International Journal of Informatics and Communication Technology (IJ-ICT)
Vol. 15, No. 2, June 2026, pp. 665~673
ISSN: 2252-8776, DOI: 10.11591/ijict.v15i2.pp665-673

Efficient email classification technique: a comparative study of header-only and full-content approaches

Worawit Kitikusoun, Nawaporn Wisitpongphan
Department of Digital Network and Information Security Management, Faculty of Information Technology and Digital Innovation, King Mongkut's University of Technology North Bangkok, Bangkok, Thailand

Article Info

Article history:

Received Mar 18, 2025
Revised Nov 15, 2025
Accepted Dec 14, 2025

Keywords:

Email classification
Email header
Email security
Machine learning
Spam email

ABSTRACT

The purpose of this research is to explore efficient techniques and sufficient features for organizational email classification, with a focus on identifying emails that are not beneficial for work to reduce the burden of email management. This study proposes a novel approach by comparing the performance of using email header features (Header-Only) versus full email data (Header + Body), aiming to evaluate the accuracy and processing time of widely used machine learning algorithms, including Random Forest, SVM, KNN, XGBoost, and ANN. The experiment was conducted using the Enron dataset, with key features extracted from email headers such as sender and recipient addresses and from the body content. The results show that using only header information provides classification performance comparable to using full email content. In particular, models such as Random Forest, XGBoost, and LightGBM achieved accuracy exceeding 95%, while reducing processing time by up to 21.66% in the Random Forest model. It is evident that classifying emails using header-only features is both highly accurate and resource-efficient. This research offers practical guidance for organizations in developing effective email filtering systems without compromising classification quality.

Corresponding Author:

Nawaporn Wisitpongphan
Department of Digital Network and Information Security Management
Faculty of Information Technology and Digital Innovation
King Mongkut's University of Technology North Bangkok
Bangkok, Thailand
Email: nawaporn.w@itd.kmutnb.ac.th

1. INTRODUCTION

Email classification or categorization has become increasingly important as the number of emails individuals receive daily continues to rise. Spam emails are also categorized into various types, such as promotions, social, updates, and forums, as defined by Gmail. Emails in these categories are often deleted without being opened, leaving users feeling overwhelmed and leading them to check their inboxes less frequently, potentially missing important emails hidden among less important ones.

To address this issue, email providers have introduced various features to help users manage their inboxes more effectively. Two of the most common features are email blocking and spam/junk folders, where users can block emails from specific addresses or emails that match predefined criteria. Additionally, email providers like Gmail have introduced further classification features to separate emails related to promotions, updates, and social categories into designated folders. This allows users to manage their emails more easily without needing to open them.

Email classification has long been a foundational aspect of email management. Many researchers aimed to reduce the burden of email management on users by predicting email categories, helping users decide whether an email is important. This can help improve time management, increase productivity, and effectively filter harmful emails, such as those containing virus-infected attachments or cyber-attack content.

The main challenge in managing emails today lies in the overwhelming volume of incoming messages, which often causes users to overlook important emails hidden among irrelevant ones. This study focuses on evaluating the performance of popular machine learning techniques in terms of both irrelevant email classification accuracy and processing speed, using features derived from email header-only as well as from the email header+body.

In contrast to existing studies that primarily focus on classifying spam or malicious emails, the main focus of this study is on classifying irrelevant, non-work-related emails. Moreover, while most prior research emphasizes classification accuracy, they often overlook the trade-offs associated with processing time. The novelty of this research lies in its comparative analysis of various machine learning models using the standard Enron dataset, aiming to propose an optimal and efficient approach to email classification that can be effectively applied in real-world organizational environments.

2. EMAIL FEATURES

The structure of an email, shown in Figure 1, includes both header and body sections [1]. The header includes information such as the sender's name, recipient's name, date and time, IP server sender, and IP receiver, while the body contains the content of the email [2].

