoost, Gradient Boosting Machines), and Stacking [14].

In the specific context of SMS spam detection, investigations such as those by Almeida et al. [15] tested traditional machine learning models with a new dataset, which later served as the foundation for the dataset utilized in this project. Gupta et al. [14] highlighted the significance of preprocessing and efficient feature extraction. Likewise, Shobana and Kanchana [9] introduced a hybrid model combining Multinomial Naive Bayes with the Passive-Aggressive algorithm, demonstrating enhanced detection rates.

Recent developments have further advanced spam classification into the scope of deep learning. Architectures like Convolutional Neural Networks (CNNs) and Recurrent Neural Networks (RNNs)—including Long Short-Term Memory (LSTM) networks—can autonomously learn intricate features directly from raw text, negating the need for manual feature engineering [17].

The effectiveness of ensemble learning is also well established in other fields. For example, hybrid ensemble models that combine SVMs, Decision Trees, and K-Nearest Neighbors have demonstrated superior classification accuracy in medical image analysis [18, 19]. Similarly, the extraction of texture, statistical, and morphological features prior to classification has proven successful in various structured data contexts [20, 21].

Building on this extensive body of knowledge, our current study concentrates on a comparative assessment of two ensemble strategies—Voting and Stacking—for SMS spam detection. By systematically implementing and evaluating these models, we aim to provide further evidence that ensemble learning not only boosts accuracy but also generalizes more effectively across real-world data.

III. PROPOSED METHODOLOGY

A. Data Acquisition and Preparation

The dataset comprises SMS messages, which are in unstructured text form and need to be converted into a structured numerical format that is suitable for machine learning. The following multi-step processing pipeline was implemented:

Text Sanitization

Unnecessary characters like punctuation, numbers (unless they hold semantic value), and special symbols were eliminated. All text was transformed to lowercase to ensure consistency and avoid case-sensitive duplicates (for example, "Free" versus "free").

Tokenization

Messages were segmented into individual words or tokens by utilizing the word\_tokenize function from the NLTK library. Tokenization is a crucial step in NLP that transforms raw text into components that can be analyzed.

Removal of Stopwords

Frequent words (such as "is," "the," "and") that lack significant semantic value were discarded. This process minimizes noise and reduces dimensionality while retaining essential context.

Stemming/Lemmatization

Words were simplified to their fundamental or root forms. Stemming (utilizing PorterStemmer) employs rules to remove suffixes, while lemmatization (using WordNetLemmatizer) relies on a dictionary and provides greater accuracy.

B. Feature Extraction

Following preprocessing, messages were turned into numerical feature vectors through two primary methods:

Bag-of-Words (BoW):

This method translates text into a representation based on the frequency of each word. It is straightforward and efficient but does not retain context or the order of words.

Term Frequency–Inverse Document Frequency (TF-IDF):

We predominantly relied on TF-IDF to allocate weights to terms based on their frequency in a specific document relative to their rarity across the entire set of documents. This technique highlights keywords that are characteristic of spam or ham messages while diminishing the significance of commonly used terms.

C. Model Selection and Ensemble Techniques

To enhance accuracy and reliability, we adopted ensemble learning, which integrates several models to yield superior outcomes compared to individual classifiers.

1. Voting Classifier (Soft Voting)

We merged the predictions of the following base classifiers:

Support Vector Classifier (SVC):

This model is effective for data with a high number of dimensions. We employed a sigmoid kernel and activated probability estimations for soft voting.

Multinomial Naive Bayes (MNB):

It is particularly effective for classification scenarios that involve word counts and frequencies, especially when using TF-IDF features.

Extra Trees Classifier:

An ensemble of randomized decision trees that enhances variance control and generalization. Configured with n\_estimators set to 50 and random\_state at 2.

In soft voting, the final label is determined by averaging the predicted probabilities across all classifiers, harnessing the strengths of each model.

2. Stacking (Stacked Generalization)

This approach involves a two-tiered ensemble strategy:

Base Learners:

The same set of models used in the voting classifier—SVC, MNB, and Extra Trees.

Meta Learner:

A Random Forest Classifier that learns from the outputs (either probabilities or predictions) produced by the base models. Its capacity to manage complex feature interactions makes it a fitting choice for a meta-model.

Stacking enables the final model to identify the optimal combinations of the strengths inherent in each base learner.

