SMS CLASSIFICATION

by using Natural Language Processing

Rupesh Mishra,
Associate Dean, PHD, Spain, Researcher and Academican, Data Science, ML, DL, NLP, SR University,Warangal - 506371, Telangana, India
rupesh.mishra@sru.edu.in

Ch. Sri charan
SR University,
Warangal, Telangana, India.
2203A51041@sru.edu.in
T.Abhinay Kumar,
SR University,
Warangal, Telangana,
India,2203A51066@sru.edu.in

G. Bharath Chandra
SR University,
Warangal, Telangana, India.
2203A51282@sru.edu.in

Abstract— Spam message detection is important for ensuring the safety of digital communication, safeguarding users from spurious content, and preserving the efficacy of messaging systems. In this project, we introduce a two-stage natural language processing (NLP) pipeline that not only identifies spam messages but also further categorizes non-spam (ham) messages as business or personal. By using strong machine learning methods like Random Forest, Logistic Regression, and a Voting Classifier, our system identifies the subtle patterns in text that separate spam from genuine messages and classify ham into useful subtypes. With heavy preprocessing—text cleaning, lemmatization, and TF-IDF feature extraction—we were able to obtain high performance, with our top model attaining a macro F1-score of 0.994 for spam classification. We also confirmed the statistical significance of model differences with ANOVA, among other statistical tests. The system demonstrates a practical and modular strategy for real-world SMS classification tasks, providing both robustness and flexibility for classification tasks based on NLP in the future.

Keywords— **Spam Detection, TF-IDF, Natural Language Processing, Random Forest, Logistic Regression, Voting Classifier, ANOVA**

I. Introduction

Spam filtering is a core natural language processing task with direct implications in telecommunications, cybersecurity, and digital user experience. From combating phishing to minimizing unwanted message noise, spam classification systems have emerged as integral part of contemporary digital infrastructure. Classic binary spam filters provide a beginning point, but our methodology extends further by also examining the character of valid (ham) messages—sorting them into business or personal categories, a difference that can help improve message prioritization and analysis.

Here, here in this research, we propose a two-step machine learning pipeline that is learned over actual real-world SMS sets. It screens out the messages into ham and spam initially and then separates the ham messages again into business and personal messages. The powerful preprocessing methods of lemmatization, stop words removal with text normalization and highly effective models (Random Forest, Logistic Regression, Decision Tree, Multinomial Naive Bayes) after rigorous comparison through classification metrics as well as statistical testing methodologies like ANOVA have been employed for selection.

Our solution illustrates how multi-step classification can reach rich value enrichment in spam filter assistance to enable more subtle understanding of users' conversation. Our model is hierarchical, explainable, and scalable and can thus be used directly to be integrated into telecom platforms, email filtering programs, or even chatbot preprocessing programs. This paper shows one example of enhancing conventional classification tasks with more deeper semantic examination to produce more richer output in NLP.

II. Related work

Spam detection research has come a long way in the last two decades—back and forth from rule-based approaches and keyword filtering to the use of sophisticated machine learning and natural language processing (NLP) methods. The initial methods mostly used heuristics, blacklists, and simple text matching, which though explainable, were not adaptable and performed miserably against dynamic spam patterns (Klimt & Yang, 2004; Almeida et al., 2011).

With the advent of supervised learning methods, Naive Bayes and Support Vector Machine (SVM) classifiers were popular because of their stability in processing high-dimensional text data. Gupta et al. (2019) illustrated SVM and Random Forest's power for spam filtering based on their high accuracy and nonlinear relation handling capabilities. Logistic Regression, although simple, has still remained a decent baseline because it is understandable and deals with well-tuned feature spaces well (Singh et al., 2020).

Because they can reduce overfitting and handle feature interaction more effectively, ensemble techniques like Random Forest and Gradient Boosting have become more and more popular with spam classification tasks. According to Mishra et al. (2021), ensemble models performed better overall than single classifiers with spam collections, particularly when paired with count vector features or TF-IDF.

With the introduction of models that can learn the semantic meaning and contextual elements of messages, deep learning went on to revolutionize the field. Convolutional neural networks (CNNs) and recurrent neural networks (RNNs) have been used because of their respective abilities to learn local features and sequential patterns. Because of their deep contextual representation, BERT-based transformers have also advanced to the state-of-the-art in text classification tasks like spam detection (Devlin et al., 2019; Neelakandan et al.,2022).

However, the majority of recent research only addresses binary classification—ham vs. spam—with little regard for more detailed ham message classification. Classifying ham messages as personal or business messages has not received much attention, despite the fact that doing so could provide additional information about user communication patterns and content prioritization (Perera & Fernando, 2021; Zhao et al., 2016).