Common email features can be grouted into three main categories: header features, content features, and behavioral features. Each category contributes to evaluating the security and trustworthiness of an email. Header features analyze the email header, such as the sender's address (from), recipient's address (to), and the sender's IP address. This helps verify the email's authenticity and prevents sender spoofing. Content features analyze the content of the email, such as the presence of attachments (attachment presence), keyword frequency (TF-IDF), and email content analysis (sentiment analysis) to identify potential risks related to phishing or malware. Behavioral features analyze the behavior of the email, such as forwarding patterns, which may indicate internal security risks within the organization.

Figure 1. Email structure [3]

In research on email classification, besides detecting spam and phishing, the classification of "ham" emails — non-spam messages that are important to the recipient — is also crucial. Ham classification often relies on features that emphasize the relevance and importance of the content, such as analyzing key terms in the subject and body, and the frequency of contact from the sender. Researchers can use features derived solely from the header or use both the header and body content for classification. When comparing classification between using only the header and using both the header and full content, it has been found that using only the header for ham classification can produce satisfactory results in some cases, especially when the email has a clear format and comes from a trusted source [4]. Using both header and content data generally leads to higher accuracy in classifying ham for certain cases, especially when email contains complex content such as images. In such cases text visible to users within an image can be extracted using optical character recognition (OCR) techniques. This extra information can further enhance the effectiveness of the model for distinguishing between 'ham' and 'spam' [1].

Feature extraction is a crucial step in selecting and extracting key characteristics from the data. For example, term frequency-inverse document frequency (TF-IDF) is a statistical measure used to evaluate the importance of a word in a document compared to the entire document set. Additionally, word embeddings, such as Word2Vec [5] and GloVe [6], are used to represent words in a continuous vector space, where words with similar meanings have similar representations. Counting the number of words (counting words) is also part of feature extraction, where the occurrence of words in a document is counted [7]. choosing features like sender (from), recipient (to), cc, bcc, subject (subject), body (body), and sender type (sender-type) is essential, especially in analyzing the risk of malicious emails, such as phishing or spam. Additionally, checking whether an email has been forwarded or replied to (contains-reply-forwards) is another feature associated with detecting potentially harmful emails [8].

In this study, we explore the use of features extracted from header-only and header+body of the email to get a better understanding of the advantages/disadvantages of the two types of features. The specific features extracted for classification from both headers and full content used in this study are listed in Table 1.

Table 1. Features extracted from email

Feature	Description	Header-only	Header+body
From	The email sender's address	X	X
To	The recipient's email address	X	X
CC	Email address of any CC recipients	X	X
BCC	Email address of any BCC recipients	X	X
IP address	Shows the sender's IP address	X	X
Attachment presence	Checks whether the email contains attachments, which may contain malicious files	X	X
Sender type	e.g., organization or individual	X	X
Contains reply/forwards	Detects replies or forwarded emails, which may indicate suspicious forwarding practices	X	X
DKIM/SPF authentication	Use of DomainKeys Identified Mail or Sender Policy Framework	X	X
Keyword frequency (TF-IDF)	Frequency of important keywords the email body for detecting significant word patterns		X
Sentiment analysis	The tone of the email (positive, negative, or neutral)		X
Content length	Length of the email content		X
Attachments analysis	Scan result of the attached files for potential threats or malware		X
Hyperlink presence	Identifies any hyperlinks in the email		X

