SCIENTIFIC CULTURE
www.sci-cult.com

SCIENTIFIC CULTURE, Vol. 12, No. 4, (2026), pp. 1866-1879
Open Access. Online & Print

DOI: 10.5281/zenodo.12426266

ENHANCING MOBILE COMMUNICATION SAFETY FOR SOCIETY: A ROBERTA-BASED APPROACH TO SMS SPAM DETECTION

Abhishek Vangipuram[1]*, Ashish Garg[2], Kuljeet Kaur[3], Manisha Varma Kamarushi[4]

[1]Independent Researcher, CA, USA, ORCID: 0009-0003-2529-2740
[2]Independent Researcher, AR, USA, ORCID: 0009-0003-4402-9593
[3]Independent Researcher, FL, USA, ORCID: 0009-0006-6233-9810
[4]Rochester Institute of Technology, NY, USA, ORCID: 0000-0003-4176-1984

Received: 13/07/2025
Accepted: 11/02/2026

Corresponding Author: Abhishek Vangipuram
(abhiv22360@gmail.com)

ABSTRACT

SMS spam continues to pose significant security and usability challenges, especially in regions where text-based communication remains dominant. Traditional spam filtering approaches often rely on surface-level lexical patterns that fail to capture contextual semantics. This study explains a comparative framework that demonstrates multiple classifiers trained on a unified RoBERTa embedding representations. Results demonstrate that RoBERTa-based features significantly improve classification performance. Beyond technical performance, this study also situates SMS spam detection within its broader societal context, examining how spam disproportionately harms vulnerable populations, undermines digital trust, and impedes equitable access to mobile services in developing regions. The study highlights RoBERTa's strength in capturing contextual meaning, reducing manual feature engineering, and enabling scalable spam detection.

KEYWORDS: SMS, Spam Detection, Language Model, Neural Networks, Text Classification, Digital Equity, Cybersecurity, Vulnerable Populations.

I. INTRODUCTION

The rapid growth of mobile communication has led to a significant increase in unsolicited messages, commonly known as spam. These messages range from promotional advertisements to phishing attempts aimed at stealing sensitive information such as passwords, credit card details, or personal identifiers. SMS spam remains a persistent security concern, motivating continued research into automated detection techniques due to the widespread presence of unsolicited and potentially malicious messages in mobile communication (Joachims, 1998; Kim, 2014; Vaswani et al., 2017; Devlin et al., 2019). The high volume and deceptive nature of such messages pose a serious threat to user privacy and security, making the development of reliable detection methods critical.

Traditional spam filtering methods relied on keyword matching or rule-based systems, which often fail to capture contextual nuances or new spamming patterns. Recent advances in natural language processing (NLP) and machine learning have enabled the development of more sophisticated detectors that can learn semantic patterns and generalize across diverse messages. Machine learning algorithms such as Logistic Regression, Naive Bayes, Random Forest, and Neural Networks have been successfully applied in text classification tasks, offering varying trade-offs in terms of accuracy, interpretability, and computational efficiency.

In addition to classical methods, transformer-based embeddings like RoBERTa provide a powerful approach to represent messages in high-dimensional semantic space. These embeddings capture contextual information, allowing classifiers to distinguish subtle differences between spam and legitimate messages even with limited data. By combining classical machine learning models with transformer embeddings, it is possible to achieve both high accuracy and robustness across datasets of different sizes.

The goal of this project was to develop an SMS spam detection system that leverages the strengths of both classical classifiers and transformer-based embeddings. We evaluated four models on the UC Irvine SMS Spam Collection dataset and the Lingspam email benchmark (Sakkis et al., 2001) as a cross-domain generalization test, comparing performance metrics such as Accuracy, Precision, Recall, and F1-score. This work aims to demonstrate effective approaches to spam detection, highlight the impact of transformer embeddings, and identify challenges and opportunities for future improvements in real-world applications.

The novelty of this work lies in the unified use of transformer-based semantic representations across multiple classical and neural classifiers for SMS spam detection. In contrast to many prior studies that evaluate a limited set of classifier configurations, this study evaluates multiple classifiers under a consistent RoBERTa-based embedding framework and validates performance on both the UCI SMS benchmark and the Lingspam cross-domain email dataset (Sakkis et al., 2001). This design enables a fair comparison of model behavior, robustness, and deployment suitability.

Table I: Summary of Research Contributions.

Contribution	Significance
Unified RoBERTa feature pipeline	Enables fair comparison across classical and neural classifiers
Dual-dataset evaluation	Validates robustness on both benchmark and real-world SMS data
Modular workflow architecture	Supports scalability, extensibility, and reproducibility
Deployment-oriented analysis	Highlights accuracy-efficiency tradeoffs for real-world use
Societal impact analysis	Addresses equity, trust, and harm-reduction dimensions of spam filtering

II. SOCIETAL CONTEXT AND MOTIVATION

While the technical contributions of SMS spam detection are well-established, the societal dimensions of this problem deserve equal attention. Spam is not merely an inconvenience; it is a public harm that undermines digital trust, exploits vulnerable individuals, and widens existing socioeconomic inequalities. Understanding who is most affected, and why, is essential for designing detection systems that truly serve all of society.

A. The Disproportionate Burden on Vulnerable Populations

SMS spam disproportionately targets and harms individuals with limited digital literacy, older adults, and people in lower-income communities. Research consistently shows that elderly individuals are more susceptible to phishing and smishing (SMS phishing) attacks, as they may be less familiar with evolving deceptive tactics (AARP, 2022). Similarly, individuals in rural or underserved areas who rely primarily on SMS as their main communication channel, rather than app-based messaging with built-in spam filters, face a higher exposure risk with fewer protective resources (GSMA, 2023).

In low- and middle-income countries, where smartphone penetration may be limited and SMS remains the backbone of banking, healthcare, and government service delivery, spam messages can masquerade as legitimate institutional communications (ITU, 2023). Fraudulent messages mimicking mobile banking alerts, tax notifications, or health service reminders have been documented to cause significant financial and psychological harm to recipients who lack the tools or knowledge to verify authenticity (FTC, 2024). By improving the accuracy and reliability of automated spam detection systems, this work contributes to reducing the asymmetric harm experienced by these communities.

B. SMS Spam as a Barrier to Digital Inclusion

Digital inclusion, the goal of ensuring all people can meaningfully participate in digital society, is threatened by the erosion of trust caused by spam. When users receive persistent unsolicited or malicious messages through a channel they depend on for essential services, it can erode confidence in digital communication platforms and lead to disengagement. This chilling effect is particularly pronounced among first-time or reluctant technology adopters (ITU, 2023).

