Publishing Excellence Inspiring Progress
ARISTON PUBLICATIONS
CompSci & AI Advances
ISSN: 3066-3806 https://aristonpubs.com/journal/compsci-ai-advances https://doi.org/10.69626/cai.2026.0317

RESEARCH ARTICLE

A Comprehensive Evaluation of Text Classification Methods for Phishing and Smishing Detection Using TF–IDF and Semantic Sentence Embeddings

Annasaheb M. Chougule [1,*], Kavita S. Oza [1], Rohit B. Diwane [1], Vishal T. Patil [2]

ABSTRACT: The increasing prevalence of phishing and smishing attacks demands robust automated detection systems capable of identifying malicious content across multiple communication channels. This study presents a systematic comparative analysis of text classification techniques for detecting these threats, evaluating traditional TF-IDF vectorization against modern sentence embedding representations. We constructed a novel dataset comprising 27,000 carefully annotated messages collected from cyber cell sources in India, supplemented with 2,000 synthetically generated records to enhance diversity and model generalization. A comprehensive set of eleven machine learning classifiers was evaluated under both feature extraction paradigms, with particular attention paid to generalization behavior, model stability, and the training-testing performance gap. Under TF-IDF representation, the Support Vector Classifier demonstrated reliable performance with a training accuracy of 97.87% and a test accuracy of 89.37%, indicating controlled generalization. The Multilayer Perceptron and Stochastic Gradient Descent classifiers achieved competitive test accuracies of 88.28% and 86.06% respectively. Sentence embeddings, which capture deeper semantic context and linguistic nuance, yielded consistently superior results across all classifiers, with the Support Vector Classifier achieving 94.69% test accuracy and the tuned Multilayer Perceptron reaching 92.70%. The K-Nearest Neighbors classifier followed closely at 90.27%. Tree-based models exhibited comparatively lower stability, with the Decision Tree Classifier achieving only 81.41% with TF-IDF and 73.10% with sentence embeddings. Unlike prior studies that examine traditional features or deep embeddings in isolation, this work provides a controlled, large-scale comparative framework that jointly considers feature paradigms, classifier families, and communication channels. The findings underscore the advantage of semantically enriched representations for identifying sophisticated social engineering threats and establish a benchmark for future research in NLP-based cybersecurity applications.

Keywords: Text Classification, Natural Language Processing, TF-IDF, Sentence Embedding, Phishing Detection, Smishing Detection

Received: 16 August 2026; Revised: 13 October 2026; Accepted: 28 October 2026; Published Online: 10 November 2026

1. INTRODUCTION

The digital ecosystem upon which modern society increasingly depends has simultaneously become a fertile ground for malicious actors, with social engineering attacks representing one of the most persistent and evolving threats to information security. Among these, phishing and smishing have emerged as particularly prevalent and damaging forms of cybercrime, exploiting both human psychology and technological vulnerabilities to compromise sensitive

[1] Department of Computer Science, Shivaji University, Kolhapur, Vidya Nagar, Ujalaiwadi, Maharashtra 416004, India
[2] Department Of Computer Science, Miraj Mahavidyalaya, Shivaji University, Miraj-416436, Maharashtra, India

* Author to whom correspondence should be addressed: amc.rs.csd@unishivaji.ac.in (Annasaheb M. Chougule)

© Ariston Publications 2026. All rights reserved.

personal and financial information. Phishing attacks, which typically manifest through deceptive emails, and their SMS-based counterpart, smishing, have grown in sophistication and volume, posing significant risks to privacy, financial stability, and organizational reputation. The proliferation of internet-based communication channels has not only facilitated the expansion of these threats but has also rendered traditional security measures increasingly inadequate, as attackers continuously adapt their strategies to bypass conventional detection mechanisms [1, 2].

The detection and prevention of phishing and smishing attacks have consequently become critical areas of research within the cybersecurity domain, attracting substantial attention from both academic and industry practitioners. Early approaches to threat detection largely relied on rule-based systems and blacklisting techniques, which, while effective against known threats, proved insufficient against the dynamic and increasingly sophisticated nature of modern attacks [3, 4]. This limitation prompted a significant shift toward machine learning-based approaches, which offered the promise of adaptive and generalizable detection capabilities. Researchers have explored a wide spectrum of algorithms, ranging from traditional classifiers such as Support Vector Machines and Naive Bayes to more complex ensemble methods and neural network architectures, each demonstrating varying degrees of effectiveness across different datasets and threat scenarios [5-8].

A critical determinant of success in machine learning-based text classification is the choice of feature representation, which fundamentally influences the model's ability to capture relevant patterns in the data. Traditional approaches have predominantly employed sparse representations such as bag-of-words and Term Frequency-Inverse Document Frequency, which encode term importance based on statistical frequency within documents and across corpora. These methods have demonstrated considerable success, particularly in scenarios where lexical patterns are strong discriminators between classes, offering computational efficiency and interpretability [5, 8]. However, their inherent limitation lies in the inability to capture semantic relationships and contextual nuances, which are often crucial for distinguishing between sophisticated phishing attempts and legitimate communications that employ similar vocabulary [9].

The advent of deep learning has ushered in a new paradigm of dense semantic representations, including word and sentence embeddings, which encode textual meaning into continuous vector spaces. Models such as Word2Vec, GloVe, and more recently, transformer-based architectures like BERT and its variants, have revolutionized natural language processing by capturing contextual relationships and semantic similarities with unprecedented fidelity [10, 11]. Sentence embeddings, in particular, offer the advantage of representing entire messages as fixed-length vectors that preserve holistic meaning, making them particularly well-suited for short-text classification tasks such as phishing and smishing detection [12, 13]. These representations have demonstrated superior performance across numerous NLP benchmarks, suggesting their potential for enhancing cybersecurity applications.

Despite considerable progress in both feature engineering and classification algorithms, significant gaps persist in the existing literature. Many prior studies have examined phishing and smishing detection in isolation, focusing on either email-based or SMS-based threats rather than adopting an integrated approach that reflects the multi-channel nature of real-world social engineering campaigns [14, 15]. Furthermore, a substantial portion of research has prioritized the comparison of classification algorithms while giving comparatively less systematic attention to the impact of feature representation paradigms. The controlled comparison between traditional sparse representations and modern dense semantic representations across a diverse set of classifiers remains relatively underexplored, leaving an important gap in understanding how far classical machine learning models can be enhanced through improved text representations [16].

The novelty of this paper lies in its systematic, controlled, and large-scale comparative analysis of TF-IDF and sentence embedding representations for phishing and smishing detection, conducted on a purpose-built dataset that addresses the limitations of existing public corpora. Unlike previous studies that have often relied on small or narrowly scoped datasets, we construct and utilize a curated collection of 27,000 real-world messages sourced from cyber cells in India, supplemented with synthetic data to enhance diversity and generalization. Our evaluation encompasses eleven distinct machine learning classifiers under both feature paradigms, with particular emphasis on generalization behavior, model stability, and the training-testing performance gap, rather than focusing exclusively on peak accuracy. This comprehensive approach enables us to isolate the impact of feature engineering on detection performance and to identify the most effective combinations of representation and classifier for this critical cybersecurity task.

Existing research has employed machine learning for phishing detection with varying degrees of success. Several studies have demonstrated the effectiveness of SVM and ensemble methods for malicious email classification [1, 3], while others have explored deep learning approaches for enhanced detection capabilities [4, 10, 11]. In the smishing domain, researchers have applied Naive Bayes, Random Forest, and neural network-based models to detect SMS-based threats [6, 7, 14]. However, the vast majority of these studies have operated within a single feature paradigm, limiting their ability to draw comparative conclusions about representation effectiveness. Moreover, the absence of a unified framework for evaluating both phishing and smishing threats jointly, combined with the scarcity of large-scale, linguistically diverse datasets, constrains the generalizability of reported findings. The limited attention to generalization behavior and model stability further undermines confidence in the reliability of proposed models for real-world deployment.

The methodological approach of this study involves the

© Ariston Publications 2026. All rights reserved.

collection and preprocessing of a diverse corpus of phishing, smishing, and legitimate messages, followed by feature extraction using both TF-IDF and sentence embedding techniques. A comprehensive evaluation of eleven machine learning classifiers is conducted under both representations, with hyperparameter tuning performed to optimize performance. The analysis extends beyond accuracy reporting to examine precision, recall, F1-score, and the generalization gap between training and testing performance, providing a robust assessment of model behavior across representation spaces.

2. LITERATURE REVIEW

The detection of phishing and smishing attacks through text classification has attracted substantial research interest over the past decade, yielding a diverse body of literature that spans traditional machine learning approaches, deep learning architectures, and feature representation techniques. This section provides a comprehensive review of existing work in this domain, with particular emphasis on the methods, datasets, and models employed, while identifying the gaps that motivate the present study.

Early research in malicious text detection predominantly relied on traditional machine learning algorithms combined with sparse feature representations. Suryaningrum conducted a comparative study on hate speech classification using a Bahasa Indonesia dataset, demonstrating that TF-IDF vectorization consistently outperformed count vectorization, achieving an accuracy of 88.77% with Support Vector Machine classifiers [17]. This finding underscored the importance of term weighting schemes in capturing discriminative patterns within textual data. Similarly, Saputra investigated SMS spam detection using the TF-IDF approach on the SMS Spam Collection dataset, showing that appropriate preprocessing steps including tokenization, stemming, and stopword removal significantly enhanced classification accuracy [18]. These studies established TF-IDF as a reliable baseline for text classification tasks, particularly when computational efficiency and interpretability are prioritized.