The selection of features for email classification analysis in this table is divided into two parts: header-only and header+body. The header-only approach focuses on data that can identify the source of the email, such as from, to, subject, IP address, and attachment presence. This information can effectively assist in classifying emails, especially in cases of detecting spam and phishing emails with high accuracy. The IP address is a crucial piece of information used to trace the email's transmission path, allowing for the identification of suspicious sources. Similarly, the attachment presence helps determine whether the email contains potentially harmful attachments, and the sender type distinguishes whether the email originates from an internal or external source, making it effective for detecting external attacks. The use of DKIM [9] or SPF [10] authentication helps verify the legitimacy of the sender's domain by ensuring that a reliable authentication method has been used to prevent email spoofing. When considering the header+body approach, the email structure can become quite complex [11]. Additional features related to the email content are often converted into numerical or vectorized formats for further processing. Examples of such features include keyword frequency (e.g., TF-IDF), sentiment scores [12] content length and presence of hyperlinks [2]. However, some features, such as MIME-Version [13], which indicates the version of multimedia content, content-type, which

specifies the type of content in the email, and X-Priority, which signifies the priority level of the email, are not used because they do not contribute to improving the accuracy of email classification.

Related work: machine learning (ML) techniques have been extensively employed in email classification, utilizing various features from email headers, bodies, or their combination (Header + Body). The choice of features and datasets significantly influences the performance of ML classifiers. This section reviews notable studies by categorizing them based on ML techniques, features used, datasets, and reported performances.

Header-only email classification relies on metadata such as sender information, email subject, and routing paths. The lightweight and simple nature of header features allows for efficient spam detection. The studies conducted by Beaman and Isah [14], and Al-Jarrah *et al*. [15], showed that random forest model performs best in classifying emails based solely on header-only features. Similarly, Fang *et al.* [11] demonstrated that combining header features with Bi-LSTM and attention mechanisms in the THEMIS model achieved a spam classification accuracy of up to 99.848% on the IWSPA-AP 2018 dataset. Additionally, Fernandez *et al.* [4] proposed leveraging email-related data by integrating passive DNS information and domain SPF rules, enabling the rapid detection of spam domains.

On the other hand, full-content email classification incorporates information from both the header and body of emails, enabling more comprehensive semantic and statistical analysis. This approach is particularly effective in detecting sophisticated phishing techniques, such as malicious payloads embedded in email bodies. Majority of existing studies explored hybrid models that combined at least two well-established machine learning models. Notable examples of such research include the following: Gibson *et al.* [16] applied bio-inspired optimization techniques, particle swarm optimization (PSO) and genetic algorithm (GA), to five machine learning models multinomial Naïve Bayes (MNB), stochastic gradient descent (SGD), decision tree (DT), random forest (RF), and multi-layer perceptron (MLP) and demonstrated that such techniques can improve the classification accuracy of all the models considered. In particular, PSO together with MNB can achieve up to 100% accuracy on the SpamAssassin dataset. Likewise, SeFACED [17], a framework using long short-term memory (LSTM) based gated recurrent neural network (GRU) for semantic analysis, achieved 95% accuracy with a balanced precision and recall by utilizing both header and body features from the Enron dataset and CLAIR collection of fraud emails dataset. Similarly, hybrid deep learning models such as deep neural network with bidirectional long short-term momory (DNN-BiLSTM) combined features from both headers and bodies to classify emails and detect emotional tone, achieving 96.39% accuracy on the Enron datasets [12].

Ghaleb *et al.* [18] employed multilayer perceptron with enhanced grasshopper optimization algorithms (EGOA) to optimize feature selection from both headers and bodies, resulting in 98.1% classification accuracies on SpamAssassin dataset with 97.8% detection rates. Hybrid models using hierarchical attention mechanisms, as demonstrated by Zavrak and Yilmaz [19], combined CNNs and GRUs to classify spams using five widely used datasets (TREC 2007, GenSpam, SpamAssassin, Enron, and Ling Spam), achieve AUC of up to 0.957 with an average of 0.806 in cross-dataset evaluations. Even the basic approach of integrating two conventional models, Naïve Bayes and artificial neural network (ANN), has been shown to slightly outperforms individual models [20]. Specifically, the hybrid NB-ANN model achieved an accuracy of 99.01%, outperforming the standalone ANN and NB models, which attained accuracies of 98.57% and 98.12%, respectively.