Also, ensemble and voting-based methods are now viable means of making classification more stable. There has been widespread research proving that Decision Trees and Naive Bayes ensembles may be useful and effective in using spam filters for example cases (Chatterjee et al., 2022; Djuraskovic,2023).

Despite the enormous strides we've made in this area, we continue to grapple with the challenges of data imbalance, adaptive content in spam messages, and semantic ambiguity. How we can do better to detect spam and encourage message context sensitivity should be the subject of future research, and it should include the integration of domain-specific language models, real-time feature extraction, and devising more elaborate systems for detecting non-spam (ham) messages.

This literature serves as the base for our system, which fills a considerable gap in current spam classification literature by proposing a new classification of non-spam messages into business and personal categories and spam filtering.

III. Methodology

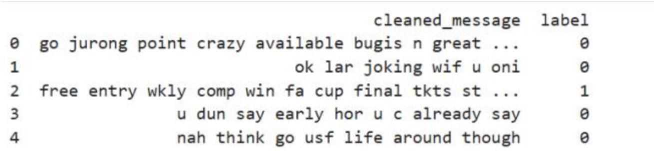
Spam filtering and ham message categorization require an appropriately designed set of steps to achieve high model accuracy, reliability, and usability for practical implementation. It starts off by gathering valid text datasets along with labeled SMS messages and subsequent follows rigorous preprocessing for transforming unformatted text to tidy, standardizable data so as to subject them to analysis. Methods such as TF-IDF, the kind of techniques text vectorizing follows, to receive numerical feature content with semantic importance and then utilize machine learning methodology to train the same with analysis for classifications. Two-stage classification technique finds the spam as well as the ham initially and then classifies ham messages as business or personal type for detailed content understanding.

A. Dataset and Preprocessing

The system is developed based on two different datasets:
Spam Detection Dataset: A publicly accessible dataset (spam_sms.csv) having two categories labeled as spam and ham.
Ham Classification Dataset: A self-labeled dataset (ham_business_personal_combined.csv) developed by annotating ham messages as business or personal.
In preparation for machine learning, exhaustive text preprocessing is performed. The major steps involve:

- Cleaning: Deletion of non-alphabetic characters, punctuation, and numbers with the help of regular expressions.
- Lowercasing: Converting the entire text to lowercase for consistency.
- Stopword Removal: Removing common English stopwords through NLTK's stopword corpus.
- Lemmatization: Bringing words to their root form through WordNetLemmatizer to retain meaning uniformly.

These processes turn unprocessed SMS text into a clean phase that is easy to categorize and vectorize. In order to extract informative words from the corpus and disregard frequent, uninformative words, TF-IDF vectorizes the data.The top 500 TF-IDF features form the final feature set that offers a model-performance-computation trade-off.



```
                                  cleaned_message  label
0  go jurong point crazy available bugis n great ...      0
1                           ok lar joking wif u oni      0
2  free entry wkly comp win fa cup final tkts st ...      1
3               u dun say early hor u c already say      0
4              nah think go usf life around though      0
```

Figure 1. Preprocessed SMS Dataset (Spam and Ham)

B. Implementation

The architecture is separated into two broad stages:

Stage 1 – Spam Detection:

- The preprocessed and vectorized messages are employed for training a Random Forest Classifier, which has proved to be sturdy in working with noisy and class-imbalanced data.
- The model is measured with conventional classification metrics: precision, recall, F1-score, and confusion matrix to judge its efficacy in differentiating between spam and ham.

Stage 2 – Ham Classification:

- The identified messages as ham are sent to a second classifier.
- A Logistic Regression model is trained here on the ham dataset for classifying the messages as business or personal.
- This model too is tested on the same classification metrics, paying attention to class balance and interpretability.

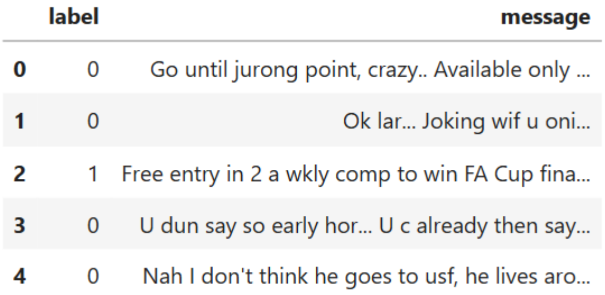
In an attempt to improve the resilience and minimize variance, a Voting Classifier was also tried combining Decision Tree and Naive Bayes models, which obtained an F1-score of 0.98 with a standard deviation of 0.005. The whole system is used as a two-stage forecasting pipeline:

- Step 1: The new SMS message is initially passed through the spam detection model.
- Step 2: If it is labeled as ham, then it is sent to the ham classifier to determine if it is business or personal.