The limitations of rule-based detection systems in adapting to evolving threat patterns prompted researchers to explore machine learning alternatives. A comprehensive review of malicious email detection techniques highlighted the effectiveness of Naïve Bayes, Support Vector Machines, and Random Forest classifiers for spam detection, while also noting the potential of deep learning approaches for phishing identification despite challenges related to adaptability and computational requirements [19]. Eze and Shamir further advanced this line of inquiry by investigating phishing email attacks enhanced by artificial intelligence, demonstrating that ensemble learning techniques combined with TF-IDF and deep learning features could substantially improve detection capabilities [20]. Their work illustrated the complementary nature of traditional and modern feature representations in addressing sophisticated threats.

In the specific domain of smishing detection, traditional machine learning models have demonstrated considerable promise. Kohilan and colleagues developed a machine learning-based framework for detecting smishing attacks at the end-user level, employing Naive Bayes and Support Vector Machine classifiers on a smishing dataset and achieving precision rates as high as 99.3% [21]. Their study identified the handling of hybrid languages and spelling errors as critical challenges requiring further investigation. Mishra and Soni implemented a neural network-based smishing detector that outperformed conventional methods, particularly when URL-based features were incorporated into the feature set [22]. Their findings suggested that deep learning approaches could achieve superior accuracy through the integration of multiple feature types. Awumee and colleagues introduced SmishShield, a machine learning-based smishing detection system specifically designed for the Ghanaian context, which utilized Random Forest and Naive Bayes classifiers to achieve high detection accuracy [23]. This work provided a foundational basis for region-specific smishing detection systems.

The evolution of natural language processing has increasingly shifted toward semantic representations that capture contextual meaning rather than mere lexical frequency. Kulshretha and Lodha conducted a performance evaluation of various word embedding algorithms, demonstrating that contextual models such as ELMo and BERT significantly advanced NLP capabilities by converting words into vectors that preserve semantic relationships and contextual dependencies [24]. Schopf, Schneider, and Matthes explored efficient domain adaptation of sentence embeddings using adapter modules, showing that this approach could reduce computational costs while maintaining performance by adding only a few trainable parameters per domain [25]. These developments have made sophisticated semantic representations more accessible for practical applications, including cybersecurity.

Hybrid deep learning architectures have recently gained considerable attention in phishing and spam detection research. Zouak and colleagues proposed a BERT-GraphSAGE hybrid framework that combines contextual text embeddings with graph-based relational learning, demonstrating improved spam detection performance and superior generalization across multiple datasets [26]. Their work highlighted the trend toward integrating semantic representation with structural learning to capture both content and relational patterns in malicious communications. Research on transformer-driven phishing email detection has shown that attention-based architectures consistently outperform traditional recurrent neural networks due to their superior ability to model contextual dependencies within email text [27]. These findings have reinforced the importance of semantically rich representations for identifying sophisticated phishing content that employs deceptive linguistic strategies.

Multilingual phishing detection has emerged as an increasingly critical research direction, reflecting the global nature of social engineering threats. An and colleagues

© Ariston Publications 2026. All rights reserved.

developed a multilingual email phishing detection system incorporating OSINT-derived features and machine learning techniques, demonstrating improved robustness across language variations [28]. Their work addressed the challenge of linguistic diversity in phishing campaigns, which often employ multiple languages to target different demographic groups. Studies on multilingual transformer models have explored cross-language phishing detection, indicating that embedding-based approaches are effective in handling linguistic diversity and may offer advantages over language-specific models [29]. These findings suggest that semantic representations can generalize across languages, making them particularly valuable for global cybersecurity applications.

The practical deployment of NLP-based phishing detection systems has also received attention, particularly in resource-constrained environments. Shirazi and colleagues evaluated the performance of NLP transformers for URL-based phishing detection on mobile platforms, demonstrating that transformer models can achieve strong detection performance while maintaining computational efficiency suitable for mobile devices [30]. This research reflected growing interest in practical deployment considerations, including inference latency, memory consumption, and energy efficiency. Beyond traditional text-based channels, voice and cross-modal phishing detection have also been explored, with recent work on multilingual transformer models for voice phishing detection demonstrating that semantic modeling is valuable even in non-traditional phishing channels [31]. These studies collectively indicate a clear research trend toward semantic embeddings, transformer architectures, and multilingual robustness.

Despite the extensive body of research, several significant gaps remain. Most existing studies focus on either phishing emails or smishing messages in isolation, despite the reality that social engineering attacks increasingly span multiple communication channels. The integrated analysis of both email-based and SMS-based threats within a unified framework remains limited. Furthermore, the majority of prior work has primarily compared classification algorithms while giving less systematic attention to the impact of feature representation paradigms. Controlled comparisons between traditional sparse representations such as TF-IDF and modern dense semantic representations including sentence embeddings across multiple classifiers are relatively scarce. Additionally, numerous studies rely on small or narrowly scoped public datasets that may not fully capture the linguistic diversity, hybrid language use, and evolving patterns of real phishing and smishing attacks. The lack of large-scale comparative studies that systematically evaluate classical machine learning classifiers in conjunction with modern embedding-based representations on mixed phishing-smishing datasets leaves an important gap in understanding how far traditional models can be enhanced through improved text representations. The present study addresses these gaps by constructing a large-scale dataset of 27,000 real and synthetic messages, conducting a controlled comparison between TF-IDF and sentence embedding representations across eleven classifiers, and analyzing generalization behavior and model stability alongside peak accuracy.

3. RESEARCH METHODOLOGY

The methodological framework of this study was designed to enable a rigorous and systematic comparison of text classification techniques for phishing and smishing detection, encompassing data collection and preprocessing, exploratory data analysis, feature extraction using both traditional and modern approaches, model training and hyperparameter optimization, and comprehensive evaluation with statistical validation. This section describes each component of the methodology in detail, providing sufficient information for replication and critical assessment.

3.1. Data Collection and Preprocessing

The foundation of this research rests upon a purpose-built dataset constructed to address the limitations of existing public corpora for phishing and smishing detection. The comprehensive dataset comprises 25,000 records collected from real-world sources through direct visits to Cyber Cells in Sangli, Kolhapur, and Pimpri-Chinchwad Pune, spanning messages from 2023 to 2025. These records encompass phishing emails, smishing SMS messages, and legitimate communications, providing a diverse representation of the linguistic patterns and deceptive strategies employed in social engineering attacks. To address potential data scarcity and enhance model generalization, an additional 2,000 synthetic records were generated using a generative AI model, resulting in a final dataset of 27,000 messages. The inclusion of synthetic data served to augment the diversity of the corpus and reduce the risk of overfitting during model training. The original data was collected in various raw formats, including screenshots and plain text messages, which were subsequently transformed into a structured dataset through systematic extraction and consolidation procedures.

The preprocessing pipeline consisted of several carefully designed steps to clean and normalize the textual data while preserving discriminative information essential for classification. The initial phase involved data cleaning, which included the removal of non-alphabetical characters, conversion of all text to lowercase to ensure case invariance, and handling of missing values through appropriate imputation or exclusion strategies. Tokenization was performed to segment the text into individual tokens, followed by the elimination of stopwords, with the notable exception of the word "not" for sentiment-related tasks where negation carries semantic significance. Stemming was applied using the Porter Stemmer, which was selected based on the predominance of English language content in the dataset, as confirmed by word cloud analysis. This choice ensured computational efficiency while maintaining sufficient linguistic normalization for effective feature

© Ariston Publications 2026. All rights reserved.

extraction. The dataset was subsequently balanced to ensure equal representation across all four classes, namely Non Smishing, Smishing, Phishing, and Non Phishing, with each class containing 6,750 messages. This balancing step was critical to prevent model bias toward majority classes and to improve the ability to correctly identify minority class instances, which are often the most consequential from a cybersecurity perspective.

All data annotation was performed by a team of two expert annotators from the cyber cell, ensuring domain expertise in the identification and classification of phishing and smishing content. The reliability of the annotation process was established through an initial agreement phase, wherein the annotators independently labeled a small, shared subset of the data before proceeding to annotate the full corpus independently. The high agreement rate of 98 percent on this initial subset demonstrated the reliability and consistency of the labeling process, ensuring that the ground truth for the dataset was accurate and could be reliably used for training and evaluating machine learning models. The final output of the preprocessing phase was a clean, balanced, and annotated corpus ready for feature extraction and subsequent modeling.

3.2. Exploratory Data Analysis

Exploratory data analysis was conducted to gain insights into the characteristics and distributions of the dataset, informing subsequent modeling decisions and providing interpretative context for the experimental results.

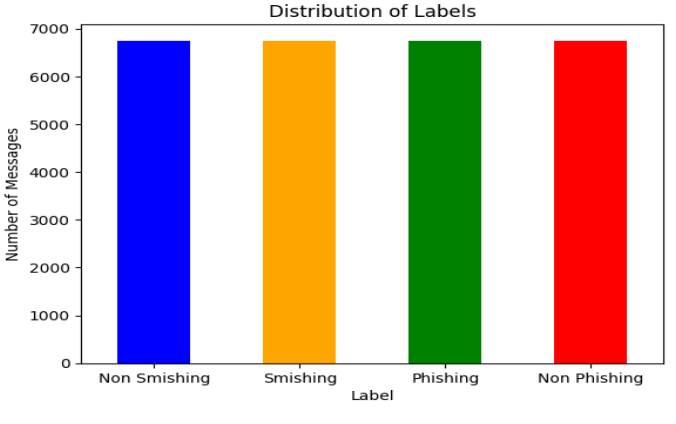

Fig. 1. Distribution of Labels - Bar chart showing equal frequency of 6,750 instances for each of the four classes: Non Smishing, Smishing, Phishing, and Non Phishing]

The distribution of labels across the dataset is presented in Figure 1, which illustrates the frequency of each message category. The distribution is perfectly balanced, with exactly 6,750 instances for each of the four classes: Non Smishing, Smishing, Phishing, and Non Phishing. This balanced configuration ensures that classification models trained on this dataset will not be biased toward any particular label, promoting fair and equitable performance across all threat categories.