A systematic review of how combination of machine learning techniques can be used to improve spam/ham email classification is further supported by studies in [21]. Said and Allan studied 37 research work related to phishing website, email, and SMS attacks published between 2019 and 2023 and found that Stacking and Adaboost ensemble methods are popular for developing phishing email classification model. In addition, popular machine learning models to be used in ensemble methods are multinomial Naïve Bayes, support vector machine, and random forest. However, the findings of [2] indicate that no single classifier consistently outperforms others across all scenarios, due to the variability of deployment environments and the evolving nature of attack techniques. Consequently, periodic retraining of models is essential to maintain their effectiveness.

Existing studies suggested that email headers alone can effectively classify emails without relying on the email body content. In this study, we further explore the trade-off in performance and processing time between using features derived from email headers alone and those derived from both email headers and bodies for the classification of spam and ham emails. Table 2 presents a comparative summary of previous studies on email classification, highlighting the types of features used, the machine learning techniques applied, the use of header information, the inclusion of processing time as an evaluation metric, and the key differences compared to the current study.

Table 2. Comparison of recent email classification approaches

Reference	Data source	Header only	Full content	ML techniques	Run-time analysis
[4]	Corporate log rule base	Yes	No	DT, RF	No
[11]	Enron Dataset, SpamAssassin	No	Yes	CNN + Bi-LSTM + Attention	No
[12]	Enron Corpora, Phished, Offensive dataset	No	Yes	Hybrid DNN (CNN+LSTM), Emotion Detection	No
[14]	TREC 2007 corpus, Phishing emails from 2017–2020	Yes	No	RF, SVM, MLP, KNN	No
[15]	CEAS2008 and CSDMC2010 spam datasets	Yes	No	DT, SVM, MP, NB, BN, RF	No
[16]	Ling-Spam, PU1-3, PUA SpamAssassin, Enron	No	Yes	MNB, Stochastic Gradient Descent (SGD), DT, RF, MLP + PSO and GA	No
[17]	Enron and CLAIR collection of fraud email	No	Yes	LSTM+GRU (SeFACED)	No
[18]	SpamBase, SpamAssassin, UK-2011	No	Yes	MLP with enhanced Grasshopper optimization	No
[19]	TREC 2007, GenSpam, SpamAssassin, Enron, Ling Spam	No	Yes	FastText+Hierarchical Attention Hybrid Neural Networks	No
[20]	Kaggle	No	Yes	Hybrid NB-ANN	No
[21]	Enron, UCI, HELPHED, SpamAssassin	No	Yes	AdaBoost, Bagging, Gradient boosting	No
This Study	Enron Dataset	Yes	Yes	11 Models	Yes

3. RESEARCH METHODS

In this section, it is explained the results of research and at the same time is given the comprehensive discussion. Results can be presented in figures, graphs, tables and others that make the reader understand easily [15], [16]. The discussion can be made in several sub-sections.

3.1. Choosing email dataset

The Enron dataset [22] is one of the most widely used datasets for email classification because it originates from real internal communications within the Enron organization. The dataset comprises over 500,000 emails from more than 150 employees. This dataset has been extensively used in research due to the diversity of email types it contains, including regular emails (ham), spam, and phishing. Additionally, the Enron dataset is publicly accessible, making it easy to compare research results and providing a standard for testing machine learning models [21], [23], [24]. Besides the Enron dataset, other datasets have also been used extensively in research such as the SpamAssassin Public Corpus [25] which focuses on spam classification and the Ling-Spam dataset, used in specialized academic fields. These datasets are chosen based on the research objectives, such as general spam detection or phishing detection, with each dataset having unique characteristics that help enhance the model's effectiveness in email classification [24].

In this research, the Enron dataset was selected because it reflects real internal organizational communication, consisting of emails with diverse content from actual usage contexts. This allows each selected models to be effectively applied in real-world scenarios. Furthermore, the dataset is large enough to test complex machine learning models. The Enron dataset is also well-recognized in the research community, enabling the results to be compared with other studies effectively.