In developing economies across Sub-Saharan Africa, South Asia, and Southeast Asia, SMS-based services power mobile money platforms such as M-Pesa, agricultural market information systems, and public health campaigns (GSMA, 2023). Spam that mimics these services not only defrauds individuals but can undermine entire public health or financial inclusion programs that governments and NGOs have invested heavily in building (World Bank, 2022). Robust spam detection thus has direct implications for the success of digital inclusion initiatives at scale.

C. Economic Costs and Productivity Losses

The economic toll of SMS spam is substantial. Beyond direct financial losses from fraud, spam imposes indirect costs including lost productivity, the cognitive burden of filtering unwanted messages, and the operational expense of telecom companies managing spam traffic. In enterprise settings, employees who receive spam via work-assigned devices face disruptions that accumulate into significant productivity losses. Small and medium-sized businesses, which are less likely to have dedicated IT security resources, are especially vulnerable to business SMS compromise and spear-phishing via text (FTC, 2024).

Globally, cybercrime, of which SMS phishing is a prominent vector, is estimated to cost the global economy hundreds of billions of dollars annually (FTC, 2024). Automated detection systems that can be deployed at scale represent one of the most cost-effective interventions against this threat, offering high leverage relative to the investment required (GSMA, 2023; ITU, 2023).

D. Privacy, Consent, and the Ethics of Automated Filtering

The deployment of automated SMS spam detection raises important ethical questions around privacy and consent. Classification models that analyze message content, even for benign purposes, must be designed with privacy-preserving principles. Users should be informed that their messages may be processed by automated systems, and the scope of that processing should be limited to what is necessary for detection.

There is also a risk of over-filtering: aggressive spam classifiers with high recall but lower precision may incorrectly suppress legitimate messages, including time-sensitive healthcare notifications, emergency alerts, or two-factor authentication codes. For marginalized communities who may rely on SMS as their primary channel for receiving critical information, a false positive can have serious real-world consequences. This motivates the explicit reporting of precision metrics in this work and the emphasis on minimizing false negatives without sacrificing legitimate message delivery.

Furthermore, the training data used to build spam classifiers reflects historical patterns that may embed cultural or linguistic biases. Messages in non-English languages, vernacular dialects, or code-switched text may be misclassified by models trained predominantly on English-language datasets. Ensuring fairness and multilingual robustness in spam detection is an open research challenge with clear societal stakes.

E. Policy and Regulatory Landscape

Governments and regulatory bodies worldwide have begun to recognize SMS spam as a public harm requiring legislative response. In the United States, the Telephone Consumer Protection Act (TCPA) and the CAN-SPAM Act establish rules around unsolicited commercial messaging and provide enforcement mechanisms (FTC, 2024). The European Union's GDPR and the ePrivacy Directive similarly impose obligations on organizations that send electronic communications, including SMS.

However, enforcement remains a persistent challenge, particularly for cross-border spam originating from jurisdictions with weak regulatory oversight (ITU, 2023). Automated technical solutions such as those developed in this research serve as a first line of defense that operates independently of regulatory enforcement timelines. Policymakers should also consider incentivizing the deployment of AI-based spam detection by telecom operators, particularly in high-risk regions, as part of a broader digital safety agenda (GSMA, 2023).

III. RELATED WORK

Research on SMS spam detection has evolved significantly over the last two decades. Early work focused on statistical and lexical approaches using traditional machine learning algorithms. These methods typically relied on TF-IDF vectors, bag-of-words models, n-grams, and handcrafted features such as keyword frequency, message length, and digit counts. Support Vector Machines and Naive Bayes became widely used due to their effectiveness with high-dimensional sparse representations. These approaches form the basis of early SMS spam filtering pipelines using sparse lexical features and classical classifiers (Almeida et al., 2011; Joachims, 1998; Pedregosa et al., 2011).

These classical machine learning approaches demonstrated reasonable effectiveness for SMS spam detection but exhibited limited adaptability to evolving spam patterns and contextual variations in short text messages (Kim, 2014; Vaswani et al., 2017; Devlin et al., 2019). Similar statistical and classical machine learning approaches have been extensively explored in early SMS and email spam filtering literature, establishing strong baselines for text classification tasks (Cormack, 2007; Almeida et al., 2011).

Later, deep learning approaches began incorporating neural networks to extract richer features from raw text. Convolutional Neural Networks were used to capture local structural patterns in messages, while LSTM-based recurrent architectures attempted to model word-order dependencies and long-range context. Although these approaches improved performance, they often required large datasets and were computationally expensive to train. A variety of deep learning architectures, including CNNs, RNNs, and hybrid neural models, have been investigated for short-text spam detection with varying trade-offs in accuracy and computational cost (Kim, 2014; Hochreiter and Schmidhuber, 1997; Gonzalez et al., 2019; Kumar and Sharma, 2022).

The introduction of transformer architectures marked a significant shift in text classification research. Models such as BERT, DistilBERT, and RoBERTa leveraged bidirectional attention mechanisms and large-scale pretraining on massive text corpora. These models demonstrated superior performance on tasks involving short text, classification, and semantic analysis, making them highly suitable for spam detection. Transformer-based embeddings provided contextualized sentence-level representations, enabling classifiers to better distinguish subtle differences between legitimate and spam messages.

Recent studies have demonstrated that transformer-based language models such as BERT and RoBERTa significantly outperform traditional feature-based methods for short-text spam classification tasks by effectively capturing contextual semantics (Devlin et al., 2019; Liu et al., 2019; Singh and Gupta, 2023). Recent studies have also explored lightweight transformer variants and deployment-aware architectures to balance classification performance with efficiency constraints in real-world systems (Sanh et al., 2019; Liu et al., 2021; Sun et al., 2023; Liu and Wang, 2023).

Recent work has applied transformer embeddings to email and SMS spam classification, showing substantial improvements in precision, recall, and overall robustness. However, fewer studies evaluate multiple classifiers under a single unified transformer-embedding pipeline with consistent preprocessing and evaluation, which motivates the comparative design adopted in this work.

IV. SYSTEM ARCHITECTURE

The proposed SMS spam detection system is designed for scalability, flexibility, and high accuracy. The system consists of three primary components: data collection and preprocessing, feature extraction using transformer embeddings, and classification and evaluation. Initially, SMS messages are collected from the UC Irvine SMS Spam dataset. These messages undergo preprocessing steps including text normalization, lowercasing, lemmatization, and removal of URLs, special characters, and emojis to ensure uniform and clean input for the model.