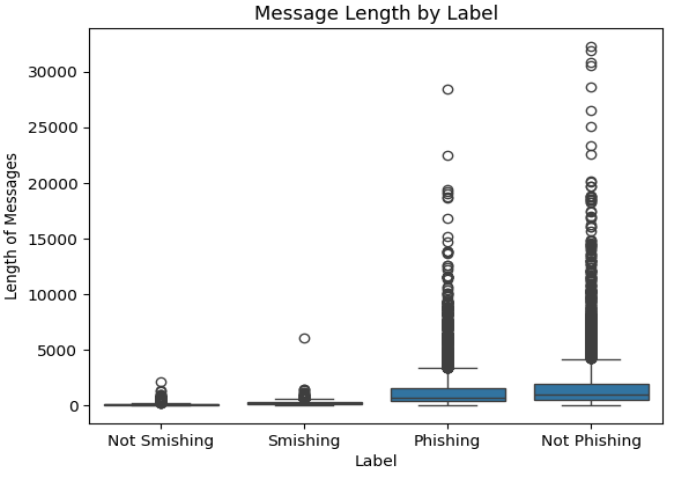

Fig. 2. Message length by label - Box plot showing distribution of message character counts across the four classes, with Phishing and Non Phishing showing greater variability and outliers.

Figure 2 presents a box plot comparing the distribution of message lengths across all four categories, revealing notable variations in text structure. Phishing and Non Phishing messages display significant variation in length, with outliers extending beyond 25,000 characters, indicating substantial diversity in the complexity and detail of email-based communications. In contrast, Non Smishing and Smishing messages are relatively shorter and exhibit less variability, suggesting a more uniform text structure typical of SMS-based communications. This observation has implications for feature extraction, as the shorter length of SMS messages may require different representation strategies compared to longer email content.

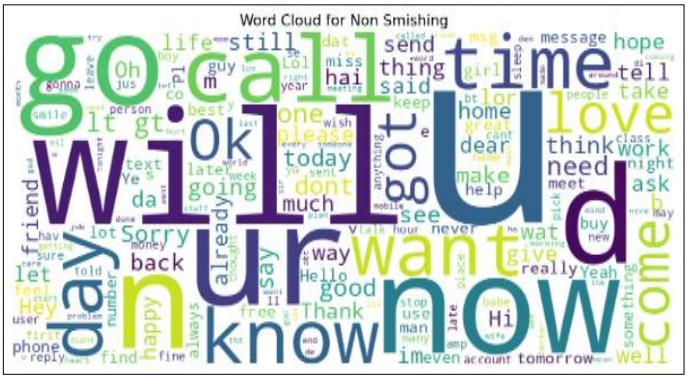

Fig. 3. Word Cloud for Non Smishing - Visualization showing prominent terms such as "ur," "will," "know," "call," and "love" in casual, conversational patterns.

© Ariston Publications 2026. All rights reserved.

Word cloud visualizations were generated for each message category to identify the most prominent terms and linguistic patterns. Figure 3 focuses on Non Smishing messages, where "ur" emerges as the most frequent term with over 450 occurrences, followed by "just," "know," and "like" with more than 275 occurrences each. Additional common terms such as "day," "good," "time," and "ok" suggest that these messages are casual and conversational in nature, reflecting typical personal communication. The accompanying word cloud reinforces this observation with prominent display of terms like "ur," "will," "know," "call," and "love," indicating the informal and friendly tone characteristic of legitimate non-malicious messages.

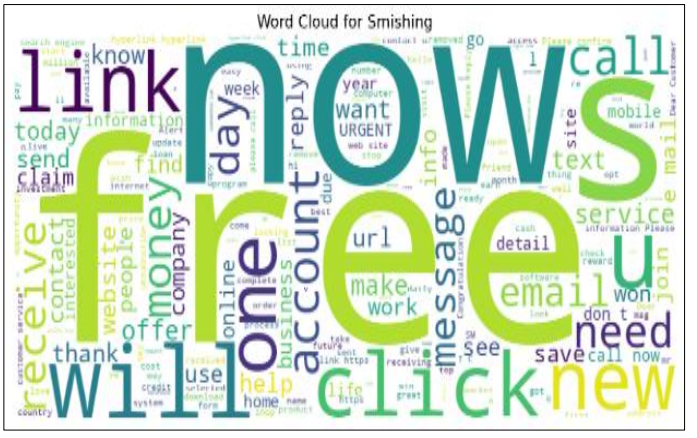

Fig. 4. Word Cloud for Smishing - Visualization showing prominent terms such as "now," "free," "com," "click," "link," "account," and "information" in urgent, manipulative patterns.

Figure 4 depicts the common vocabulary in Smishing messages, where "now" appears most frequently with nearly 2,000 occurrences, followed by "free" with approximately 1,500 occurrences, and "com" with over 1,250 occurrences. Terms like "click," "http," "link," "account," and "information" are highly prevalent, signaling the structured and urgent language typical of fraudulent attempts. The word cloud prominently displays these deceptive terms, highlighting the manipulative linguistic strategies employed in smishing attacks designed to prompt immediate victim responses, such as clicking malicious links or divulging sensitive personal information.

Figure 5 presents the most frequent words in Phishing messages, with "card" being the most used term at over 5,300 occurrences, followed closely by "atm," "company," and "com" at over 4,900 occurrences each. Other common terms including "information," "email," "address," "contact," and "delivery" point to a focus on identity theft and personal data harvesting, while terms like "payment," "bank," and "number" emphasize the financial intent behind these communications. These linguistic patterns reveal a formal yet deceptive tone characteristic of phishing communications designed to mimic legitimate institutional correspondence and prompt immediate action from recipients.

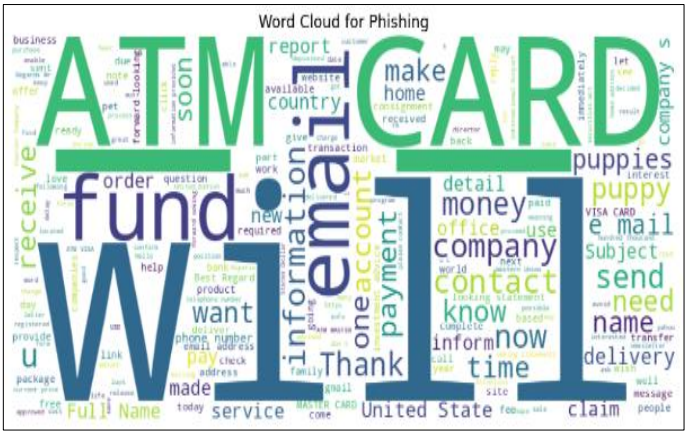

Fig. 5. Word Cloud for Phishing - Visualization showing prominent terms such as "card," "atm," "company," "information," "email," "payment," and "bank" in formal, deceptive patterns.

Figure 6 captures the common vocabulary in Non Phishing messages, where the word "isro" dominates with nearly 10,000 mentions, followed by "ect" with approximately 8,000 occurrences, and others such as "com," "subject," "ray," and "hou." These terms suggest official or academic correspondence, with the word cloud including entries like "university," "language," "research," and "information." This linguistic profile reflects a formal and informative communication style typical of legitimate organizational emails, serving as a useful contrast to the deceptive patterns observed in phishing messages.

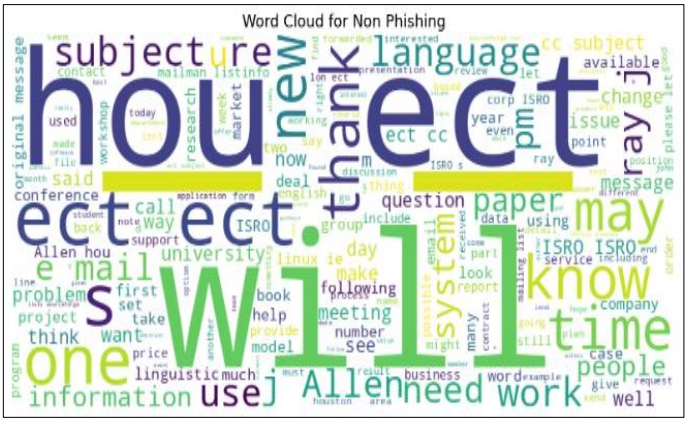

Fig. 6. Word Cloud for Non Phishing - Visualization showing prominent terms such as "isro," "ect," "university," "language," "research," and "information" in formal, informative patterns.

N-gram analysis was conducted to examine common word sequences that provide additional discriminative information beyond individual terms. Figure 7 displays the top 20 most common 2-grams in Non Smishing messages, with the most frequent bigram being "lt gt" at over 220 occurrences,

© Ariston Publications 2026. All rights reserved.

followed by informal expressions such as "let know," "good morning," "dont know," and "im going." These phrases confirm the casual and conversational nature of non-smishing texts, further supporting their classification as benign.

