JURNAL TEKNOLOGI INFORMASI
Jurnal Keilmuan dan Aplikasi Bidang Teknik Informatika

[E-ISSN 2656-0321]
[Vol 19 No 2]
[Agustus 2025]

# Spam Email Classification Using Support Vector Machine (SVM) and TF-IDF: A Case Study with the TREC 2007 and Enron-Spam Datasets

I Gusti Ngurah Darma Paramartha[1*], I Made Ardi Sudestra[2], Adie Wahyudi Oktavia Gama[3], Gede Humaswara Prathama[4]

[acd] Department of Information Technology, Universitas Pendidikan Nasional, Bali, Denpasar, Indonesia
[b] Department of Computer Science, Universitas Pendidikan Ganesha, Bali, Singaraja, Indonesia
ngurahdarma@undiknas.ac.id[1], ardi.sudestra@student.undiksha.ac.id[2], adiewahyudi@undiknas.ac.id[3], huma@undiknas.ac.id[4]
* I Gusti Ngurah Darma Paramartha

ARTICLE INFO

**Keywords**

Spam Email Classification, Support Vector Machine (SVM), TF-IDF Vectorizer, Machine Learning, Text Classification

ABSTRACT (10PT)

Spam emails represent a substantial concern within the digital landscape, impeding users with unsolicited communications. This study elucidates the utilization of a Support Vector Machine (SVM) coupled with a TF-IDF Vectorizer for categorizing emails into spam and non-spam classifications. The model was developed utilizing two publicly accessible pre-processed datasets: the TREC 2007 Public Spam Corpus and the Enron-Spam Dataset. By employing the TF-IDF algorithm, which allocates heightened importance to infrequent yet pertinent terms, alongside SVM, renowned for its efficacy in textual classification, the model exhibits remarkable efficacy, achieving an accuracy of 99.04%, a precision of 98.57% and a recall of 99.62%. These findings underscore the model's formidable capacity to discern spam emails while concurrently minimizing false positives accurately. This is critical for real-world applications where authentic emails must not be erroneously categorized as spam. Furthermore, this study elaborates on the justification for the selection of TF-IDF and SVM in the context of spam email classification, in addition to the evaluation outcomes of the model, which align with existing literature, wherein the integration of SVM with TF-IDF has demonstrated substantial performance in spam detection endeavours.

## 1. Introduction

In the era of digital communication, email has become an indispensable tool for personal and professional interaction. However, alongside its benefits, email also presents significant challenges, one of the most pervasive being the problem of spam. Spam emails are unsolicited messages sent in bulk, often for advertising or malicious purposes. These emails can disrupt users, overwhelm inboxes, and lead to productivity losses. More critically, spam emails can pose serious security risks, including phishing attacks, malware distribution, and other forms of cybercrime. Consequently, developing effective spam filtering systems has become essential to ensure secure and efficient communication [1].

Classifying emails as spam or non-spam is a typical problem in text classification, where the goal is to categorize unstructured text data into predefined categories [2]. Spam detection, in particular, is a binary classification problem where emails must be classified into two categories: spam (undesirable) or ham (legitimate) [2]. Traditional rule-based filtering systems often need to catch up, as they cannot adapt to the dynamic nature of spam emails, which continuously evolve in content and tactics [3]. To address these limitations, machine learning (ML) techniques have emerged as a more effective solution, capable of learning complex patterns from data and providing high levels of accuracy in classification tasks [4], [5].

Among the various machine learning algorithms, Support Vector Machine (SVM) has gained significant attention for its ability to perform well in high-dimensional spaces, such as those encountered in text classification [6]. SVM operates by finding a hyperplane that optimally separates the two classes (spam and non-spam) in the feature space, maximizing the margin between the classes [7]. Its effectiveness in handling non-linear decision boundaries and high-dimensional data makes it an ideal choice for spam email classification. SVM has been successfully applied to a wide range of text classification problems, including sentiment analysis, topic classification, and, as in this study, spam detection [8].