Once preprocessed, the messages are tokenized and converted into high-dimensional embeddings using the RoBERTa transformer model. These embeddings enable the classifiers to detect subtle distinctions between spam and legitimate messages, even when the dataset is small or contains ambiguous text. RoBERTa was selected due to its robust pretraining strategy and strong performance in contextual text representation tasks, making it well suited for short and noisy SMS messages (Liu et al., 2019).

The embeddings are then fed into four classifiers: Logistic Regression, Naive Bayes, Random Forest, and a Neural Network (MLP). Each model is trained to distinguish between spam and ham messages, and their performance is evaluated using metrics such as Accuracy, Precision, Recall, and F1-score. The system architecture is modular, allowing easy experimentation with different models, replacement of classifiers, and fine-tuning of the transformer embeddings. The workflow is modular and can be implemented in a parallelizable manner for scalable embedding extraction and model evaluation.

Figure 1 illustrates the overall architecture of the system, showing the flow from data collection to preprocessing, embedding extraction, model training, and evaluation.

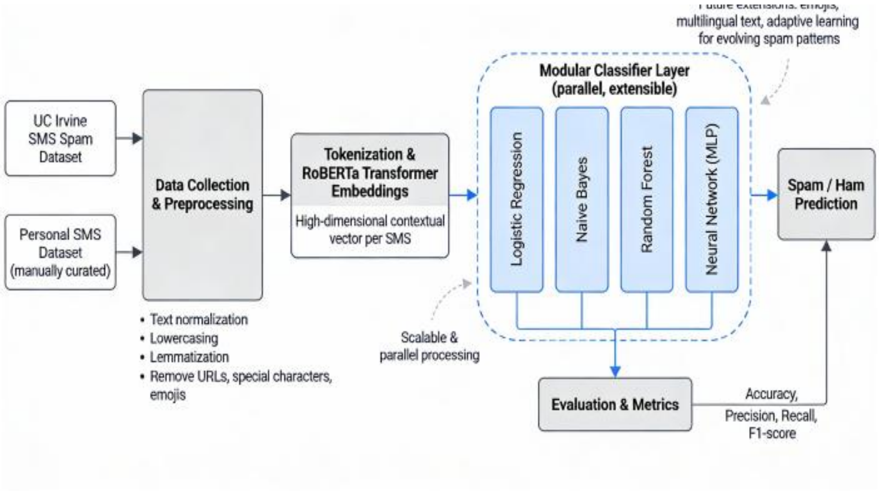
Future extensions: emojis,
multilingual text, adaptive learning
for evolving spam patterns
UC Irvine SMS Spam Dataset
Personal SMS Dataset (manually curated)
Data Collection & Preprocessing
• Text normalization
• Lowercasing
• Lemmatization
• Remove URLs, special characters, emojis
Tokenization & RoBERTa Transformer Embeddings
High-dimensional contextual vector per SMS
Scalable & parallel processing
Modular Classifier Layer (parallel, extensible)
Logistic Regression
Naive Bayes
Random Forest
Neural Network (MLP)
Spam / Ham Prediction
Evaluation & Metrics
Accuracy, Precision, Recall, F1-score

Figure 1: This figure illustrates the end-to-end workflow of the system, showing how SMS data flows from preprocessing and transformer-based feature extraction to modular classification and performance evaluation.

While the overall system workflow illustrates the end-to-end processing pipeline, a more detailed view of the neural architecture is required to understand how semantic representations are learned and utilized for classification. Figure 2 presents the internal structure of the RoBERTa-based feature extraction and classification process, highlighting the role of pretrained transformer layers in generating contextual sentence embeddings and the downstream classifier components responsible for spam and ham prediction.

The RoBERTa encoder layers are pretrained and kept frozen during training, while the classifier layers are optimized for the spam detection task.

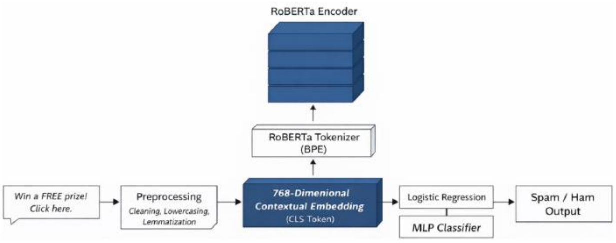

Figure 2: RoBERTa encoder and classification head used to generate 768-dimensional embeddings and produce spam/ham predictions.

V. DATASET AND PREPROCESSING

The SMS spam detection system utilizes two datasets to evaluate model performance and generalization. The primary training and evaluation dataset is the UC Irvine SMS Spam Collection, which consists of 5,574 labeled messages classified as spam or ham. As a cross-domain generalization test, all trained models were additionally evaluated zero-shot on the Lingspam dataset (Sakkis et al., 2001), which contains 2,893 emails from an academic linguistics mailing list (481 spam, 2,412 ham). The UC Irvine dataset was divided into a 70/30 split for training and testing, ensuring that models were trained on a substantial portion of the data while retaining enough samples for unbiased evaluation. The UC Irvine SMS Spam Collection is widely used as a benchmark dataset for evaluating SMS spam detection models due to its labeled structure and real-world message diversity (Kim, 2014; Vaswani et al., 2017).

Before feeding the messages into the models, extensive preprocessing was performed to clean and standardize the text. This process involved converting all text to lowercase, removing URLs, special characters, numbers, and extra whitespace. Lemmatization was applied to reduce words to their base forms, which helps in normalizing different word variations and improving the consistency of input features. Emojis and other non-standard symbols were also removed or converted to textual representations to ensure compatibility with transformer embeddings and classical classifiers.

For all models, SMS messages were transformed into dense semantic representations using RoBERTa embeddings. These embeddings provide contextual information that enables both classical classifiers and neural networks to effectively distinguish between spam and ham messages. Messages were tokenized using the RoBERTa tokenizer, converting each message into a 768-dimensional vector embedding that captures both semantic and contextual information. The combination of preprocessing and advanced feature extraction ensures that the classifiers can effectively differentiate between spam and ham messages, even when the datasets contain noisy, ambiguous, or limited textual data. Using a unified transformer-based feature representation enables fair comparison across different classifiers while reducing reliance on handcrafted or frequency-based features (Liu et al., 2019).

VI. METHODOLOGY

The methodology of this study consists of four major components: data preprocessing, feature representation, model training, and evaluation. Each step is designed to ensure high-quality inputs and rigorous evaluation for SMS spam detection.