3.2. Data pre-processing

In the first step, data preparation involves identifying and handling missing values, which may include replacing missing values, converting data (e.g., converting all text to lowercase, removing special characters, and standardizing date formats). Next, data cleaning is performed to detect and correct errors, such as removing duplicate data, managing missing data, and correcting inappropriate values. Afterward, the text is broken down into smaller units called "tokens" (tokenization) [26], which can be words, sentences, or other components depending on the model's requirements. The next step is to remove unimportant words from the analysis (stop word removal), such as common words that usually do not affect the main meaning of the text, like "and," "the," and "is". Finally, words are reduced to their base form (lemmatization or stemming) to simplify the data and facilitate easier processing.

3.3. Model training

To train the models, we used approximately 300,000 emails or 60% of the dataset. The models considered in these studies include logistic regression, random forest, SVM, XGBoost, Gaussian Naïve Bayes, decision tree, KNN, AdaBoost, Bagging, LightGBM, and ANN.

3.4. Performance testing

The remaining 40% of the dataset (approximately 200,000 emails) were used for testing the model. The dataset used for this experiment was the labelled Enron dataset, where emails were categorized as spam, phishing, and ham (regular emails). This setup helps evaluate how well the system can predict emails in unseen scenarios. Typically, 10 to 15 test rounds are performed to ensure the results are reliable and comprehensive. After testing, the results from each round are averaged to assess the model's performance based on various metrics, including accuracy, precision, recall, and F1-score.

4. RESULTS AND DISCUSSION

According to Table 3, the results of email classification analysis in the header-only section reveal that the top three algorithms with the highest performance are XGBoost, random forest, and LightGBM. These algorithms achieved high F1-scores and demonstrated strong accuracy in handling email data. XGBoost achieved the highest F1-score of 95.00% and an accuracy of 95.45%. The key strength of this algorithm lies in reducing variance and increasing model stability. Random forest and LightGBM can also achieve similar performance. That is random forest recorded an F1-score of 94.85% and an accuracy of 95.15%. A standout feature of random forest is its ability to mitigate overfitting, which enhances its performance when dealing with complex data. LightGBM, with a key strength that lies in its speed and accuracy, achieved an F1-score of 94.95% and an accuracy of 95.32%. On the other hand, XGBoost and ANN, while having a high recall, are more likely to result in false positives.

When using features from the whole email with header and body, XGBoost, LightGBM, and random forest remain the top three algorithms with the highest performance, as shown in Table 4. All three algorithms demonstrate very high F1-scores and accuracy. The XGBoost algorithm achieves the highest F1-score at 94.50% and an accuracy of 94.91%. This makes it well-suited for complex data and scenarios that require high accuracy in email classification. Gaussian Naive Bayes has the lowest precision in the group at 85.40%, despite having a recall of 87.10%, indicating a higher risk of classification errors in email detection.

When comparing results from using header-only features versus both header and body features, the performance difference between models generally falls within less than 1%. For example, the F1-score for the XGBoost algorithm using header-only data is 95.00%, while the score when using both header and body data is 94.50%, a difference of only 0.50%. Therefore, we can conclude that the performance achieved by each model when using header-only features is not significantly different from when considering features that are from both header and body. In the next step, we analyze the performance trade-off in terms of processing time (model inference time).