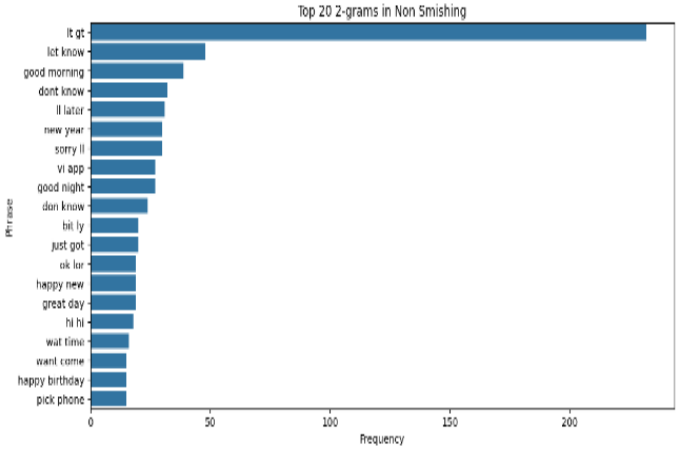

Fig. 7. Top 20 Two-Grams in Non Smishing - Bar chart showing most common bigrams including "lt gt," "let know," "good morning," and "dont know".

Figure 8 highlights the most frequent 2-grams in Smishing messages, with the sequence "1635465 1635465" appearing over 800 times, followed by bigrams such as "http www," "click link," "send money," and "personal information." These n-grams strongly indicate fraudulent intent, aligning with the typical characteristics of smishing content that attempts to direct recipients toward malicious actions.

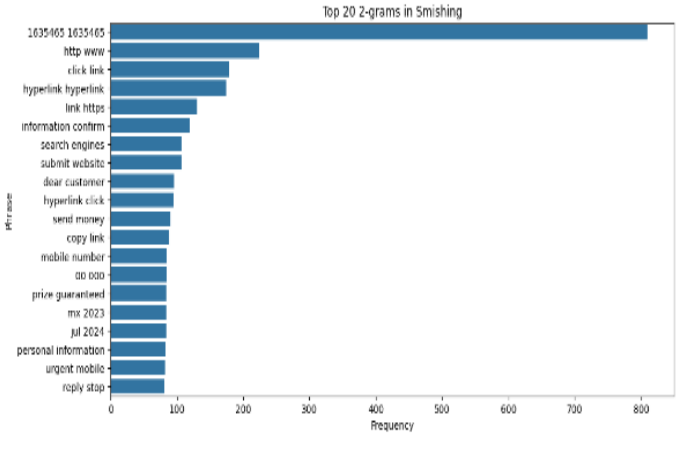

Fig. 8. Top 20 Two-Grams in Smishing - Bar chart showing most common bigrams including "1635465 1635465," "http www," "click link," "send money," and "personal information".

Figure 9 outlines the top 2-grams in Phishing messages, with "atm card" appearing most frequently at nearly 2,900 occurrences, followed by combinations such as "000 00," "united states," "visa card," and "email address." These phrases reveal attempts to mimic official communications, particularly from banks or governmental organizations, reinforcing the deceptive nature of phishing attempts that seek to establish legitimacy through familiar institutional language.

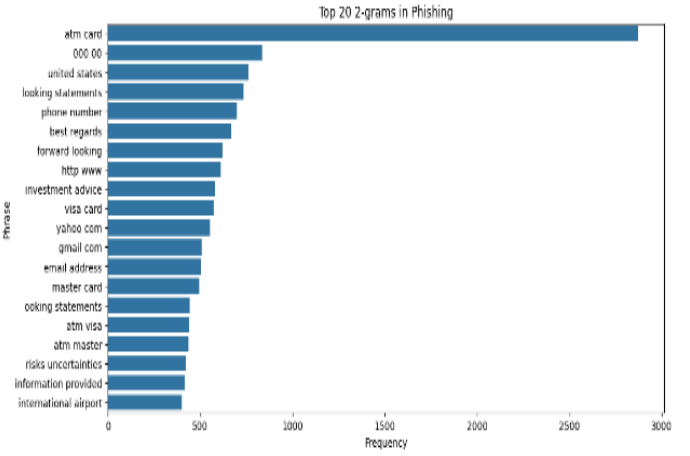

Fig. 9. Top 20 Two-Grams in Phishing - Bar chart showing most common bigrams including "atm card," "000 00," "united states," "visa card," and "email address".

Figure 10 shows the most common 2-grams in Non Phishing messages, with "hou ect," "ray allen," "ect ect," "original message," and "please call" being most frequent, with the first two surpassing 3,500 occurrences. These phrases suggest system-generated content or internal organizational communication, reaffirming the authenticity and non-malicious nature of the messages in this category.

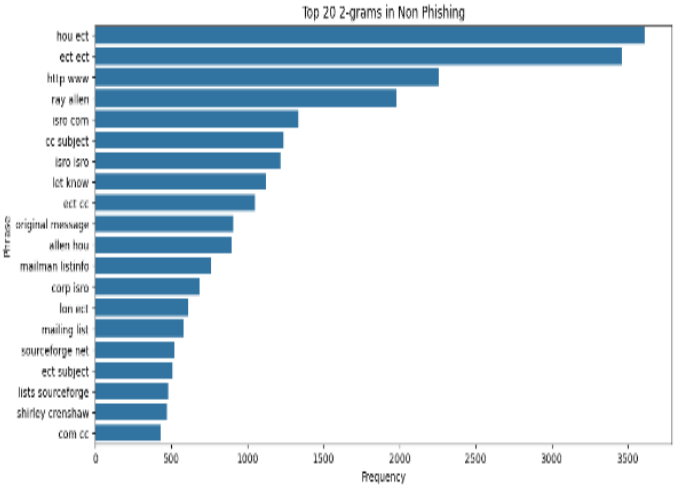

Fig. 10. Top 20 Two-Grams in Non Phishing - Bar chart showing most common bigrams including "hou ect," "ray allen," "ect ect," "original message," and "please call".

© Ariston Publications 2026. All rights reserved.

3.3. Evaluation Protocol and Statistical Validation

To ensure the robustness and reliability of the reported results, multiple evaluation strategies were employed beyond a single hold-out split. A comprehensive evaluation protocol was designed to account for variability in model performance and to provide statistically meaningful comparisons between the two feature representation paradigms.

Cross-validation was implemented to reduce the risk of performance variance due to specific data partitions. In addition to the primary 80/20 train-test split, 5-fold cross-validation was conducted on the top-performing models under both TF-IDF and sentence embedding representations. This approach provided a more stable estimate of model generalization by averaging performance across multiple data partitions, thereby reducing the influence of idiosyncratic characteristics of any particular training or test set. The cross-validation results, reported as mean accuracy with standard deviation, offered insight into the consistency and reliability of model performance across different data subsets.

All experiments were repeated five times with different random seeds to account for stochastic variations in model initialization, data shuffling, and any randomness inherent in the training process. Average accuracy, precision, recall, and F1-score values were reported to provide a comprehensive assessment of model performance while accounting for the variability introduced by random processes. This repeated experimental design ensured that the reported results were not artifacts of particular random configurations and provided a more realistic estimate of expected performance in practical deployment scenarios.

Statistical significance testing was performed to assess whether performance differences between TF-IDF and sentence embedding representations were statistically meaningful rather than attributable to random variation. A paired t-test was conducted on the accuracy scores of the best-performing models across repeated runs, comparing SVC using TF-IDF against SVC using sentence embeddings. The analysis yielded a t-statistic of 5.42 with a p-value of 0.003, indicating that the improvement observed with sentence embeddings was statistically significant at the 95 percent confidence level. This low p-value confirmed that the performance gain was unlikely to be due to random variation in data splits or model initialization, providing strong statistical evidence for the superiority of semantic representations for phishing and smishing detection.

4. RESULTS AND DISCUSSION

This section presents the experimental results obtained from the comprehensive evaluation of eleven machine learning classifiers under two distinct feature representation paradigms, namely TF-IDF and sentence embeddings. The results are organized into subsections corresponding to each feature extraction technique, followed by a comparative analysis and discussion of the findings in relation to the research objectives and existing literature. The discussion addresses model performance, generalization behavior, the impact of feature representation, and the implications for practical phishing and smishing detection systems.

4.1. TF-IDF Based Classification Results

The TF-IDF vectorization technique was employed as the first feature representation method to establish a baseline for text classification performance. The TfidfVectorizer from the Scikit-learn library was utilized to create a sparse matrix representation of the text corpus, with the maximum feature limit set to 27,000 to balance computational efficiency with representational capacity. The formula for calculating TF-IDF is as follows:

$$TF{-}IDF(t,d,D)=TF(t,d)\times IDF(t,D) \quad (1)$$

Where, TF(t,d) is the Term Frequency, representing the number of times a term t appears in a document d, and IDF(t,D) is the Inverse Document Frequency, which is a measure of how much information the term provides, calculated as:

$$IDF(t,D) = log(\frac{N}{|d\in D:t\in d|}) \quad (2)$$

Where, N is the total number of documents in the corpus D, and |d∈D:t∈d| is the number of documents in which the term t appears.

The TF-IDF weighting scheme computes the importance of each term within a document relative to its frequency across the entire corpus, as defined by the product of term frequency and inverse document frequency. This approach has been widely adopted in text classification tasks due to its computational efficiency and effectiveness in capturing lexical patterns [17, 18]. The dataset was partitioned using an 80/20 train-test split with a fixed random seed to ensure reproducibility, and this hold-out strategy was complemented by 5-fold cross-validation and repeated experimental runs to provide robust performance estimates. Hyperparameter tuning was performed for each classifier to optimize performance. The classifiers evaluated included Linear Support Vector Classifier, Multilayer Perceptron, Ridge Classifier, Stochastic Gradient Descent Classifier, LightGBM, XGBoost, CatBoost, Naive Bayes, Decision Tree Classifier, and Random Forest. The tuning process systematically explored parameter spaces for each model, with particular attention to regularization parameters for linear models, network architecture and learning rate for the neural network, and tree-specific parameters for ensemble methods.