The TF-IDF (Term Frequency-Inverse Document Frequency) technique is widely used for transforming text data into numerical representations that machine learning algorithms can understand [9], [10]. Unlike simpler methods such as Bag of Words (BoW), which only count the frequency of terms, TF-IDF provides a more nuanced representation by considering both the frequency of terms within a specific document and the inverse frequency of terms across the entire corpus [10]. This weighting mechanism ensures that terms standard across many documents are given less importance, while terms that are unique and more discriminative are weighted more heavily. As such, TF-IDF has proven to be highly effective in capturing the distinctive features of spam emails, making it an excellent choice for feature extraction in spam classification tasks [11].

This paper builds upon the TREC 2007 Public Spam Corpus and the Enron-Spam Dataset to develop and evaluate a spam classification model using a combination of TF-IDF vectorization and SVM. Both datasets contain labelled examples of spam and non-spam emails, allowing for the training and testing of classification models. By leveraging the strengths of both TF-IDF and SVM, this study aims to provide a robust framework for accurate and efficient email spam detection [12]. The following sections detail the methodology, results, and discussion of the proposed model's performance, demonstrating this approach's potential in real-world email filtering applications.

## 2. Research Method

The process of constructing a spam email classification model using the Support Vector Machine (SVM) algorithm combined with the TF-IDF Vectorizer follows a rigorous, systematic approach [12]. This methodology is divided into several key stages, each playing a crucial role in the development of an effective and accurate model. These stages encompass the acquisition and preprocessing of datasets, the extraction of meaningful features from the textual data, the training of the classification model, and the subsequent evaluation of its performance. Each stage is designed to ensure that the model learns from the data in a way that maximizes its ability to distinguish between spam and non-spam emails [13].

### Datasets Used

The model was built using two well-known and publicly available datasets: the TREC 2007 Public Spam Corpus and the Enron-Spam Dataset. Both datasets contain labeled email data, where each email is marked as either spam (label = 1) or non-spam (label = 0). These labeled datasets are crucial for supervised learning tasks, as they provide the necessary ground truth for training and testing the classification model.

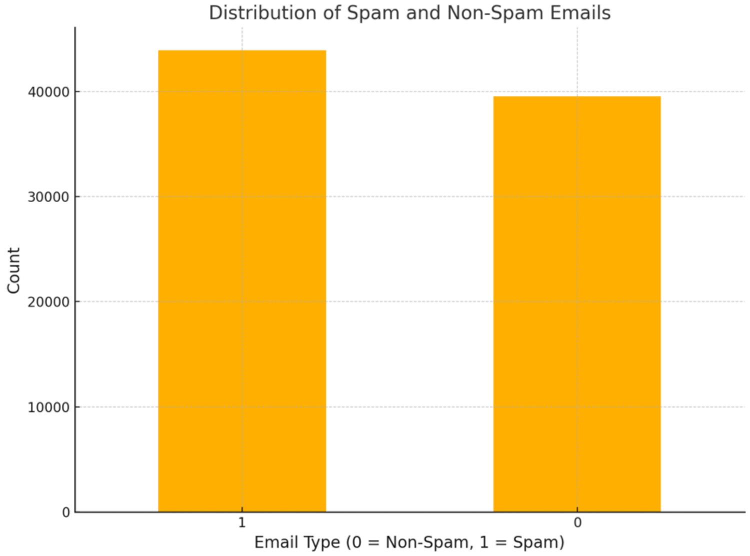

Figure 1. Distribution of Spam and Non-Spam Emails in the Dataset.

The TREC 2007 Public Spam Corpus is a collection of emails categorized as either spam or non-spam (ham). This dataset was originally created for the Text REtrieval Conference (TREC), which provides benchmarks for text classification tasks. It includes emails that represent real-world spam filtering challenges, making it suitable for model training and evaluation. The dataset is available for download in its original form from the TREC 2007 website and can also be accessed in a preprocessed version from Kaggle.