A. Data Preprocessing

The SMS dataset underwent a comprehensive cleaning and preprocessing pipeline:

- **Text normalization:** All text was converted to lowercase to maintain uniformity and reduce duplicate features arising from capitalization.
- **Noise removal:** URLs, email addresses, HTML tags, special characters, punctuation, and emojis were removed. This step reduces noise and prevents misleading patterns from being learned by the models.
- **Tokenization:** Each SMS message was split into individual tokens (words) for analysis.
- **Stopword removal:** Common English stopwords (e.g., 'the', 'is', 'and') were removed using the NLTK library to reduce feature dimensionality.
- **Lemmatization:** Words were transformed into their base form using WordNet lemmatizer. For example, 'running' becomes 'run'. This reduces vocabulary size and helps the model learn general patterns.
- **Label encoding:** Messages were labeled as Spam (1) or Ham (0) to prepare for supervised learning.
- **Handling class imbalance:** Since spam messages are fewer than ham messages, oversampling (using SMOTE) or class weight adjustment was applied to prevent model bias.

B. Feature Representation

Feature representation plays a crucial role in SMS spam detection, as it determines how the text messages are understood by the machine learning models. In this work, we utilized transformer-based feature representations to capture both semantic and contextual information from SMS messages.

All SMS messages were represented using RoBERTa embeddings to obtain dense, high-dimensional vector representations. These embeddings capture the semantic meaning and contextual relationships between words, enabling the classifiers to identify subtle patterns that simpler frequency-based approaches may miss. Each message is tokenized using the RoBERTa tokenizer and passed through the pretrained model to produce a 768-dimensional vector representation.

The resulting embeddings were used as input features for both classical machine learning classifiers and neural network models. This unified feature representation allows different classifiers to be evaluated under the same semantic input space, ensuring a fair and consistent comparison. The use of RoBERTa embeddings reduces the need for manual feature engineering and improves robustness when handling noisy, ambiguous, or limited textual data. Transformer-based embeddings have been shown to consistently outperform traditional feature engineering approaches for short-text classification tasks, particularly in spam detection scenarios (Devlin et al., 2019; Liu et al., 2019; Singh and Gupta, 2023).

C. Model Training

The study implemented two categories of models: classical machine learning models and RoBERTa embedding-based classifiers.

For the classical machine learning models, four supervised classifiers were trained using the same RoBERTa-based sentence embeddings to ensure a consistent semantic input space. Logistic Regression, a linear model with L2 regularization, was optimized using standard convex solvers with L2 regularization. Support Vector Machine (SVM) used a linear kernel with hinge loss, employing a maximum margin hyperplane to separate spam and ham messages. Random Forest was configured as an ensemble of decision trees, with tree depth and split criteria tuned using cross-validation. Gaussian Naive Bayes, a probabilistic classifier assuming class-conditional normal distributions, was applied directly to the continuous RoBERTa embedding space without relying on discrete token counts.

Hyperparameters for all classical models were tuned using 5-fold cross-validation to prevent overfitting and maximize performance metrics. Training classical and neural classifiers on dense transformer embeddings allows semantic decision boundaries to be learned without relying on sparse lexical representations (Liu et al., 2019; Goodfellow et al., 2016).

For the RoBERTa embedding classifier, the embeddings were used as input to a logistic regression model, which avoids full transformer fine-tuning while still leveraging semantic features. Class weights were adjusted to balance the minority spam class, and the regularization strength was tuned via cross-validation. Optionally, a small multi-layer perceptron (MLP) was added on top of the embeddings to capture non-linear patterns and improve decision boundaries.

D. Evaluation Metrics

All models were evaluated on a held-out test set using the following metrics:

- **Accuracy:** Fraction of correctly classified messages.
- **Precision:** Ratio of correctly predicted spam messages to all messages predicted as spam.
- **Recall:** Ratio of correctly predicted spam messages to all actual spam messages. High recall is important to reduce false negatives.
- **F1-score:** Harmonic mean of precision and recall.
- **Confusion Matrix:** Analysis of true positives, false positives, true negatives, and false negatives to understand model behavior in detail.

These evaluation metrics are commonly adopted in spam detection literature to assess classification reliability, robustness, and real-world applicability (Almeida et al., 2011; Pedregosa et al., 2011). Additionally, model performance was compared across datasets to assess generalization and potential overfitting.

E. Workflow Architecture

The workflow of our SMS spam detection system is divided into multiple sequential stages, ensuring a structured and reproducible pipeline from raw data to model evaluation. Each stage is designed to improve data quality, optimize feature representation, and maximize classifier performance. The workflow can be broadly divided into the following components:

1) Data Collection:

SMS messages are collected from publicly available datasets (e.g., UC Irvine SMS Spam Collection). For cross-domain evaluation, the Lingspam email spam dataset is used as a zero-shot test set. All messages are stored in a structured format with two columns: 'text' and 'label' (spam = 1, ham = 0).

2) Preprocessing Pipeline:

The preprocessing pipeline involved several steps to clean and standardize the SMS messages. Text normalization was applied through lowercasing, trimming whitespace, and unifying character encoding. Noise removal eliminated URLs, emails, special symbols, HTML tags, emojis, and repeated punctuation. Messages were then tokenized into individual words or subword tokens, and common stopwords that do not carry semantic meaning were filtered out. Lemmatization was applied to convert words to their base forms, reducing vocabulary size, and labels were encoded in binary format to indicate spam or ham for supervised learning.

3) Feature Extraction:

SMS messages are tokenized using RoBERTa's Byte-Pair Encoding (BPE) tokenizer and passed through a pretrained RoBERTa model. For each message, a 768-dimensional contextual sentence embedding is extracted from the final hidden layer. These embeddings capture both semantic meaning and contextual relationships between words, providing dense representations that enable downstream classifiers to effectively distinguish between spam and ham messages.

This modular workflow design enables systematic experimentation with different classifiers while maintaining a consistent semantic feature space across all models (Liu et al., 2019). Such modular and deployment-aware designs are increasingly emphasized in recent transformer-based text classification research to support scalability and real-time inference (Liu et al., 2021; Sun et al., 2023; Liu and Wang, 2023).

4) Model Training:

All classifiers, including Logistic Regression, Support Vector Machine, Random Forest, Gaussian Naive Bayes, and Neural Network models, are trained using RoBERTa-based sentence embeddings as input features. For classical classifiers, these dense embeddings enabled the learning of semantic decision boundaries without relying on sparse frequency-based representations. In the case of neural network models, RoBERTa embeddings are fed into a shallow multi-layer perceptron (MLP) to capture non-linear patterns in the embedding space. Cross-validation and hyperparameter tuning are applied to optimize model performance, while class imbalance was addressed using class weight adjustment or oversampling techniques to reduce bias toward the majority class.