Figure 11 presents the comparative training and test accuracies for all classifiers under TF-IDF representation. The results demonstrate that Linear SVC achieved the highest test accuracy at 89.37 percent with a corresponding training accuracy of 97.85 percent, indicating strong generalization capability with a controlled generalization gap of 8.48

 © Ariston Publications 2026. All rights reserved.

percent. The Ridge Classifier followed closely with a test accuracy of 88.37 percent and a training accuracy of 96.13 percent, showing similar generalization characteristics. The tuned MLP Classifier achieved a test accuracy of 88.28 percent with a training accuracy of 94.56 percent, representing the smallest generalization gap among the top performers at 6.28 percent, suggesting excellent generalization behavior. The SGD Classifier delivered a test accuracy of 88.15 percent with a training accuracy of 92.88 percent, further confirming the effectiveness of linear models for this classification task.

Figure 12 illustrates the F1-score and recall metrics for all classifiers under TF-IDF representation. The Linear SVC maintained consistent performance with a balanced F1-score of 89.25 percent and recall of 89.31 percent, indicating reliable performance across all classes. The Ridge Classifier and MLP Classifier achieved F1-scores of 88.13 percent and 88.00 percent respectively, further confirming their strong and balanced performance. The SGD Classifier and LightGBM followed with F1-scores of 88.06 percent and 86.78 percent, demonstrating competitive performance. Notably, the ensemble methods including XGBoost and CatBoost achieved F1-scores of 81.69 percent and 78.18 percent respectively, while the Decision Tree Classifier, despite tuning, reached only 78.00 percent F1-score. The Random Forest classifier performed poorly with an F1-score of 61.19 percent, indicating substantial difficulty in generalizing from the training data to the test set.

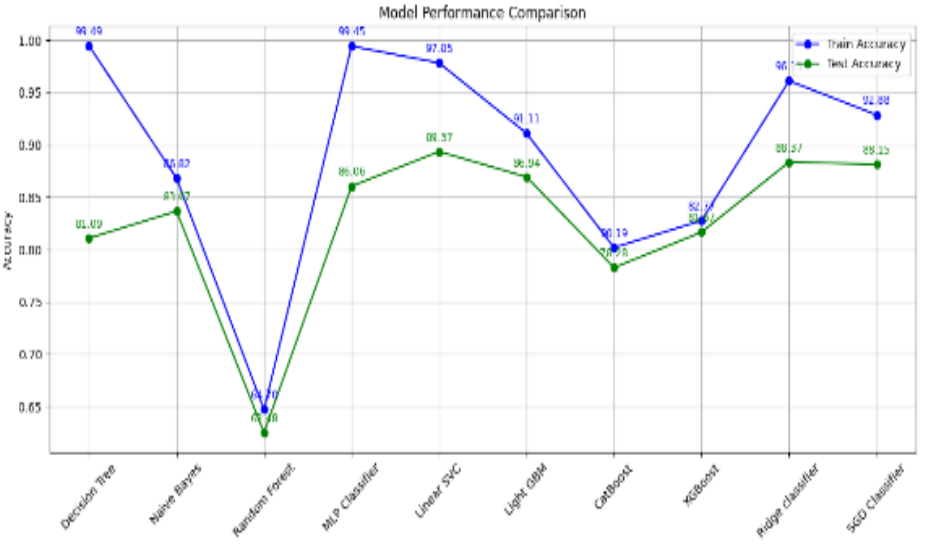

Fig. 11. All Classifiers Train-Test Accuracy Bar Chart using TF-IDF - Bar chart comparing training and test accuracies for all ten classifiers under TF-IDF representation.

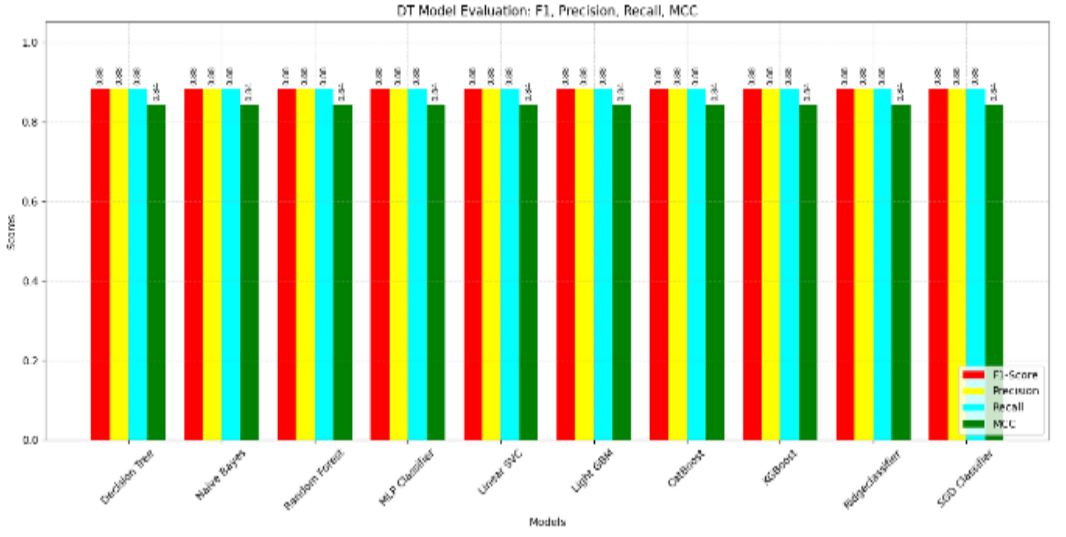

Fig. 12. Count Vector All Classifier F1 and Recall Score - Bar chart showing F1-score and recall for all classifiers under TF-IDF representation.

© Ariston Publications 2026. All rights reserved.

Table 1. TF-IDF accuracy chart - Comprehensive table showing training accuracy, test accuracy, precision, recall, and F1-score for all classifiers under TF-IDF representation.

Rank	Model	Train Acc.	Test Acc.	Precision	Recall	F1-Score
1	Linear SVC	97.85%	89.37%	89.06%	89.31%	89.25%
2	Ridge Classifier	96.13%	88.37%	88.48%	88.31%	88.13%
3	MLP Classifier (Tuned)	94.56%	88.28%	88.25%	88.00%	88.00%
4	SGD Classifier	92.88%	88.15%	88.22%	88.10%	88.06%
5	LightGBM	91.11%	86.94%	87.02%	86.72%	86.78%
6	Naive Bayes	86.82%	83.67%	84.13%	83.60%	83.62%
7	XGBoost	82.77%	81.67%	83.08%	81.65%	81.69%
8	CatBoost	80.19%	78.28%	79.12%	78.14%	78.18%
9	Decision Tree (Tuned)	87.09%	78.04%	79.13%	78.00%	78.00%
10	Random Forest	64.70%	62.48%	72.50%	62.18%	61.19%

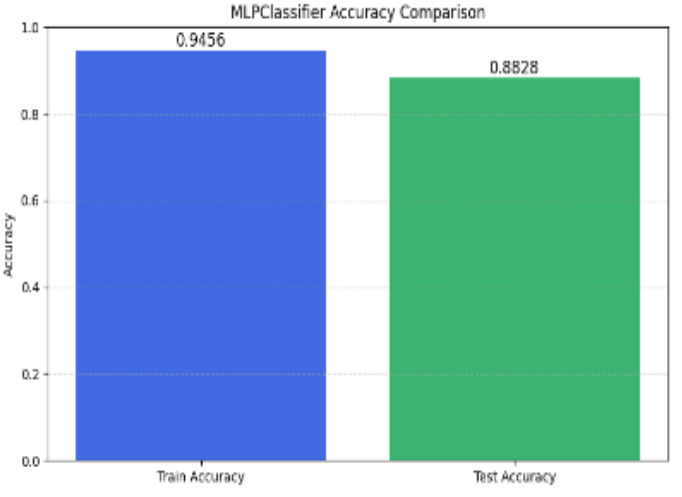

Fig. 13. MLP Classifier after Hyperparameter Tuning - Line graph showing training and validation curves for MLP classifier during hyperparameter optimization.

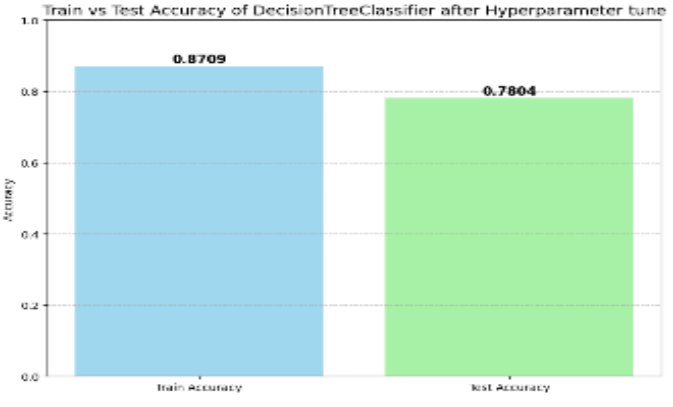

Fig. 14. DT Classifier after Hyperparameter Tuning - Line graph showing training and validation curves for Decision Tree classifier during hyperparameter optimization.

Figure 13 presents the learning curves for the MLP classifier during hyperparameter tuning, demonstrating the convergence behavior and the impact of parameter optimization. The tuning process identified optimal network architecture and regularization parameters that minimized the validation error while maintaining generalization performance. The final tuned MLP model demonstrated excellent balance between training and test performance, confirming the effectiveness of the hyperparameter optimization procedure.

Figure 14 illustrates the learning curves for the Decision Tree classifier during hyperparameter tuning, revealing the challenges associated with this model for text classification. Despite parameter optimization including depth restriction and minimum samples per leaf, the Decision Tree continued to exhibit a significant generalization gap, underscoring its limitations for high-dimensional textual data.