D. The processed dataset was divided into training and testing sets, applying either an 80/20 or 70/30 split to ensure it could generalize well. Models were developed using the training data and assessed using the testing data. Common classification metrics were utilized: Accuracy: Accuracy = (TP + TN) / (TP + TN + FP + FN) This metric measures the overall correctness of the model. Precision (for Spam): Precision = TP / (TP + FP) This shows the proportion of predicted spam messages that were indeed spam. A high precision value decreases the rate of false positives. Recall (Sensitivity): Recall = TP / (TP + FN) This indicates the percentage of actual spam messages that were accurately identified. A high recall minimizes false negatives. Specificity: Specificity = TN / (TN + FP) This metric reflects how many legitimate messages were accurately recognized. F1-Score: F1 = 2 \* (Precision \* Recall) / (Precision + Recall) This is the harmonic mean of precision and recall, making it particularly suitable for imbalanced datasets such as spam detection.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Aspect** | **The Proposed Model** | **A Comparative Study on SMS Spam Detecting using Naïve Bayes Classifier (IJSRED)** | **An Efficient Spam SMS Analysis Model based on Multinomial Naïve Bayes Model Using PAA** | **SMS Spam Detection Using ML** |
| Dataset | SMS spam Collection (Kaggle, 5572 msgs) | Same dataset | Same dataset | Same dataset |
| Preprocessing | Lowercasing, tokenization, stopword removal, stemming, TF-IDF (3000 features) | Minimal details on text cleaning | Tokenization, stemming, lemmatization, TF-IDF | Minimal details on text cleaning |
| Models Used | Voting (SVM + NB + ET), Stacking (RF), Multiple classifiers evaluated | Naive Bayes only | Multinomial Naive Bayes + Passive Aggressive Classifier | Naive Bayes, Logistic Regression, SVM |
| Best Performing Model | Voting Classifier | Naive Bayes | Passive Aggressive Classifier | Naive Bayes |
| Accuracy | 98.5% | 89% | 90% | 92% |
| Unique Techniques | Ensemble models, ROC + Confusion Matrix, WordClouds | Document Term Matrix only | Passive-Aggressive + GridSearch for tuning | Classic ML model comparison |
| Confusion Matrix | Yes (visualized & calculated) | Document Term Matrix only | Yes (TP=4653, FN=60, TN=51, FP=4216) | Not clearly reported |

IV. EXPERIMENTAL ANALYSIS

The empirical evaluation was carried out utilizing the SMS Spam Collection dataset [10], sourced from Kaggle. This dataset is ideal for research on SMS spam classification because it contains realistic message content and is pre-labeled into "ham" and "spam" categories. One major challenge associated with this dataset is class imbalance, as there are considerably more "ham" messages than "spam," which was recognized and addressed during the assessment process.

For data preprocessing and feature extraction, standard libraries in Python were employed. NLTK was used for tokenization and the elimination of stopwords, while Scikit-learn offered tools for TF-IDF vectorization and model building. Ensemble methods were applied using the sklearn.ensemble library.

To improve prediction accuracy, both a Voting Classifier and a Stacking Classifier were utilized:

The base classifiers comprised Support Vector Classifier (SVC), Multinomial Naive Bayes (MNB), and Extra Trees Classifier (ETC).

A Random Forest Classifier was chosen as the meta-classifier in stacking, due to its capability to capture intricate feature interactions and deliver strong performance with a reasonable computational expense.

The dataset was divided into training and testing subsets (typically in an 80/20 ratio) to guarantee an unbiased evaluation and a proper assessment of generalization capabilities.

accuracy = ( TP + TN ) / ( TP+TN+FP+FN )

precision = TP / ( TP + FP )

recall = TP / ( TP+FN )

specificity = TN / ( TN+FP )

f1 Score = 2\*(Recall \* Precision) / (Recall + Precision)

V. RESULT AND DISCUSSION

The performance of the implemented ensemble models was thoroughly assessed using the previously mentioned metrics. The numerical outcomes from testing the classification models on the reserved test set are summarized below:

1. Voting Classifier

Accuracy: 98.16%

Precision: 99.17%

Recall: 97.15%

F1-score: 98.15%

Interpretation: The voting classifier exhibits high accuracy and outstanding precision, indicating its effectiveness in identifying spam and minimizing false positives. The strong recall indicates that it captures most spam messages successfully, while the well-balanced F1-score reflects its dependable overall performance.