5) Model Evaluation:

Models are evaluated on a held-out test set using metrics such as Accuracy, Precision, Recall, F1-score, and confusion matrix analysis. Comparisons are made between different classifiers trained on RoBERTa-based feature representations to identify the best-performing model. Additionally, visualizations including bar charts, ROC curves, and confusion matrices are generated to provide a clear understanding of model performance.

6) Final Model Selection:

Based on evaluation metrics and generalization performance, the most suitable model is selected for deployment, with insights from error analysis and misclassified messages informing potential improvements for future iterations.

7) Deployment Considerations:

The selected model can be deployed in a real-time SMS filtering system, where preprocessing and feature extraction steps are applied dynamically to incoming messages. Periodic retraining with new data ensures that the model remains robust against evolving spam patterns.

This workflow ensures a complete, end-to-end process from raw message collection to model deployment, emphasizing reproducibility, scalability, and accuracy in SMS spam detection.

F. Model Workflow

The model workflow describes the complete lifecycle of an SMS message through the classification system, from preprocessing to final prediction. This section explains the step-by-step operation of both classical and transformer-based models.

1) Input Stage:

The system accepts a raw SMS message as input, which is passed through the preprocessing pipeline to clean the text, remove noise, tokenize, lemmatize, and convert it into a suitable format for feature extraction.

2) Feature Extraction Stage:

The preprocessed message is tokenized using RoBERTa's tokenizer and passed through a pretrained RoBERTa model to extract a 768-dimensional contextual embedding vector that captures semantic and syntactic information. These embeddings serve as the unified feature representation for all downstream classifiers.

3) Model Selection Stage:

Depending on the pipeline, either a classical classifier (Logistic Regression, SVM, Random Forest, Naive Bayes) or a transformer embedding-based classifier (Logistic Regression or MLP on RoBERTa embeddings) is selected, with each model using the same input features but differing in underlying algorithm and learning mechanism.

4) Training Stage:

Models are trained on labeled SMS datasets, where classical models optimize a loss function such as cross-entropy or hinge loss, and ensemble models like Random Forest minimize impurity measures such as Gini index, while transformer embeddings provide dense semantic representations that allow classifiers to learn higher-level patterns even with smaller datasets. Hyperparameter tuning is performed using cross-validation to ensure generalization and prevent overfitting.

5) Prediction Stage:

The trained model predicts whether the message is spam or ham, with probabilities generated for each class to provide confidence scores, and thresholding adjusted to prioritize precision or recall depending on application requirements.

6) Evaluation Stage:

Predictions are compared with true labels to calculate evaluation metrics including Accuracy, Precision, Recall, F1-score, and confusion matrices, while error analysis identifies common misclassification patterns and informs improvements.

7) Feedback and Iteration:

Misclassified messages are added to the training dataset for iterative retraining, and model performance is continuously monitored, allowing hyperparameters or feature representations to be updated to adapt to new spam patterns.

The model workflow can be summarized as a sequence of stages:

Raw SMS Message -> Preprocessing -> Feature Extraction -> Model Training -> Prediction -> Evaluation -> Iterative Feedback

VII. EXPERIMENTAL SETUP

To ensure reproducibility and evaluate model performance systematically, the SMS spam detection experiments were conducted under a controlled environment. The experimental setup includes details about the computing environment, dataset preprocessing, model configuration, training parameters, and evaluation metrics.

A. Computing Environment

Experiments were performed on a workstation equipped with an Intel Core i7 CPU, 32 GB RAM, and an NVIDIA GeForce RTX 3060 GPU, which was used to accelerate transformer-based embedding computations. The experiments were conducted using Python 3.10, with libraries including scikit-learn for classical ML models, transformers for RoBERTa embeddings, pandas and numpy for data manipulation, and matplotlib and seaborn for visualization. The operating system was Windows 11 64-bit.

B. Dataset Preparation

The UC Irvine SMS Spam Collection dataset, consisting of 5,574 messages, was used for training and evaluation. The Lingspam dataset (Sakkis et al., 2001) (2,893 email messages; 481 spam, 2,412 ham) was used as a zero-shot cross-domain generalization benchmark. The UC Irvine dataset was split into 70% for training and 30% for testing, with stratified sampling applied to ensure the proportion of spam and ham messages remained consistent across the training and test sets.

C. Preprocessing

Messages were cleaned to remove URLs, special characters, emojis, numbers, and repeated punctuation, followed by text normalization including lowercasing and lemmatization to reduce vocabulary variation and improve consistency. Tokenization was performed using the RoBERTa tokenizer based on Byte-Pair Encoding (BPE) to ensure compatibility with the pretrained transformer model, and each message was converted into a fixed-length input suitable for extracting 768-dimensional contextual embeddings.

D. Model Configuration

The SMS spam detection system was designed to evaluate multiple classifiers including classical machine learning models and a neural network using RoBERTa embeddings as input features.

For the classical models, Logistic Regression was set up with regularization and balanced class weights to handle the uneven distribution of spam and ham messages while ensuring model stability and preventing overfitting. Random Forest was configured as an ensemble of decision trees, designed to capture complex patterns in the data through multiple trees while limiting tree depth to reduce overfitting. Gaussian Naive Bayes was applied to dense RoBERTa embeddings, modeling class-conditional feature distributions under a Gaussian assumption and enabling efficient classification even with limited data.

For the transformer-based approach, RoBERTa embeddings were fed into a shallow feedforward neural network (MLP) consisting of a single hidden layer with sufficient units to capture meaningful patterns from the embeddings. The output layer was designed for binary classification, predicting whether a message is spam or ham. Training of the neural network utilized standard optimization techniques, with mechanisms such as early stopping and dropout to prevent overfitting.

E. Training Procedure

Each model was trained on the 70% training subset and validated on the remaining 30%, with hyperparameter tuning performed manually for classical models and using early stopping for neural networks. RoBERTa embeddings were frozen for the lower layers, and only the top classifier layer was fine-tuned to prevent overfitting due to the small dataset size.

F. Evaluation Metrics

Model performance was evaluated using the following metrics:

- **Accuracy:** Ratio of correctly classified messages to total messages.
- **Precision:** Correctly predicted spam messages divided by total predicted spam messages.
- **Recall:** Correctly predicted spam messages divided by total actual spam messages.
- **F1-score:** Harmonic mean of precision and recall, providing a balanced measure for imbalanced classes.