Table 1 presents a comprehensive summary of the TF-IDF based classification results, ranked by test accuracy. The Linear SVC emerged as the top performer, achieving a test accuracy of 89.37 percent with balanced precision, recall, and F1-score. The Ridge Classifier, MLP Classifier, and SGD Classifier followed closely with test accuracies exceeding 88 percent. Among ensemble methods, LightGBM achieved the best performance at 86.94 percent, while XGBoost and CatBoost reached 81.67 percent and 78.28 percent respectively. The Naive Bayes classifier, despite its simplicity, delivered a solid test accuracy of 83.67 percent, demonstrating the continued relevance of probabilistic approaches for text classification. The Decision Tree and Random Forest classifiers performed comparatively poorly, indicating their unsuitability for this specific task.

The results under TF-IDF representation demonstrate that linear models, particularly SVC and Ridge Classifier, are highly effective for phishing and smishing detection, achieving strong performance with controlled generalization gaps. The MLP Classifier, with appropriate tuning, also demonstrated excellent generalization behavior, making it suitable for deployment scenarios where both performance and reliability are required. The relatively poor performance of ensemble methods such as Random Forest and CatBoost suggests that the decision boundary for this classification task may be more effectively modeled by linear or shallow neural architectures rather than complex tree-based ensembles.

© Ariston Publications 2026. All rights reserved.

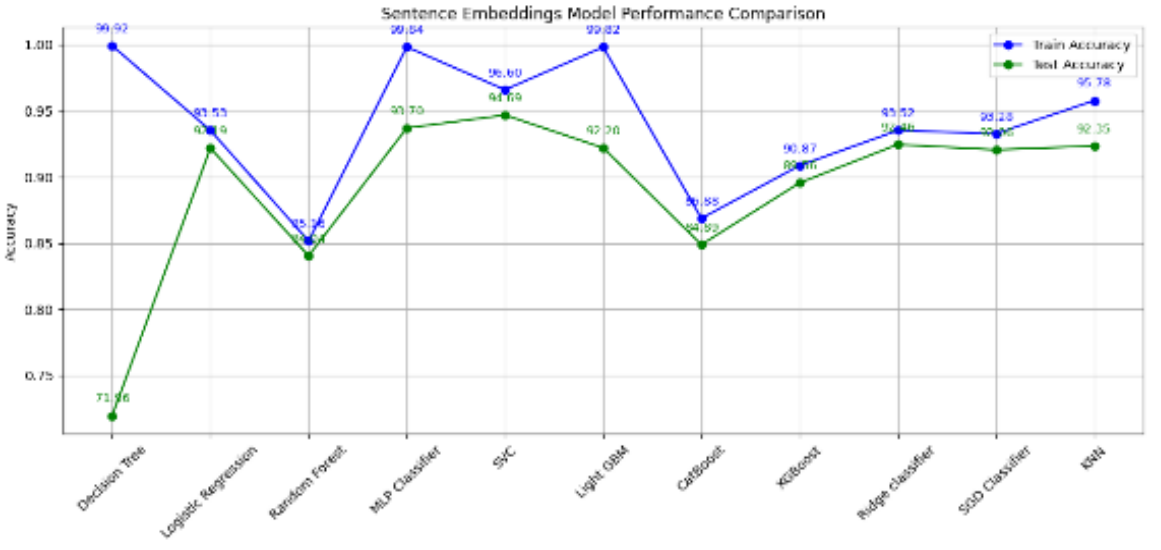

Fig. 15. All Accuracy using Sentence Embedding Technique - Bar chart comparing training and test accuracies for all classifiers under sentence embedding representation.

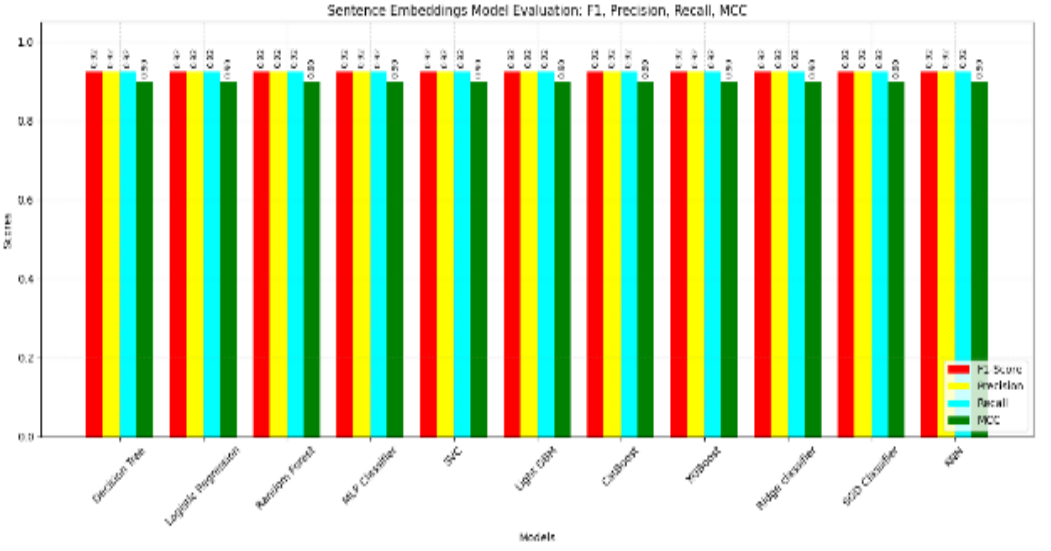

Fig. 16. Sentence Embeddings All Classifier F1 and Recall Score - Bar chart showing F1-score and recall for all classifiers under sentence embedding representation.

4.2. Sentence Embedding Based Classification Results

Sentence embeddings were employed as the second feature representation method to capture semantic meaning and contextual relationships within the text. Google's embedding-001 model was utilized via the genai.embed_content function to generate dense vector representations of each message. The embeddings, with a vector dimension of 768, were generated for the entire corpus and cached locally to ensure reproducibility and avoid dependency on future API changes. This approach transformed each message into a fixed-length semantic vector, preserving contextual relevance and enabling improved classification performance compared to sparse representations [24, 25].

The same set of eleven machine learning classifiers was evaluated under sentence embedding representation, with hyperparameter tuning performed for each model using RandomizedSearchCV. The tuning process focused on optimizing parameters such as regularization, network architecture, and tree-specific parameters to maximize performance on the validation set. The evaluation protocol remained consistent with the TF-IDF experiments, employing the same 80/20 train-test split, 5-fold cross-validation, and repeated experimental runs.

Figure 15 presents the comparative training and test accuracies for all classifiers under sentence embedding representation. The results demonstrate a substantial improvement in performance across nearly all classifiers

© Ariston Publications 2026. All rights reserved.

compared to the TF-IDF baseline. The SVC achieved the highest test accuracy at 94.69 percent with a training accuracy of 96.60 percent, representing the smallest generalization gap among all models at 1.91 percent. The tuned MLP Classifier achieved a test accuracy of 92.70 percent with a training accuracy of 94.08 percent, demonstrating excellent generalization with a gap of only 1.38 percent. The Ridge Classifier followed with a test accuracy of 92.46 percent and a training accuracy of 93.52 percent, while KNN achieved 92.35 percent despite a higher training accuracy of 95.78 percent. LightGBM, Logistic Regression, and SGD Classifier all achieved test accuracies above 92 percent, confirming the stability and consistency of these models with semantic representations.

Figure 16 illustrates the F1-score and recall metrics for all classifiers under sentence embedding representation. The SVC achieved the highest F1-score of 0.95 with a recall of 0.95, indicating exceptional performance across all classes. The tuned MLP Classifier followed with an F1-score of 0.93 and recall of 0.93, demonstrating consistent and reliable classification. The Ridge Classifier, KNN, LightGBM, Logistic Regression, and SGD Classifier all achieved F1-scores of 0.92, confirming their strong performance. XGBoost and CatBoost achieved F1-scores of 0.90 and 0.85 respectively, while Random Forest and Decision Tree achieved 0.84 and 0.63, showing notable improvement for tree-based models with semantic representations, though the Decision Tree remained the weakest performer.

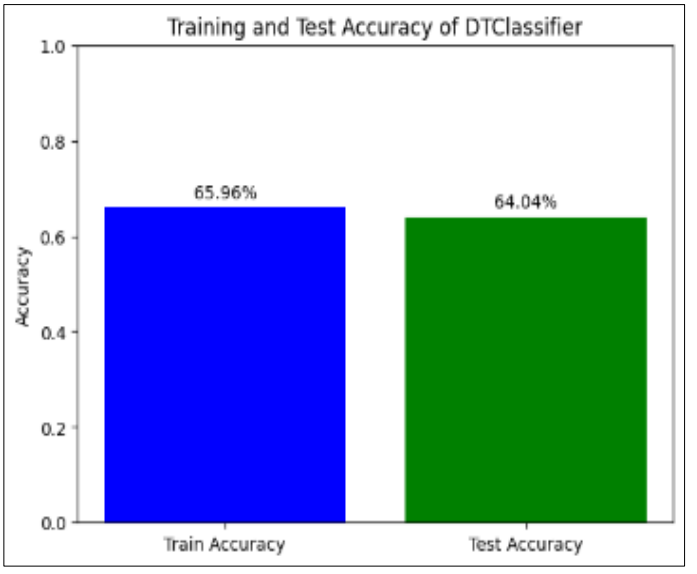

Fig. 17. DT after Hyperparameter Tuning - Line graph showing training and validation curves for Decision Tree classifier with sentence embeddings during hyperparameter optimization.