The Enron-Spam Dataset contains a collection of emails from the Enron corporation. Curated for the purpose of spam email classification, it includes both legitimate corporate communication and unsolicited spam messages. The diversity of the Enron dataset makes it valuable for evaluating spam detection algorithms, as it captures a wide range of email content.

Both datasets have been pre-processed to remove unnecessary features, such as headers, footers, and other non-textual elements that could interfere with model training. They are also cleaned to standardize the text and remove stop words, which are common in spam emails but do not contribute to distinguishing between spam and non-spam.

### Preprocessing of Data

Data preprocessing plays a crucial role in ensuring that the input data is in the right format for machine learning models. For email data classification, the preprocessing pipeline typically involves several key steps: text cleaning, tokenization, and feature extraction [14].

Text cleaning is the first step in the preprocessing process, where irrelevant or noisy data is removed from the email content. This may involve stripping out HTML tags, special characters, email signatures, and other non-textual elements that do not contribute to the classification task. By eliminating these unnecessary artifacts, the raw email content is transformed into a cleaner, more focused version that is ready for further processing.

Following text cleaning, the next step is tokenization, which involves splitting the cleaned text into individual words or terms. This process breaks down the raw text into smaller, manageable units that machine learning algorithms can work with. Tokenization converts the text into a structured format, making it possible to identify and analyze the frequency and distribution of words in the dataset.

Once the text has been cleaned and tokenized, the next crucial step is feature extraction. In this case, TF-IDF (Term Frequency-Inverse Document Frequency) vectorization is commonly used to convert the text into numerical features that can be fed into a machine learning model. TF-IDF is a statistical measure that helps to identify the importance of each word in a document relative to the entire corpus. By calculating the frequency of a term within a specific document and comparing it to how often it appears across the entire dataset, the TF-IDF method assigns higher weights to terms that are frequent in a given document but rare across all documents [2]. This ensures that distinctive and potentially more informative terms are emphasized, helping the model to better distinguish between spam and non-spam emails.

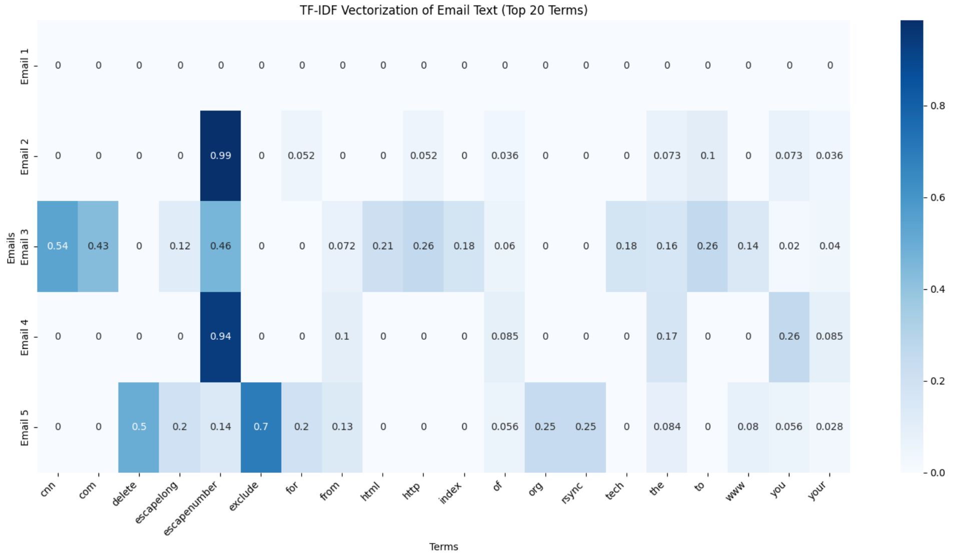

Figure 1. TF-IDF vectorization process, transforming email text into numerical feature vectors.