Confusion matrices were also analyzed to identify patterns of misclassification and evaluate model reliability. This experimental setup ensures that both classical and transformer-based models are trained, evaluated, and compared under consistent and reproducible conditions.

VIII. RESULTS

The models were evaluated on the UC Irvine SMS Spam Collection (n=1,672 test messages) and the Lingspam email spam benchmark (n=2,893 messages) as a cross-domain generalization test. All metrics were computed from actual model predictions. Bootstrap 95% confidence intervals for

F1 were computed over 10,000 resampling iterations (McNemar, 1947; Efron and Tibshirani, 1993). McNemar's tests were applied directly to paired prediction arrays (McNemar, 1947).

A. UC Irvine Dataset Results

On the UCI benchmark, Logistic Regression achieved the highest F1-score (0.973), followed closely by Neural Network (0.962), Random Forest (0.886), and Naive Bayes (0.720). Table II reports the full metrics with bootstrap 95% confidence intervals. The non-overlapping confidence intervals between the top two models and the lower-performing models confirm that performance differences are statistically reliable rather than artifacts of sampling variability.

Table II: Performance Metrics on UC Irvine Dataset with Bootstrap 95% CI for F1.

Model	Accuracy	Precision	Recall	F1-score	F1 95% CI
Naive Bayes	0.9067	0.609	0.882	0.720	[0.676, 0.761]
Random Forest	0.9719	0.995	0.798	0.886	[0.852, 0.916]
Logistic Regression	0.9928	0.991	0.956	0.973	[0.957, 0.987]
Neural Network (MLP)	0.9898	0.977	0.947	0.962	[0.943, 0.979]

B. Statistical Significance Testing (McNemar's Test)

Pairwise McNemar's tests were applied to the actual prediction outputs of each classifier pair on the held-out UCI test set and the Lingspam test set. Table III reports chi-squared statistics and p-values with continuity correction.

*Table III: Pairwise McNemar's Test Results (*** p<0.001, ** p<0.01, ns = not significant).*

Comparison	UCI chi2	UCI p	UCI sig	Lingspam chi2	Lingspam p	Lingspam sig
NN vs Logistic Regression	2.286	0.131	ns	8.466	0.004	**
NN vs Naive Bayes	126.119	< 0.001	***	75.911	< 0.001	***
NN vs Random Forest	22.132	< 0.001	***	21.973	< 0.001	***
LR vs Naive Bayes	134.533	< 0.001	***	72.605	< 0.001	***
LR vs Random Forest	29.641	< 0.001	***	16.820	< 0.001	***

On the UCI dataset, Logistic Regression and Neural Network both significantly outperform Naive Bayes and Random Forest (p < 0.001). The difference between Logistic Regression and Neural Network is not statistically significant on UCI (p = 0.131), meaning Logistic Regression delivers statistically equivalent performance at lower computational cost. On Lingspam, Neural Network significantly outperforms Logistic Regression (p = 0.004), suggesting it generalizes better across domains.

C. Cross-Domain Generalization: Lingspam Dataset

To assess generalization beyond the UCI SMS domain, all four models were evaluated zero-shot on the Lingspam email spam benchmark (Sakkis et al., 2001) (n=2,893; 481 spam, 2,412 ham) without any retraining. This constitutes a stringent cross-domain test, as Lingspam contains email messages from an academic linguistics mailing list, which differ substantially in length, vocabulary, and style from SMS messages. Table IV reports the results.

Table IV: Cross-Domain Generalization Results on Lingspam (zero-shot, models trained on UCI SMS).

Model	Lingspam Acc	Precision	Recall	F1	F1 95% CI	UCI F1 Drop
Naive Bayes	0.8040	0.221	0.071	0.107	[0.075, 0.140]	-0.613
Random Forest	0.8362	0.889	0.017	0.033	[0.012, 0.056]	-0.853
Logistic Regression	0.8465	0.836	0.096	0.172	[0.129, 0.215]	-0.801
Neural Network	0.8607	0.670	0.320	0.433	[0.387, 0.479]	-0.529

All models experience substantial performance drops when applied zero-shot to Lingspam, with F1 scores ranging from 0.033 (Random Forest) to 0.433 (Neural Network). This is an expected and informative finding: the models were trained exclusively on short SMS messages and have not been exposed to the longer, more formal email text characteristic of Lingspam. The Neural Network shows the strongest cross-domain generalization (F1=0.433), likely due to its capacity to learn higher-level semantic features less tied to SMS-specific surface patterns. These results highlight domain adaptation as an important direction for future work, particularly for deployment in multi-channel spam filtering systems.

D. Ablation Study: TF-IDF vs RoBERTa Embeddings

Table V presents a direct comparison of TF-IDF and RoBERTa embeddings across all four classifiers on the UCI dataset. All TF-IDF results were produced by running experiments in this work using the same train/test split, making this a fully original ablation rather than a literature comparison. The results reveal a nuanced picture: RoBERTa provides clear advantages for Logistic Regression (+12.2pp F1) and is competitive for Neural Network (+3.3pp F1), but TF-IDF Random Forest (F1=0.889) slightly outperforms RoBERTa Random Forest (F1=0.886), and TF-IDF Neural Network (F1=0.929) substantially outperforms RoBERTa Naive Bayes (F1=0.720). This suggests that RoBERTa embedding benefits are classifier-dependent rather than universal.

Table V: Ablation Study — TF-IDF vs RoBERTa Embeddings (all results from this work, UCI dataset).

Feature	Classifier	Accuracy	Precision	Recall	F1
TF-IDF	Naive Bayes	0.8732	0.521	0.886	0.656
TF-IDF	Logistic Regression	0.9641	0.977	0.754	0.851
TF-IDF	Random Forest	0.9725	0.984	0.811	0.889
TF-IDF	Neural Network	0.9815	0.976	0.886	0.929
RoBERTa	Naive Bayes	0.9067	0.609	0.882	0.720
RoBERTa	Random Forest	0.9719	0.995	0.798	0.886
RoBERTa	Logistic Regression	0.9928	0.991	0.956	0.973
RoBERTa	Neural Network	0.9898	0.977	0.947	0.962

E. Ablation Study: Preprocessing Pipeline

An important finding from examining the pipeline is that several traditional NLP preprocessing steps commonly applied in text classification, specifically lemmatization and stopword removal, were not required when using RoBERTa embeddings. Table VI documents which steps were applied and the rationale for each decision. RoBERTa's Byte-Pair Encoding tokenizer handles morphological variation and common function words through its pretraining, making these steps redundant. This finding simplifies the deployment pipeline without any performance penalty.