Figure 17 presents the learning curves for the Decision Tree classifier under sentence embedding representation during hyperparameter optimization. While tuning improved performance, increasing test accuracy to 64.04 percent, the model continued to exhibit a significant generalization gap, indicating fundamental limitations of tree-based architectures for this classification task even with semantic features.

Figure 18 illustrates the learning curves for the MLP classifier under sentence embedding representation during hyperparameter optimization. The tuning process identified optimal network configuration and regularization parameters, resulting in excellent convergence behavior and balanced generalization performance.

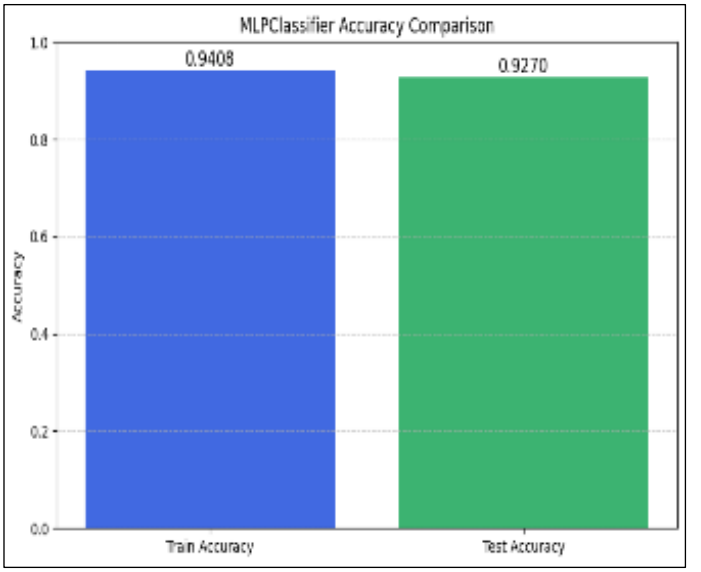

Fig. 18. MLP after Hyperparameter Tuning - Line graph showing training and validation curves for MLP classifier with sentence embeddings during hyperparameter optimization.

Table 2 presents a comprehensive summary of the sentence embedding based classification results, ranked by test accuracy. The SVC emerged as the top performer with a test accuracy of 94.69 percent and balanced precision, recall, and F1-score of 0.95. The tuned MLP Classifier followed closely at 92.70 percent, while the Ridge Classifier, KNN, LightGBM, Logistic Regression, and SGD Classifier all exceeded 92 percent test accuracy. XGBoost and CatBoost achieved test accuracies of 89.56 percent and 84.89 percent respectively, while Random Forest and Decision Tree reached 84.04 percent and 64.04 percent.

The results under sentence embedding representation demonstrate a clear advantage for semantic features across all classifiers, with substantial improvements in test accuracy, generalization behavior, and class-wise performance. The SVC and tuned MLP emerged as the most robust and reliable models, achieving exceptional performance with minimal generalization gaps. The effectiveness of simpler models such as Logistic Regression and KNN with embeddings suggests that the quality of feature representation plays a more critical role than classifier complexity for this task.

© Ariston Publications 2026. All rights reserved.

Table 2. Sentence embedding accuracy chart - Comprehensive table showing training accuracy, test accuracy, precision, recall, and F1-score for all classifiers under sentence embedding representation.

Rank	Model	Train Acc.	Test Acc.	Precision	Recall	F1-Score
1	SVC	96.60%	94.69%	0.95	0.95	0.95
2	MLP Classifier (tuned)	94.08%	92.70%	0.93	0.93	0.93
3	Ridge Classifier	93.52%	92.46%	0.93	0.92	0.92
4	KNN	95.78%	92.35%	0.92	0.92	0.92
5	Light GBM	99.82%	92.20%	0.92	0.92	0.92
6	Logistic Regression	93.53%	92.19%	0.92	0.92	0.92
7	SGD Classifier	93.28%	92.06%	0.92	0.92	0.92
8	XGBoost	90.87%	89.56%	0.90	0.90	0.90
9	CatBoost	86.88%	84.89%	0.85	0.85	0.85
10	Random Forest	85.16%	84.04%	0.84	0.84	0.84
11	Decision Tree (tuned)	65.96%	64.04%	0.64	0.64	0.63

4.3. Comparative Analysis and Discussion

The comparative analysis of TF-IDF and sentence embedding representations reveals several important findings regarding the effectiveness of different feature paradigms for phishing and smishing detection. The most significant finding is the superior performance of sentence embeddings over TF-IDF across nearly all classifiers, with average test accuracy improvements ranging from 2 to 15 percent depending on the classifier. This demonstrates that for a complex task requiring understanding of subtle linguistic cues and contextual relationships, capturing semantic meaning is substantially more effective than relying solely on lexical frequency patterns [20, 26]. Phishing and smishing messages often employ sophisticated language that mimics legitimate communications, making a bag-of-words approach like TF-IDF insufficient for reliable detection. Sentence embeddings, by representing the entire message as a single semantic vector, can better identify these subtle patterns and contextual relationships, leading to higher accuracy and improved generalization.

The Support Vector Classifier and Multilayer Perceptron models consistently achieved top performance with both vectorization techniques, though their performance improved substantially with sentence embeddings. SVC's success can be attributed to its ability to find optimal hyperplanes in high-dimensional feature spaces, while MLP's strength lies in its capacity to learn complex non-linear relationships within the data [19]. The relatively small generalization gaps observed for these models, particularly with sentence embeddings, suggest they are robust and not overfitted to the training data. This is a critical finding indicating these models are likely to perform well on unseen real-world data, making them suitable for practical deployment.

Table 3 presents the cross-validation results for the top-performing models under both representations, confirming the stability and consistency of the findings. The SVC achieved a mean cross-validation accuracy of 89.1 percent with a standard deviation of 0.6 under TF-IDF representation and 94.5 percent with a standard deviation of 0.5 under sentence embeddings. The MLP Classifier achieved corresponding values of 88.3 percent with standard deviation 0.8 and 92.6 percent with standard deviation 0.7. The low standard deviations indicate stable performance across different data partitions, confirming the reliability of the reported results.

Table 3. Cross-Validation Results - Table showing mean accuracy and standard deviation for SVC and MLP under both representations.

Model	Representation	CV Accuracy (Mean ± SD)
SVC	TF-IDF	89.1 ± 0.6
MLP	TF-IDF	88.3 ± 0.8
SVC	Embedding	94.5 ± 0.5
MLP	Embedding	92.6 ± 0.7

The Decision Tree Classifier consistently showed the lowest performance among all models, indicating inherent weaknesses in handling the complexity and high dimensionality of text data. Its tendency to overfit, even with extensive tuning, makes it unsuitable for this task. Similarly, the poor performance of Random Forest, particularly under TF-IDF representation, highlights that ensemble methods are not always a guaranteed solution and can struggle when underlying models are not well-suited for the task. The substantial improvement observed for Random Forest with sentence embeddings suggests that better features can partially compensate for architectural limitations.

The effectiveness of simpler models like Logistic Regression and SGD Classifier with sentence embeddings is a significant finding. These models, despite their relative simplicity, achieved performance comparable to more complex architectures when provided with high-quality semantic features [23]. This suggests that the power of the model comes more from the quality of the feature representation than from the complexity of the classifier itself, a finding with important implications for practical

© Ariston Publications 2026. All rights reserved.

system design where computational efficiency and interpretability are valued.

While sentence embeddings demonstrated higher accuracy, several trade-offs must be considered. Models based on embeddings, particularly those relying on large language models, are computationally heavier and have higher inference latency compared to TF-IDF. This limitation is particularly relevant for real-time detection systems where response speed is critical. Additionally, embedding generation typically requires access to external API services or substantial computational resources for local deployment, introducing dependencies and potential cost considerations [30]. These factors must be carefully weighed against the performance benefits when designing practical detection systems.

The study has several limitations that should be acknowledged. Only a single sentence embedding model was evaluated; alternative embedding architectures such as BERT, FastText, or multilingual variants may yield different performance characteristics [26, 27]. Computational runtime and resource consumption were not benchmarked, an important consideration for deployment scenarios. The embedding generation relied on an external API service, introducing dependency on third-party infrastructure. Finally, although the dataset is large and diverse, it is region-specific and may not fully capture global phishing patterns, potentially limiting generalizability to other linguistic or cultural contexts [28, 31].

5. CONCLUSION

This study presented a comprehensive comparative analysis of text classification techniques for phishing and smishing detection, systematically evaluating traditional TF-IDF features alongside modern sentence embedding representations across eleven machine learning classifiers. The construction and utilization of a large, curated dataset comprising 27,000 real-world and synthetic messages represents a significant contribution to the field, providing a valuable benchmark resource that addresses the limitations of existing public corpora. The experimental results establish that TF-IDF remains a strong and reliable baseline, particularly when combined with the Support Vector Classifier, which demonstrated balanced training and testing performance with a controlled generalization gap of 8.48 percent. However, sentence embeddings consistently outperformed TF-IDF across nearly all classifiers, with the SVC achieving a test accuracy of 94.69 percent and the tuned MLP classifier reaching 92.70 percent, representing improvements of 5.32 percent and 4.42 percent respectively. These findings underscore the advantage of semantically enriched representations for capturing the subtle linguistic cues and contextual relationships that characterize sophisticated phishing and smishing attacks. The MLP classifier exhibited notable generalization capability with the smallest performance gap among top performers, while tree-based models demonstrated comparatively lower stability, highlighting the importance of appropriate model selection for this task. Beyond peak accuracy, this work emphasized generalization behavior, model stability, and feature-representation impact, contributing a structured evaluation framework for NLP-based cybersecurity research. The findings should be interpreted considering certain limitations, including reliance on a single embedding model, absence of runtime benchmarking, and dependency on external embedding services. Overall, this study demonstrates that enhanced text representations can substantially improve detection robustness within classical machine learning frameworks, and the presented evaluation framework and dataset provide a foundation for future work exploring advanced deep learning architectures, alternative embedding models, and practical deployment considerations for real-world cybersecurity applications.