The TF-IDF vectorization process applied to five sample emails, showcasing the top 20 terms with the highest weights. Each row represents an individual email, while each column corresponds to a specific term selected by the TfidfVectorizer. The numerical values within the cells represent the TF-IDF scores, which indicate the importance of each word in a particular email relative to the entire dataset. Darker shades of blue signify higher TF-IDF values, emphasizing terms that are more significant in distinguishing the content of specific emails. For example, words like "delete," "exclude," and "org" show notably higher scores in certain emails, suggesting their importance in identifying unique features of those texts. This visualization provides a clear understanding of the distribution of term weights and highlights how significant words can aid a model in classifying email content.

## Model Training

For the classification task, the Support Vector Machine (SVM) algorithm is selected due to its effectiveness in handling high-dimensional data [15]. SVM is a supervised learning model that works by identifying the optimal hyperplane that separates data points from different classes in a high-dimensional space [16]. In the case of spam email classification, the objective is to find the hyperplane that best distinguishes between spam and non-spam emails [11].

The SVM model is trained using feature vectors obtained from the TF-IDF transformation. During training, the model learns to establish a decision boundary that separates the two classes. Once the model is trained, it is capable of predicting the class of new, unseen emails, based on the patterns it has learned from the training data.

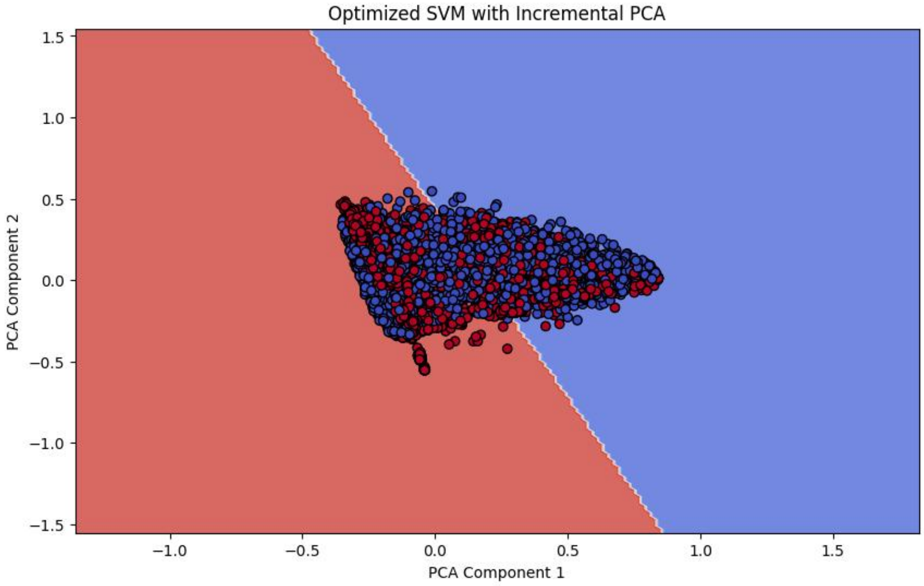

Figure 2. SVM Classification with Hyperplane Separation Using Incremental PCA.

This image illustrates the application of a Support Vector Machine (SVM) model trained to classify text data through dimensionality reduction using Principal Component Analysis (PCA). SVM is used to find the hyperplane that separates two classes of data, in this case, emails categorized as spam and non-spam. PCA is employed to reduce the high-dimensional data, making it possible to visualize the classification results in a two-dimensional space. In the image, two areas, colored red and blue, represent the two classes predicted by the model, with the hyperplane in the middle separating the two classes. The points in the image represent data that has been processed and mapped to the two principal components through PCA. Red points represent data labeled as spam, while blue points represent non-spam data. The presence of the hyperplane clearly shows the separation between the two classes, demonstrating the effectiveness of the SVM model in classifying the data. The use of PCA here allows us to visualize data, which initially had many dimensions, in two dimensions, making it easier to understand while also showing how SVM can optimally separate the two classes.

It should be noted that this image only shows a sample of 10,000 data points randomly selected from the larger dataset. This limitation was applied to ensure that the visualization process remains efficient and does not overwhelm the memory, given that the original dataset contains over 80,000 data points. By using a smaller data sample, this image still effectively demonstrates the SVM's ability to separate the data, even though only a portion of the total available data is used [17].