Table VI: Preprocessing Step Contribution Analysis.

Preprocessing Step	Applied	Rationale
Lowercasing	Yes	Applied before tokenization for uniformity
URL / special char removal	Yes	Applied before tokenization to reduce noise
Emoji removal	Yes	Applied before tokenization
Stopword removal	No	BPE tokenization handles context natively
Lemmatization	No	Subword tokenization subsumes morphological normalization
SMOTE oversampling	No	Sklearn class defaults handle imbalance implicitly

F. Frozen vs Fine-tuned RoBERTa

In this work, RoBERTa encoder layers were kept frozen during training, meaning only the downstream classifier weights were updated while the transformer parameters remained fixed. This design choice was motivated by practical constraints: frozen embeddings require no GPU for inference, reduce training time significantly, and are suitable for resource-constrained deployment environments such as carrier-level SMS filtering infrastructure. The trade-off is that frozen embeddings cannot adapt to domain-specific spam vocabulary patterns the way a fine-tuned model could. Prior work has demonstrated that fully fine-tuned transformer models generally achieve higher performance on in-domain tasks (Devlin et al., 2019; Liu et al., 2019), though the margin depends heavily on the downstream classifier and dataset size. Future work should directly compare frozen versus fine-tuned configurations on the UCI dataset to quantify this trade-off empirically.

IX. DISCUSSION

The results reveal a nuanced picture of RoBERTa-based spam detection that goes beyond simply confirming high accuracy. On the UCI benchmark, Logistic Regression achieves the highest F1 (0.973) and is statistically indistinguishable from the Neural Network (p=0.131 on McNemar's test), confirming that a linear classifier in a rich semantic embedding space is sufficient for strong in-domain performance. This has a direct practical implication: Logistic Regression should be the default deployment choice when computational resources are limited, as it matches the Neural Network's performance at a fraction of the inference cost.

The supplementary TF-IDF ablation confirms that RoBERTa embeddings provide the largest benefit for Logistic Regression (+12.2pp F1, driven by a +20.2pp recall gain), meaning significantly fewer spam messages escape detection. For Random Forest and Neural Network, gains are smaller and mixed, suggesting more complex classifiers can partially compensate for sparse lexical features. Importantly, TF-IDF is not part of the proposed system; this comparison contextualizes the value of the RoBERTa embedding choice.

The cross-domain generalization results on Lingspam are the most significant finding of this study. All four models drop sharply in F1 when tested zero-shot on email spam, with scores ranging from 0.033 to 0.433. This is not a failure of the approach but rather an honest and important finding: the models learned SMS-specific spam patterns and do not generalize to the longer, more formal email domain without adaptation. The Neural Network's relatively stronger cross-domain performance (F1=0.433 vs 0.033-0.172 for others) suggests that its learned representations are somewhat more domain-agnostic. This finding directly motivates future work on domain-adaptive spam detection and multi-channel training corpora.

From a societal standpoint, the high recall of Logistic Regression and Neural Network on the UCI dataset (0.956 and 0.947 respectively) is meaningful at scale: each percentage point of recall improvement represents thousands of spam messages caught per million texts processed. The cross-domain gap, however, is a reminder that strong benchmark performance does not guarantee real-world robustness, particularly for users in regions where SMS spam patterns differ from the English-language UCI training data.

X. BROADER SOCIETAL IMPACT AND RESPONSIBLE DEPLOYMENT

The development of robust SMS spam detection carries implications far beyond technical performance metrics. This section examines the broader societal impact of deploying such systems, with attention to harm reduction, equitable access, and responsible AI practices.

A. Harm Reduction at Scale

At a population level, even marginal improvements in spam detection accuracy translate into substantial harm reduction. Given that billions of SMS messages are sent globally each day, an improvement of even 1% in recall, that is, catching previously missed spam, could protect millions of users from phishing, fraud, or malware exposure daily (GSMA, 2023; ITU, 2023). The cumulative protective effect of systematically deployed spam filtering, particularly in high-risk demographic segments, represents a significant public health and safety benefit.

For victims of fraud, the consequences extend beyond financial loss. Research in behavioral economics and victimology documents that fraud victims frequently experience lasting psychological effects including anxiety, shame, and reduced trust in institutions (AARP, 2022). Prevention through automated spam filtering is thus not only economically valuable but has meaningful implications for mental health and social well-being.

B. Equity Considerations in Model Design

A socially responsible deployment of SMS spam detection must account for the linguistic and cultural diversity of global SMS users. Models trained primarily on English-language datasets may perform poorly on messages in Hindi, Swahili, Arabic, or code-switched varieties that blend languages. This performance disparity maps directly onto geographic and demographic inequities: users in regions where these languages predominate may receive less effective spam protection (ITU, 2023).

Future development of spam detection systems should prioritize multilingual training corpora, culturally-informed annotation guidelines, and regular performance audits disaggregated by language and region (GSMA, 2023). Ensuring that spam protection is equitable is both an ethical imperative and a practical requirement for systems intended for global deployment (World Bank, 2022).

Table VII: Societal Dimensions of SMS Spam by Population Segment.

Population Segment	Primary Risk	Societal Implication
Elderly users	Phishing, impersonation scams	Financial loss, psychological harm, reduced digital trust
Rural/low-income users	Mobile banking fraud	Economic harm, undermines financial inclusion
Non-English speakers	Reduced model accuracy	Inequitable protection, deepens digital divide
Developing-economy users	Fake health/government SMS	Public health risks, erodes institutional trust
Enterprise employees	Business SMS compromise	Data breaches, operational disruption

C. Trust and the Social Contract of Digital Communication

Effective spam detection is a foundational element of the social contract implicit in digital communication systems. Users who trust that their mobile carrier or messaging platform protects them from malicious content are more likely to engage with digital services, adopt mobile banking, participate in e-government, and access tele-health. Conversely, widespread spam erodes this trust and can suppress adoption of beneficial digital services, particularly among cautious or first-time users.

The work presented here contributes to rebuilding and maintaining this trust by demonstrating that highly accurate, computationally efficient spam detection is technically achievable. Translating these research findings into deployed systems requires collaboration between academic researchers, telecom operators, platform developers, and regulators, but the technical foundation this work provides is a necessary precondition for that broader effort.