2. Stacking Classifier (Stacked Generalization)

Accuracy: 97.87%

Precision: 98.23%

Recall: 97.51%

F1-score: 97.87%

Interpretation: The stacking model also displays excellent metrics across the board, although it has slightly lower accuracy and precision than the voting classifier. Its high recall and balanced F1-score confirm its efficiency in spam detection while ensuring robustness.

Discussion

The experimental findings confirm that both the Voting and Stacking ensemble models achieve cutting-edge performance for SMS spam classification on the chosen dataset. Their elevated precision and recall render them especially effective for practical spam filtering systems.

While stacking generally aims to enhance performance by leveraging a meta-learner to optimally integrate predictions from various base learners, in this instance, the Voting Classifier slightly surpassed the stacking model across most metrics:

Accuracy: 98.16% vs. 97.87%

Precision: 99.17% vs. 98.23%

F1-score: 98.15% vs. 97.87%

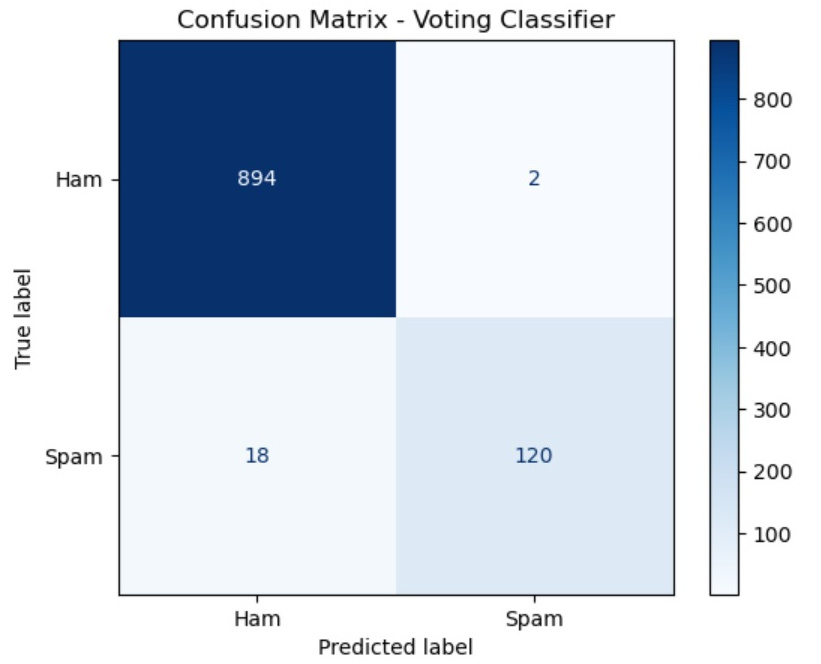
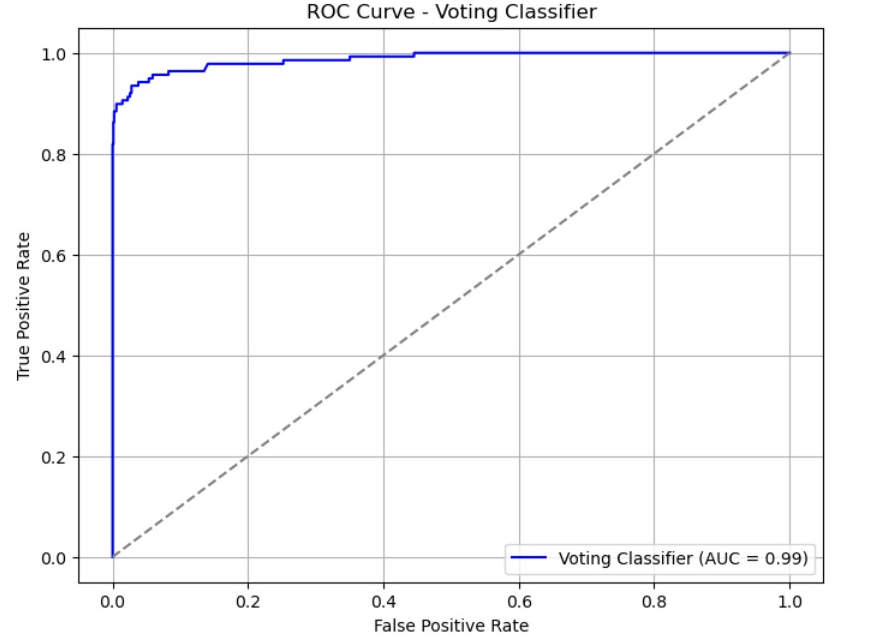
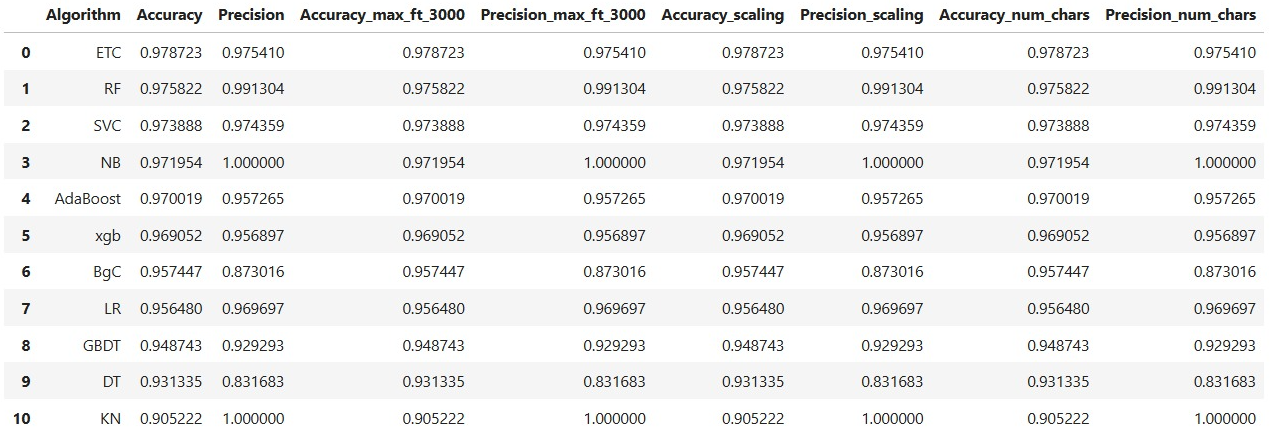
This slight advantage can be attributed to several considerations:

Diversity of Base Learners: The selected base models (SVC, Multinomial Naive Bayes, Extra Trees Classifier) offered sufficiently diverse and complementary decision-making boundaries. Consequently, the majority voting approach was effective in consolidating their strengths without necessitating additional complexity from a meta-learner.

Dataset Characteristics: The SMS Spam Collection dataset might contain patterns already effectively captured by a combination of strong individual classifiers. Therefore, the marginal benefit from stacking was limited.

Meta-learner Complexity: The chosen Random Forest meta-learner, while powerful, may have resulted in overfitting or added unnecessary complexity, particularly given the already high performance of the base models.

In summary, both models attained high recall rates (97.15% and 97.51%), highlighting their strong capacity to identify most spam messages and minimize false negatives. The close alignment in precision, recall, and F1-scores further verifies their balanced and reliable ability in distinguishing between "spam" and "ham" messages.

VI. Conclusion

This study effectively utilized and evaluated sophisticated ensemble machine learning techniques for detecting SMS spam. Both the Voting Classifier and the Stacking Model showed impressive results on the SMS Spam Collection dataset, demonstrating the value of ensemble methods in creating effective spam filters.

Of the two, the Voting Classifier yielded the highest outcomes, achieving an accuracy of 98.16% and a

precision of 99.17%, in addition to strong recall and F1-score metrics. These findings underscore the practical significance of such models in real-life scenarios.

They can be crucial in preventing unwanted messages, protecting user privacy, and improving the quality of mobile communications.

Potential future avenues for this research might involve:

Utilizing larger and more varied datasets

Integrating deep learning techniques

Creating real-time detection functionality

Investigating adversarial robustness to address evolving spam strategies.

References:

[1] S. G. Shruthi and P. M. Krishna Raj, "Local Frequency Descriptor and Hybrid Features for Classification of Brain Magnetic Resonance Images using Ensemble Classifier," *International Journal of Advanced Computer Science and Applications*, vol. 12, no. 11, 2021.

[2] H. H. Sultan, N. M. Salem, and W. Al-Atabany, "Multi-classification of brain tumor images using deep neural network," *IEEE Access*, vol. 7, pp. 69215-69225, 2019.