### Model Evaluation

Once the model is trained, its performance is evaluated using several standard classification metrics such as accuracy, precision, and recall [18]. These metrics are computed by comparing the model’s predictions with the actual labels in the test set. The accuracy of the model is 0.9904, meaning that approximately 99.04% of the predictions are correct, indicating strong overall performance in distinguishing between spam and non-spam emails. Precision, which is 0.9857, shows that when the model predicts an email as spam, it is correct 98.57% of the time. This is particularly important when minimizing false positives, as precision reflects the reliability of the model in making spam predictions.

Recall, with a value of 0.9962, reveals that the model successfully identifies 99.62% of all actual spam emails. This metric is crucial in scenarios where it's important to capture as many instances of the target class (spam) as possible, even at the cost of some false positives. Together, these metrics provide a comprehensive view of the model's effectiveness, showcasing its ability to balance high accuracy, precision, and recall. The results suggest that the model is both highly accurate and effective at identifying spam emails without significant trade-offs in performance.

Table 1. Performance Metrics of Model

| Metric | Value |
|---|---|
| Accuracy | 0.9904 |
| Precision | 0.9857 |
| Recall | 0.9962 |

**System Overview**

The system under investigation consists of several key components: data preprocessing, feature extraction, model training, and model evaluation. The process of email data classification is conducted in the following stages: (1) loading and cleaning the dataset, (2) transforming textual data into numerical features using the Term Frequency-Inverse Document Frequency (TF-IDF) method, (3) training a Support Vector Machine (SVM) classifier, (4) evaluating the model's performance on test data, and (5) applying the trained model to classify new, unseen emails. This comprehensive, end-to-end pipeline not only ensures the model's efficacy in classifying spam emails but also demonstrates scalability, making it well-suited for deployment in real-world applications [19].

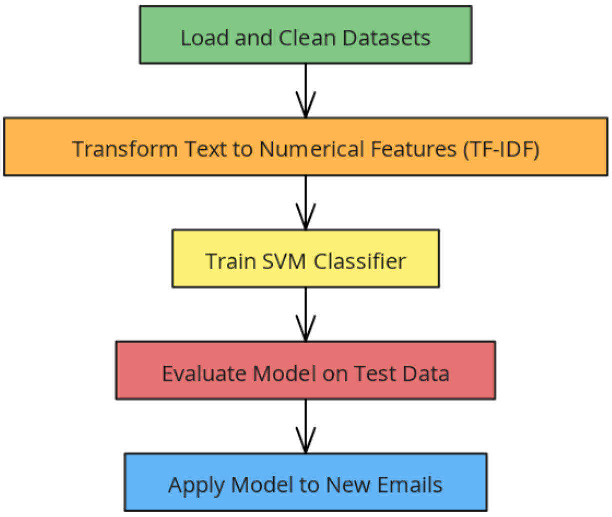
Load and Clean Datasets
Transform Text to Numerical Features (TF-IDF)
Train SVM Classifier
Evaluate Model on Test Data
Apply Model to New Emails

Figure 1. System Overview.

https://doi.org/10.47111/JTI
Available online at https://e-journal.upr.ac.id/index.php/JTI

## 2. Result and Discussion

The performance of the spam email classification model developed using Support Vector Machine (SVM) and TF-IDF vectorization was evaluated based on two well-known publicly available datasets: the TREC 2007 Public Spam Corpus and the Enron-Spam Dataset. The evaluation metrics, including accuracy, precision, recall, and F1-score, provide a comprehensive understanding of the model's effectiveness in distinguishing between spam and non-spam emails.

### Model Performance

The model achieved an overall accuracy of 99.04%, demonstrating its high ability to correctly classify both spam and non-spam emails. This level of accuracy is a strong indicator of the model's robust performance, suggesting that it is highly effective at distinguishing spam emails from legitimate ones in real-world scenarios. This result is consistent with findings in the literature, where SVM combined with TF-IDF has demonstrated high performance in spam detection tasks [20].