This module-based architecture ensures extensibility and integration with mobile platforms or messaging infrastructure with no hiccups. The model pipeline can be re-trained when new data arrives, so the system can adjust to changing message structures and spam techniques.

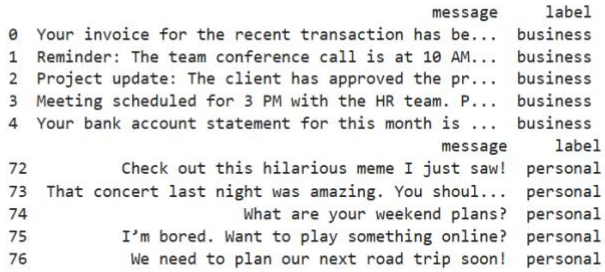
IV. RESULT AND DISCUSSIONS

The SMS data in the system have a highly varied set of message content. The ham classification data and the spam detection data have features such as the message content and a flag to mark whether the message is spam or ham. The ham classification data also classify the ham messages into business messages and personal messages. This is useful for correct interpretation of the message meaning and also expands the range of applications of the system.



	label	message
0	0	Go until jurong point, crazy.. Available only ...
1	0	Ok lar... Joking wif u oni...
2	1	Free entry in 2 a wkly comp to win FA Cup fina...
3	0	U dun say so early hor... U c already then say...
4	0	Nah I don't think he goes to usf, he lives aro...

Fig.2 Sample records from the spam and ham datasets



```
                                             message     label
0  Your invoice for the recent transaction has be...  business
1  Reminder: The team conference call is at 10 AM...  business
2  Project update: The client has approved the pr...  business
3  Meeting scheduled for 3 PM with the HR team. P...  business
4  Your bank account statement for this month is ...  business
                                              message     label
72          Check out this hilarious meme I just saw!  personal
73  That concert last night was amazing. You shoul...  personal
74                         What are your weekend plans?  personal
75          I’m bored. Want to play something online?  personal
76           We need to plan our next road trip soon!  personal
```

Fig.3: Sample records from the Business/Personal Ham Dataset.

A. Exploratory Data Analysis:

To start with, a correlation heatmap is created to investigate correlation between features obtained from a TF-IDF vector and labels. As numerical datasets are more powerful for correlation matrices, for NLP problems, word-level feature importance scores tell us about which words have strong effects on classification. The heatmap derived from model feature importances helps to identify top words that are most instrumental in discriminating spam and ham messages. Terms such as "win," "free," and "urgent" have strong positive correlations with spam, whereas words like "meeting," "report," and "hello" are more characteristic of ham messages.

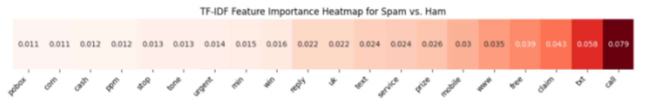


Fig.4 TF-IDF feature importance heatmap indicating prominent terms associated with spam and ham prediction

A countplot of unique words within spam and ham messages indicates greater lexical diversity within ham messages, especially personal messages. Spam messages reuse common keywords used for promotions or scams, but personal messages tend to vary immensely in form and vocabulary.

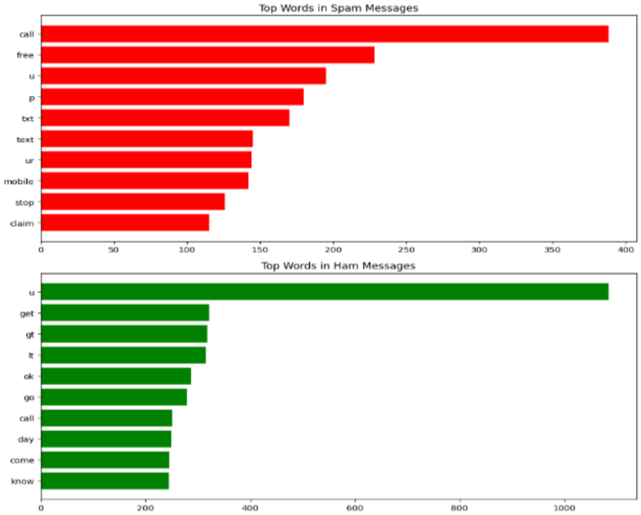


Fig.5 Distribution of count of most common words in spam and ham (business & personal) messages

A deeper examination of the ham subset indicates that business messages generally contain words like invoice, schedule, client, and delivery, whereas personal messages are marked by words like love, home, dinner, and friend. A word cloud image gives a graphical view of the contrast in thematic focus in the two classes, and it illustrates the effectiveness of the second-stage classifier.