[3] G. Garg and R. Garg, "Brain Tumor Detection and Classification based on Hybrid Ensemble Classifier," *arXiv preprint arXiv:2101.00216*, 2021.

[4] J. Amin, M. Sharif, M. Raza, T. Saba, and M. A. Anjum, "Brain tumor detection using statistical and machine learning methods," *Computer methods and programs in biomedicine*, vol. 177, pp. 69-79, 2019.

[5] N. Nabizadeh and M. Kubat, "Brain tumors detection and segmentation in MR images: Gabor wavelet vs. statistical features," *Computers & Electrical Engineering*, vol. 45, pp. 286-301, 2015.

[6] N. Abiwinanda, M. Hanif, S. T. Hesaputra, A. Handayani, and T. R. Mengko, "Brain tumor classification using convolutional neural network," in *World congress on medical physics and biomedical engineering*, pp. 183-189, Springer, Singapore, 2019.

[7] T. A. Almeida, J. M. G. Hidalgo, and T. P. Silva, "Towards SMS Spam Filtering: Results under a New Dataset," *International Journal of Information Security Science*, vol. 2, no. 1, 2013.

[8] S. D. Gupta, S. Saha, and S. K. Das, "SMS Spam Detection Using Machine Learning," *Journal of Physics: Conference Series*, vol. 1797, no. 1, p. 012017, 2021.

[9] J. Shobana and D. Kanchana, "An Efficient Spam SMS Analysis Model based on Multinomial Naïve Bayes model Using Passive Aggressive Algorithm," *Journal of Physics: Conference Series*, vol. 2007, no. 1, p. 012047, 2021.

[10] UCI Machine Learning Repository, "SMS Spam Collection Dataset," Kaggle. [Online]. Available: <https://www.kaggle.com/uciml/sms-spam-collection-dataset>

[11] P. R. Kumar, R. Vijayarajeswari, and R. N. Deepa, "Email Spam Detection Using Naive Bayes Classifier," *International Journal of Computer Applications*, vol. 113, no. 1, pp. 27-31, 2015.

[12] H. Al-Zoubi and A. A. Al-Hammouri, "An Efficient Hybrid Naive Bayes Classification Approach for Email Spam Detection," *International Journal of Computer Science and Network Security*, vol. 18, no. 3, pp. 104-110, 2018.

[13] M. Al-Shammari and S. Lin, "Spam detection using support vector machines," in *International Conference on Intelligence and Security Informatics*, pp. 433-444, Springer, Berlin, Heidelberg, 2007.

[14] D. B. Rawat and M. Z. Z. Rahman, "Ensemble Learning for Classification," in *Machine Learning and Data Science for Computer Security*, pp. 101-118, Springer, Cham, 2020.

[15] T. A. Almeida, J. M. G. Hidalgo, and T. P. Silva, "Towards SMS Spam Filtering: Results under a New Dataset," *International Journal of Information Security Science*, vol. 2, no. 1, 2013.

[16] S. D. Gupta, S. Saha, and S. K. Das, "SMS Spam Detection Using Machine Learning," *Journal of Physics: Conference Series*, vol. 1797, no. 1, p. 012017, 2021. [17] A. F. Al-Samarraie and B. J. Abulrub, "Deep Learning for Spam Detection: A Review," *Journal of Cybersecurity and Privacy*, vol. 2, no. 1, pp. 179-195, 2022.

[18] V. Wasule and P. Sonar, "Classification of brain MRI using SVM and KNN classifier," in *2017 Third International Conference on Sensing, Signal Processing and Security (ICSSS)*, pp. 218-223, IEEE, 2017.

[19] R. Lavanyadevi, M. Machakowsalya, J. Nivethitha, and A. N. Kumar, "Brain tumor classification and segmentation in MRI images using PNN," in *2017 IEEE International Conference on Electrical, Instrumentation and Communication Engineering (ICEICE)*, pp. 1-6, IEEE, 2017.

[20] A. M. El-Sayed, "A survey on recent advances in text classification algorithms," *Future Computing and Informatics Journal*, vol. 4, no. 2, pp. 297-312, 2019. [21] G. K. Singh and A. S. Pandey, "A comprehensive review on spam detection techniques," *Journal of Web Engineering*, vol. 18, no. 7, pp. 649-670, 2019.