In addition to accuracy, precision and recall are essential to understanding the model's behavior in handling different types of errors. The precision of the model was calculated at 98.57%, which indicates that when the model predicted an email to be spam, it was correct 98.57% of the time. Precision is crucial in minimizing false positives, ensuring that legitimate emails are not mistakenly categorized as spam. This is important in practical applications where users are highly sensitive to misclassifying important emails. The effectiveness of SVM in achieving high precision in text classification tasks is well-documented, as it is particularly adept at handling high-dimensional data [13].

On the other hand, the recall of 99.62% highlights the model's effectiveness in identifying spam emails. Recall is a critical metric in spam detection because it measures how well the model can capture all the spam emails in the dataset. A high recall value like this ensures that the system successfully flags almost all spam messages, reducing the risk of missing malicious emails [21]. The combination of TF-IDF and SVM has effectively improved recall by focusing on distinctive, rare terms that are more likely to appear in spam emails [22]. These results show that the proposed model balances both precision and recall effectively, making it a suitable candidate for deployment in spam filtering systems where both minimizing false positives and detecting as many spam emails as possible are important.

### Dataset Characteristics and Preprocessing Impact

The model's success can be attributed to the robust preprocessing pipeline and the datasets' quality. The TREC 2007 Public Spam Corpus and the Enron-Spam Dataset contain various email samples, encompassing different spam techniques and legitimate content. The dataset preprocessing steps, such as text cleaning, tokenization, and removing stop words, were crucial in ensuring the input data was well-suited for the classification task.

The use of TF-IDF for feature extraction further enhanced the model's performance. Unlike traditional bag-of-words methods, TF-IDF emphasizes the importance of terms more unique to specific emails and less frequent across the entire corpus. This approach improves the model's ability to focus on the most relevant words, which are more likely to appear in spam emails [9]. Studies have demonstrated that TF-IDF is highly effective for spam detection due to its ability to capture the distinctive features of spam emails [9].

### Comparison with Other Approaches

Compared to other machine learning models and traditional rule-based methods, the combination of SVM and TF-IDF outperforms many existing spam detection techniques. Traditional rule-based systems, which rely on predefined rules and keyword matching, need help with evolving spam tactics, where new types of spam might bypass these rules. In contrast, the machine learning approach taken in this study can adapt to new patterns in spam data, ensuring better detection over time.

Moreover, SVM's strength lies in its ability to work well with high-dimensional data, which is typical in text classification tasks. Separating spam and non-spam emails through hyperplanes in high-dimensional feature space makes SVM particularly suitable [23]. Adding Principal Component Analysis (PCA) for dimensionality reduction also allowed for the visualization of this high-dimensional separation, providing insights into how the model distinguishes between the two classes [24].

### Challenges and Limitations

its impressive performance, several challenges and limitations remain. One significant challenge is the diversity of spam email content, affecting the model's generalization ability. Spam emails continuously evolve in content, language, and delivery methods, making it difficult for a model trained on historical data to detect new, unknown spam tactics. The model's performance might degrade if exposed to email types or formats significantly different from those in the training datasets.

Furthermore, the issue of zero-shot learning remains a concern. Since the model was trained on specific datasets, it may perform better when presented with new types of spam that were not represented in the training phase. To address this, transfer learning approaches could be explored, where models trained on one domain (e.g., general email data) can be fine-tuned with specific types of spam data to improve the detection of new spam messages.

Another limitation lies in the trade-off between precision and recall. While the model achieves a high recall, it still faces the possibility of overfitting the training data, particularly with imbalanced datasets where spam emails are more frequent. This could lead to a situation where the model excessively flags emails as spam, potentially increasing false positives.

### Future Work

Future research could enhance the model's ability to handle emerging spam tactics by incorporating additional features such as semantic analysis and natural language processing (NLP) techniques. Leveraging deep learning models such as recurrent neural networks (RNNs) or transformers could also improve the system's capacity to capture more complex patterns in email content.