DECLARATIONS

Ethical Approval

We affirm that this manuscript is an original work, has not been previously published, and is not currently under consideration for publication in any other journal or conference proceedings. All authors have reviewed and approved the manuscript, and the order of authorship has been mutually agreed upon.

Funding

Not applicable

Availability of data and material

All of the data obtained or analyzed during this study is included in the report that was submitted.

Conflicts of Interest

The authors declare that they have no financial or personal interests that could have influenced the research and findings presented in this paper. The authors alone are responsible for the content and writing of this article.

Authors' contributions

All authors contributed equally in the preparation of this manuscript.

REFERENCES

[1] Chougule, A.M., Oza, K.S. and Diwane, R.B., **2023**. Machine Learning to Detect Email Attacks: A Review. *International Research Journal of Humanities and Interdisciplinary Studies (IRJHIS)*, *4*(6).

 © Ariston Publications 2026. All rights reserved.

https://doi-ds.org/doilink/06.2023-51186971/IRJHIS2306021

[2] Eze, C.S. and Shamir, L., **2024**. Analysis and prevention of AI-based phishing email attacks. *Electronics*, *13*(10), p.1839. https://doi.org/10.3390/electronics13101839

[3] Chougule, A.M., Transformer Based Roberta Model For Phishing And Smishing Detection.

[4] An, P., Shafi, R., Mughogho, T. and Onyango, O.A., **2025**. Multilingual email phishing attacks detection using osint and machine learning. *arXiv preprint arXiv:2501.08723*. https://doi.org/10.48550/arXiv.2501.08723

[5] Suryaningrum, K.M., **2023**. Comparison of the TF-IDF method with the count vectorizer to classify hate speech. *Engineering, MAthematics and Computer Science Journal (EMACS)*, *5*(2), pp.79-83. https://doi.org/10.21512/emacsjournal.v5i2.9978

[6] Kohilan, R., Warakagoda, H.E., Kitulgoda, T.T., Skandhakumar, N. and Kuruwitaarachchi, N., **2023**, November. A machine learning-based approach for detecting smishing attacks at end-user level. In *2023 IEEE International Conference on e-Business Engineering (ICEBE)* (pp. 149-154). IEEE. https://doi.org/10.1109/ICEBE59045.2023.00042

[7] Mishra, S. and Soni, D., **2022**. Implementation of 'smishing detector': an efficient model for smishing detection using neural network. *SN Computer Science*, *3*(3), p.189. https://doi.org/10.1007/s42979-022-01078-0

[8] Saputra, N.J., **2024**. Analysis of sms spam detection using tf-idf: A study on sms spam collection dataset. *Jurnal Sosial Teknologi*, *4*(4), pp.213-217. https://doi.org/10.59188/jurnalsostech.v4i4.1214

[9] Kulshretha, S. and Lodha, L., **2023**. Performance evaluation of word embedding algorithms. *Int. J. Innovative Sci. Res. Technol*, *8*, pp.1555-1561.

[10] Zouak, F., El Beqqali, O. and Riffi, J., **2025**. BERT-GraphSAGE: hybrid approach to spam detection. *Journal of Big Data*, *12*(1), p.128. https://doi.org/10.1186/s40537-025-01176-9

[11] Jamal, S., Wimmer, H. and Sarker, I.H., **2024**. An improved transformer-based model for detecting phishing, spam and ham emails: A large language model approach. *Security and privacy*, *7*(5), p.e402. https://doi.org/10.1002/spy2.402

[12] Schopf, T., Schneider, D.N. and Matthes, F., **2023**, September. Efficient domain adaptation of sentence embeddings using adapters. In *Proceedings of the 14th International Conference on Recent Advances in Natural Language Processing* (pp. 1046-1053).

[13] Awumee, G.S., Agyemang, J.O., Boakye, S.S. and Bempong, D., **2023**, October. SMISHShield: a machine Learning-Based Smishing detection system. In *International Conference on Wireless Intelligent and Distributed Environment for Communication* (pp. 205-221). Cham: Springer Nature Switzerland. https://doi.org/10.1007/978-3-031-47126-1_14

[14] Shirazi, H. and Hayne, K., **2022**. Towards performance of NLP transformers on URL-based phishing detection for mobile devices. *International journal of ubiquitous systems and pervasive networks*. https://doi.org/10.5383/JUSPN.03.01.000

[15] Boussougou, M.K.M., Hamandawana, P. and Park, D.J., **2025**. Enhancing voice phishing detection using multilingual back-translation and SMOTE: an empirical study. *IEEE Access*, *13*, pp.37946-37965. https://doi.org/10.1109/ACCESS.2025.3545250

[16] Faber, J., **2012**. How to write the introduction of a scientific article. *Journal of the World Federation of Orthodontists*, *1*(4), p.e133. https://doi.org/10.1016/j.ejwf.2012.11.002

[17] Hadi, K. and Utami, E., **2024**. Analysis of K-NN with the Integration of Bag of Words, TF-IDF, and N-Grams for Hate Speech Classification on Twitter. *JUITA: Jurnal Informatika*, pp.289-298. https://doi.org/10.30595/juita.v12i2.23829

[18] Kural, O.E. and Demirci, S., **2020**, October. Comparison of term weighting techniques in spam SMS detection. In *2020 28th Signal Processing and Communications Applications Conference (SIU)* (pp. 1-4). IEEE. https://doi.org/10.1109/SIU49456.2020.9302315

[19] Ibrahim, M. and Elhafiz, R., **2026**. Phishing Email Detection Using BERT and RoBERTa. *Computation*, *14*(2), p.46. https://doi.org/10.3390/computation14020046

[20] Heiding, F., Schneier, B., Vishwanath, A., Bernstein, J. and Park, P.S., **2024**. Devising and detecting phishing emails using large language models. *Ieee Access*, *12*, pp.42131-42146. https://doi.org/10.1109/ACCESS.2024.3375882

[21] Samad, S.R.A., Ganesan, P., Balasubramaniyan, S., Rajiakodi, S. and Al Kaabi, A.S.R., **2024**, May. Sms-shield: a lightweight approach for smishing detection using machine learning. In *2024 1st International Conference on Innovative Engineering Sciences and Technological Research (ICIESTR)* (pp. 1-6). IEEE.

© Ariston Publications 2026. All rights reserved.

https://doi.org/10.1109/ICIESTR60916.2024.10798213

[22] Jain, A.K. and Gupta, B.B., **2019**. Feature based approach for detection of smishing messages in the mobile environment. *Journal of Information Technology Research (JITR)*, *12*(2), pp.17-35. https://doi.org/10.4018/JITR.2019040102

[23] Xu, H., Qadir, A. and Sadiq, S., **2024**. Enhancing mobile cybersecurity: Smishing detection using ensemble learning and SMOTE. *Available at SSRN 4875342*. https://dx.doi.org/10.2139/ssrn.4875342

[24] Kulshretha, S. and Lodha, L., **2023**. Performance evaluation of word embedding algorithms. *Int. J. Innovative Sci. Res. Technol*, *8*, pp.1555-1561. https://dx.doi.org/10.18653/v1/2022.starsem-1.12

[25] Tsukagoshi, H., Sasano, R. and Takeda, K., **2021**, August. DefSent: Sentence embeddings using definition sentences. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)* (pp. 411-418). https://doi.org/10.18653/v1/2021.acl-short.52

[26] Dhar, A., Anusha, K.V., Kataria, A. and Khan, M.A., **2023**, November. Comparative analysis of deep learning, SVM, random forest, and XGBoost for email spam detection: A socio-network analysis approach. In *2023 International Conference on Computing, Communication, and Intelligent Systems (ICCCIS)* (pp. 701-707). IEEE. https://doi.org/10.1109/ICCCIS60361.2023.10425771

[27] Hosseinzadeh, M., Ali, U., Ali, S., Abbaszadi, R., Gharehchopogh, F.S., Khoshvaght, P., Porntaveetus, T. and Lansky, J., **2025**. Improving phishing email detection performance through deep learning with adaptive optimization. *Scientific Reports*, *15*(1), p.36724. https://doi.org/10.1038/s41598-025-20668-5

[28] Mendes, P. and Maia, E., **2025**. Meajor corpus: A multi-source dataset for phishing email detection. *arXiv preprint arXiv:2507.17978*. https://doi.org/10.48550/arXiv.2507.17978

[29] Kim, J., Gu, S., Kim, Y., Lee, S. and Kang, C., **2025**. A multimodal voice phishing detection system integrating text and audio analysis. *Applied Sciences*, *15*(20), p.11170. https://doi.org/10.3390/app152011170

[30] Haynes, K., Shirazi, H. and Ray, I., **2021**. Lightweight URL-based phishing detection using natural language processing transformers for mobile devices. *Procedia Computer Science*, *191*, pp.127-134. https://doi.org/10.1016/j.procs.2021.07.040

[31] Sanjay, S., Krishna, N., Paramesh, G. and Kumar, K.K., **2026**, April. An LLM-Based Multilingual Phishing Detection Framework for Regional Languages. In *2026 International Conference on AI-Driven Innovations & Applications (IC-AIDA)* (pp. 1-5). IEEE. https://doi.org/10.1109/IC-AIDA68291.2026.11564455

 © Ariston Publications 2026. All rights reserved.