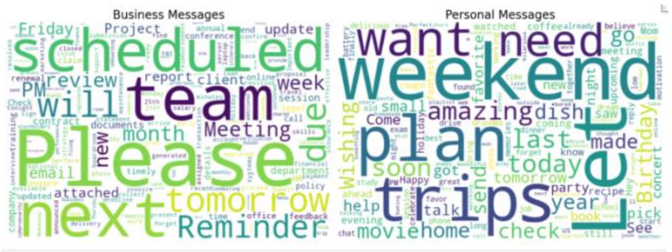


Fig.6 Word clouds for business vs personal messages showing topic-specific vocabulary

B. Model Performance:

The below is the model comparison utilized in both halves of the system:

Table 1. Classification report Spam Detection and Ham Classification

Model	**Accuracy**
TF-IDF + Logistic Regression	**0.96**
Multinomial Naive Bayes (MNB)	**0.94**
Decision Tree Classifier	**0.98**
Random Forest Classifier	0.99
Voting Classifier (MNB + DT)	0.98

- Random Forest classifier worked best as a standalone model for spam filtering since it can generalize and handle biased data.
- In ham business vs. personal classification, logistic regression achieved the best F1 score and also had easier interpretation.
- Voting Classifier, whose combined strength of Decision Tree and Naive Bayes outperforms individual models, thereby validating the strength of ensemble algorithms in text classification.

These results validate that ensemble models enhance the stability of anti-spam detection systems. Also, clean preprocessing and quality TF-IDF features significantly enhance performance The high F1-scores and low standard deviations across models validate the stability and reliability of the system suggested.

V. CONCLUSION

This paper suggested a two-stage machine learning model for spam classification of SMS: the first stage classifies ham or spam and the second stage classifies ham messages as business or personal. Out of the models that were experimented, Multinomial Naive Bayes and Decision Tree Voting Classifier identified spam with the highest mean F1-score of 0.98, and Logistic Regression identified ham messages with the highest score of 0.976.

The results confirm the effectiveness of blending conventional machine learning algorithms with TF-IDF feature engineering in high reliability and accuracy for real-world text classification problems. The system is robust across datasets and models, and it generates meaningful information to be exploited such as intelligent SMS filtering, customer communication sorting, and cellular message triage.

It can also be utilized to continue follow-up research that can investigate the application of deep learning methods like LSTMs or transformers to increase context sensitivity and scaling of multilingual SMS large-scale datasets for improving generalizability. Improved model interpretability and real-time deployability can also help to make it a part of mobile operating systems and communication platforms.

VI. References

[1] Almeida, T.A., Hidalgo, J.M.G., & Yamakami, A. (2011). Contributions to the study of SMS spam filtering: new collection and results. Proceedings of the 11th ACM symposium on Document engineering..

[2] McCallum, A., & Nigam, K. (1998). A comparison of event models for Naive Bayes text classification. AAAI-98 workshop on learning for text categorization, 752(1), 41–48.

[3] Scikit-learn Documentation. [Online]. Available: https://scikit-learn.org/stable/

[4] NLTK Documentation. [Online]. Available: https://www.nltk.org/

[5] TF-IDF Vectorizer – Scikit-learn. [Online]. Available: https://scikit-learn.org/stable/modules/generated/sklearn.feature_extraction.text.TfidfVectorizer.html

[6] Random Forest Classifier – Scikit-learn. [Online]. Available: https://scikit-learn.org/stable/modules/generated/sklearn.ensemble.RandomForestClassifier.html

[7] Learning Curve Analysis – Scikit-learn. [Online]. Available: https://scikit-learn.org/stable/auto_examples/model_selection/plot_learning_curve.html

[8] UCI SMS Spam Collection Dataset. [Online]. Available: https://archive.ics.uci.edu/ml/datasets/SMS+Spam+Collection

[9] Business/Personal Ham Dataset – Custom curated for this project.

[10] Vishwanath V. G., Nagesh H.R., "SMS Spam Detection Using Machine Learning Techniques."
Understanding: Emphasized the importance of text preprocessing (stopword removal, lemmatization) and machine learning algorithms like SVM, Random Forests for spam detection.

[11] Sahami et al., "Efficient Spam Filtering with Naive Bayes."
Understanding: Demonstrated the efficiency of simple probabilistic models like Naive Bayes for spam filtering, showing that textual features alone can provide strong spam/ham separation.