Moreover, exploring online learning methods, where the model continually updates itself with new data, would help it adapt to the evolving nature of spam. This would address the challenge of zero-shot spam and keep the model relevant over time. A larger and more diverse set of datasets could also improve the model's generalizability to different types of spam emails from various domains.

## 3. Conclusion

In this study, the combination of TF-IDF and SVM proved to be an effective method for spam email classification. The model achieved high performance across various evaluation metrics, demonstrating its potential for deployment in practical spam filtering applications. Although there are challenges related to the adaptability of the model to emerging spam tactics, the results suggest that further refinement and incorporation of advanced techniques like transfer learning and deep learning can enhance the model's robustness. Future work will focus on addressing these limitations and exploring new avenues for improving the detection of novel spam messages.

## References

[1] F. Jáñez-Martino, E. Fidalgo, S. González-Martínez, and J. Velasco-Mata, "Classification of Spam Emails through Hierarchical Clustering and Supervised Learning," May 2020, [Online]. Available: http://arxiv.org/abs/2005.08773

[2] D. Mallampati and N. P. Hegde, "Feature Extraction and Classification of Email Spam Detection Using IMTF-IDF+Skip-Thought Vectors," *Ingénierie des systèmes d information*, vol. 27, no. 6, pp. 941–948, Dec. 2022, doi: 10.18280/isi.270610.

[3] A. Bhowmick and S. M. Hazarika, "E-Mail Spam Filtering: A Review of Techniques and Trends," 2018, pp. 583–590. doi: 10.1007/978-981-10-4765-7_61.

[4] I. H. Sarker, "Machine Learning: Algorithms, Real-World Applications and Research Directions," May 01, 2021, *Springer*. doi: 10.1007/s42979-021-00592-x.

[5] L. Alzubaidi *et al.*, "Review of deep learning: concepts, CNN architectures, challenges, applications, future directions," *J Big Data*, vol. 8, no. 1, Dec. 2021, doi: 10.1186/s40537-021-00444-8.

[6] J. Cervantes, F. Garcia-Lamont, L. Rodríguez-Mazahua, and A. Lopez, "A comprehensive survey on support vector machine classification: Applications, challenges and trends," *Neurocomputing*, vol. 408, pp. 189–215, Sep. 2020, doi: 10.1016/j.neucom.2019.10.118.

[7] S. Triest, A. Villaflor, and J. M. Dolan, "Learning Highway Ramp Merging Via Reinforcement Learning with Temporally-Extended Actions," in *2020 IEEE Intelligent Vehicles Symposium (IV)*, IEEE, Oct. 2020, pp. 1595–1600. doi: 10.1109/IV47402.2020.9304841.

[8] K. Kowsari, K. Jafari Meimandi, M. Heidarysafa, S. Mendu, L. Barnes, and D. Brown, "Text Classification Algorithms: A Survey," *Information*, vol. 10, no. 4, p. 150, Apr. 2019, doi: 10.3390/info10040150.

[9] Z. Yun-tao, G. Ling, and W. Yong-cheng, "An improved TF-IDF approach for text classification," *Journal of Zhejiang University-SCIENCE A*, vol. 6, no. 1, pp. 49–55, Aug. 2005, doi: 10.1631/BF02842477.

[10] L. Almazaydeh, M. Abuhelaleh, A. Al Tawil, and K. Elleithy, "Clinical Text Classification with Word Representation Features and Machine Learning Algorithms," *International Journal of Online and Biomedical Engineering (iJOE)*, vol. 19, no. 04, pp. 65–76, Apr. 2023, doi: 10.3991/ijoe.v19i04.36099.

[11] N. Kumar, S. Sonowal, and Nishant, "Email Spam Detection Using Machine Learning Algorithms," in *2020 Second International Conference on Inventive Research in Computing Applications (ICIRCA)*, IEEE, Jul. 2020, pp. 108–113. doi: 10.1109/ICIRCA48905.2020.9183098.