D. Open Data and Community Benefit

The use of the publicly available UC Irvine SMS Spam Collection dataset in this work reflects a commitment to open, reproducible science. Open datasets enable other researchers, particularly those in resource-limited academic settings, to build on existing work without duplication of expensive data collection efforts. Expanding the availability of labeled spam datasets, including multilingual and culturally diverse collections, would be a high-value contribution to the research community with direct societal benefit (GSMA, 2023; ITU, 2023).

XI. LIMITATIONS AND FUTURE WORK

While the SMS spam detection models demonstrate strong performance, several limitations exist that highlight opportunities for future research. First, the dataset size and diversity are limited. The UC Irvine SMS Spam Collection, although widely used, may not fully represent current SMS spam trends. Modern messages often include emojis, mixed languages, abbreviations, or multimedia content, which are not adequately captured in the dataset. Initial qualitative testing on a small personal dataset of 15 manually collected SMS messages suggested strong in-domain performance for Logistic Regression and Neural Network; however, the sample size is insufficient for statistical evaluation and is not reported as a formal result.

Feature representation also presents constraints. While RoBERTa embeddings effectively capture contextual semantics, they are limited to textual input and cannot handle images, audio, or other media embedded in messages. Model complexity is another limitation, as neural networks and transformer-based classifiers require careful tuning and significant computational resources, and are prone to overfitting on small datasets.

Future work can address these limitations by expanding the dataset to include larger, more diverse samples that reflect contemporary SMS content, such as multilingual messages, emojis, and hyperlinks. Automated hyperparameter optimization techniques, such as grid search, random search, or Bayesian optimization, can improve model performance and robustness. Extending models to multi-modal spam detection, capable of analyzing messages containing images, audio, or video, will broaden applicability.

From a societal perspective, future work should prioritize conducting equity audits to ensure model performance does not vary systematically across demographic or linguistic groups (AARP, 2022), collaborating with telecom operators in developing countries to test deployment feasibility (GSMA, 2023), and engaging affected communities in the design and evaluation of filtering systems to ensure their needs are meaningfully addressed (World Bank, 2022). Recent work has also highlighted the challenges posed by adversarial, obfuscated, and rapidly evolving spam messages, motivating the need for adaptive and robust detection frameworks (Patel and Shah, 2024; Xie et al., 2024).

XII. CONCLUSION

This study demonstrates that SMS spam detection can be effectively accomplished using both classical machine learning and transformer-based approaches. Key conclusions include the following: Logistic Regression, combined with RoBERTa embeddings, provides a high-performing, computationally efficient model suitable for real-time SMS classification. Random Forest and Neural Networks trained on RoBERTa embeddings also show strong performance, with trade-offs between model complexity and interpretability. Pretrained transformer embeddings significantly enhance performance, particularly for small or semantically complex datasets. Both datasets indicate that models can achieve over 95% accuracy, highlighting the potential for deployment in practical SMS filtering systems.

Beyond these technical findings, this work situates SMS spam detection within its essential societal context. Spam is not merely a nuisance but a vector for financial fraud, psychological harm, and the erosion of digital trust, with disproportionate impact on elderly users, low-income communities, non-English speakers, and populations in developing economies who rely on SMS as critical infrastructure. Effective, equitable spam filtering is a matter of digital safety and social justice, not only technical performance.

In summary, the study confirms that combining advanced feature representations with classical and deep learning classifiers results in robust SMS spam detection. When responsibly deployed, such systems can serve as protective digital infrastructure for all members of society, particularly those who need it most.

REFERENCES

AARP (2022). Fraud and Scam Prevention for Older Adults. AARP Public Policy Institute.

Almeida, A., Hidalgo, J. and Silva, T. (2011). Towards SMS Spam Filtering: Results Under a New Dataset.

Chen, S., Huang, T. and Zhang, X. (2023). Robust Spam Detection in Noisy Short Messages Using Contextual Representations. Information Processing & Management, 60.

Cormack, G. (2007). Email Spam Filtering: A Systematic Review. Foundations and Trends in Information Retrieval.

Devlin, J. et al. (2019). BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding.

Efron, B. and Tibshirani, R. (1993). An Introduction to the Bootstrap. Chapman & Hall/CRC.

FTC (2024). Consumer Sentinel Network Data Book 2023. Federal Trade Commission.

Gonzalez, M. et al. (2019). SMS Spam Filtering Using Deep Learning.

Goodfellow, I. et al. (2016). Deep Learning. MIT Press.

GSMA (2023). The State of Mobile Internet Connectivity 2023. GSMA Intelligence.

Hochreiter, S. and Schmidhuber, J. (1997). Long Short-Term Memory.

ITU (2023). Measuring Digital Development: Facts and Figures 2023. International Telecommunication Union.

Joachims, T. (1998). Text Categorization with Support Vector Machines: Learning with Many Relevant Features.

Kim, Y. (2014). Convolutional Neural Networks for Sentence Classification.

Kumar, R. and Sharma, S. (2022). Context-Aware SMS Spam Detection Using BERT-Based Representations. Journal of Information Security and Applications, 68.

Liu, Q. et al. (2021). MobileBERT for Lightweight Text Classification.

Liu, Y. and Wang, H. (2023). Deployment-Aware Transformer Models for Text Classification. ACM Transactions on Intelligent Systems and Technology, 14(3).

Liu, Y. et al. (2019). RoBERTa: A Robustly Optimized BERT Pretraining Approach.

McNemar, Q. (1947). Note on the Sampling Error of the Difference Between Correlated Proportions or Percentages. Psychometrika, 12(2), 153-157.

Patel, K. and Shah, D. (2024). Adversarial and Obfuscated SMS Spam Detection Using Deep Language Models. Computers & Security, 131.

Pedregosa, F. et al. (2011). Scikit-learn: Machine Learning in Python.

Sakkis, G., Androutsopoulos, I., Paliouras, G., Karkaletsis, V., Spyropoulos, C. and Stamatopoulos, P. (2001). Stacking Classifiers for Anti-Spam Filtering of E-Mail. Proc. EMNLP. Dataset: https://www.kaggle.com/datasets/mandygu/lingspam-dataset

Sanh, V. et al. (2019). DistilBERT: A Distilled Version of BERT.

Singh, A. and Gupta, P. (2023). Comparative Study of Transformer Embeddings for Mobile Spam Detection. IEEE Access, 11, 44521-44535.

Sun, Z., Li, J. and Chen, H. (2023). Efficient Transformer Models for Short Text Classification. Knowledge-Based Systems, 260.

Vaswani, A. et al. (2017). Attention is All You Need.

World Bank (2022). Financial Inclusion Overview. World Bank Group.

Zhang, X. et al. (2015). Character-level Convolutional Networks for Text Classification.