[12] F. Jáñez-Martino, R. Alaiz-Rodríguez, V. González-Castro, E. Fidalgo, and E. Alegre, "Classifying spam emails using agglomerative hierarchical clustering and a topic-based approach." [Online]. Available: https://talosintelligence.com/reputation_center/email_rep

[13] A. Dasgupta and S. Mehr, "Enhanced MNB Method for SPAM E-mail/SMS Text Detection Using TF-IDF Vectorizer," *American Journal of Mathematical and Computer Modelling*, vol. 9, no. 1, pp. 1–8, Apr. 2024, doi: 10.11648/j.ajmcm.20240901.11.

[14] V. S. Vinitha and D. K. Renuka, "Performance Analysis of E-Mail Spam Classification using different Machine Learning Techniques," in *2019 International Conference on Advances in Computing and Communication Engineering (ICACCE)*, IEEE, Apr. 2019, pp. 1–5. doi: 10.1109/ICACCE46606.2019.9080000.

[15] C. Wang, J. Zhou, H. Huang, and H. Shen, "Classification Algorithms for Unbalanced High-Dimensional Data with Hyperbox Vertex Over-Sampling Iterative Support Vector Machine Approach," in *2020 Chinese Control And Decision Conference (CCDC)*, IEEE, Aug. 2020, pp. 2294–2299. doi: 10.1109/CCDC49329.2020.9164585.

[16] B. Wang, L. Zhou, Y. Gu, and H. Zou, "Density-Convoluted Support Vector Machines for High-Dimensional Classification," *IEEE Trans Inf Theory*, vol. 69, no. 4, pp. 2523–2536, Apr. 2023, doi: 10.1109/TIT.2022.3222767.

[17] M. Adam *et al.*, "Sentiment Analysis on Acceptance of COVID-19 Vaccine for Children based on Support Vector Machine," *Journal of Advanced Research in Applied Sciences and Engineering Technology Journal homepage*, vol. 58, pp. 252–270, 2026, doi: https://doi.org/10.37934/araset.58.2.252270.

[18] M. Owusu-Adjei, J. Ben Hayfron-Acquah, T. Frimpong, and G. Abdul-Salaam, "A systematic review of prediction accuracy as an evaluation measure for determining machine learning model performance in healthcare systems," Jun. 04, 2023. doi: 10.1101/2023.06.01.23290837.

[19] S. Chua, A. Tan, P. N. E. Nohuddin, and M. H. Ahmad Hijazi, "Comparing the Effectiveness and Efficiency of Machine Learning Models for Spam Detection on Twitter," *Journal of Advanced Research in Applied Sciences and Engineering Technology*, pp. 127–138, Oct. 2024, doi: 10.37934/araset.61.2.127138.

[20] C. N. Mohammed and A. M. Ahmed, "A semantic-based model with a hybrid feature engineering process for accurate spam detection," *Journal of Electrical Systems and Information Technology*, vol. 11, no. 1, p. 26, Jul. 2024, doi: 10.1186/s43067-024-00151-3.

[21] M. A. Shaaban, Y. F. Hassan, and S. K. Guirguis, "Deep convolutional forest: a dynamic deep ensemble approach for spam detection in text," *Complex & Intelligent Systems*, vol. 8, no. 6, pp. 4897–4909, Dec. 2022, doi: 10.1007/s40747-022-00741-6.

[22] T. S. Guzella and W. M. Caminhas, "A review of machine learning approaches to Spam filtering," *Expert Syst Appl*, vol. 36, no. 7, pp. 10206–10222, Sep. 2009, doi: 10.1016/j.eswa.2009.02.037.

[23] C. Dewi, F. A. Indriawan, and H. J. Christanto, "Spam classification problems using support vector machine and grid search," *International Journal of Applied Science and Engineering*, vol. 20, no. 4, 2023, doi: 10.6703/IJASE.202312_20(4).006.

[24] J. Bernard, M. Hutter, M. Zeppelzauer, M. Sedlmair, and T. Munzner, "ProSeCo: Visual analysis of class separation measures and dataset characteristics," *Comput Graph*, vol. 96, pp. 48–60, May 2021, doi: 10.1016/j.cag.2021.03.004.