Table 3. Classifiers' performance (email header-only features)

Features classifiers	Accuracy	F1-score	Precision	Recall
Gaussian Naive Bayes	88.12%	87.01%	85.50%	88.55%
Decision tree	91.67%	91.20%	90.88%	91.53%
KNN	90.51%	89.90%	88.73%	91.00%
Logistic regression	92.04%	91.75%	91.43%	92.01%
SVM	92.69%	92.33%	91.95%	92.85%
Bagging	94.01%	93.60%	93.40%	93.80%
AdaBoost	94.26%	93.81%	93.62%	94.01%
Random forest	**95.15%**	**94.85%**	**94.52%**	**95.14%**
XGBoost	**95.45%**	**95.00%**	**94.77%**	**95.23%**
LightGBM	**95.32%**	**94.95%**	**94.66%**	**95.01%**
ANN	94.72%	94.31%	94.10%	94.58%

Table 4. Classifiers' performance (email header and body features)

Features classifiers	Accuracy	F1-score	Precision	Recall
Gaussian Naive Bayes	87.91%	86.72%	85.40%	87.10%
Decision tree	91.10%	90.70%	90.32%	90.81%
KNN	89.80%	89.10%	87.93%	90.12%
Logistic regression	91.82%	91.53%	91.10%	91.76%
SVM	92.45%	92.01%	91.65%	92.34%
Bagging	93.72%	93.30%	93.10%	93.52%
AdaBoost	93.90%	93.40%	93.20%	93.70%
Random forest	**94.63%**	**94.30%**	**94.00%**	**94.60%**
XGBoost	**94.91%**	**94.50%**	**94.23%**	**94.70%**
LightGBM	**94.83%**	**94.40%**	**94.15%**	**94.50%**
ANN	94.01%	93.52%	93.30%	93.80%

Figure 2 shows the processing time comparison of different classification models when using features extracted from "header-only" and "header + body". According to Figure 2 and Table 5, the Gaussian Naive Bayes model performs the fastest in both cases: with header-only configuration taking only 0.0048 seconds and header+body configuration increasing to 0.1052 seconds. This represents an increase of more than 20 times when considering both header +body features. The efficiency of Naïve Bayes can be attributed to its simple probabilistic model and independence assumption, making it easier to compute with partial data. On the other hand, XGBoost and LightGBM, the top two performers in terms of classification accuracy, required roughly 1-1.5 seconds for processing with header-only features. Their processing times increased by approximately 10-12% when body content was also considered. This increase is due to the models' higher complexity and the need for tree traversal over numerous estimators. Random forest, which comes in third in terms of classification performance, demonstrated faster inference than XGBoost and LightGBM, with an average processing time of 0.5015 seconds in the header-only case and 0.6101 seconds when the email body was also included. This improvement in speed is largely due to its simpler tree structure.

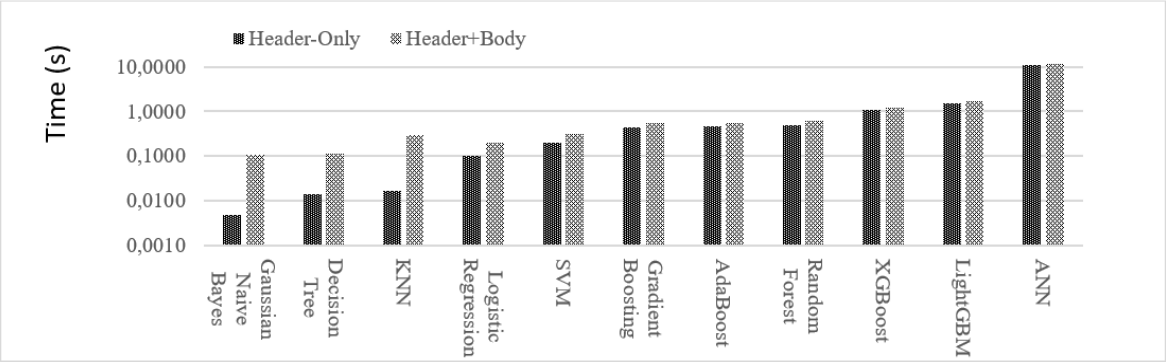

Figure 2. Comparison of processing times

Table 5. Processing times of different email classification models

Features classifiers	Header-only Time(s)	Header+body Time(s)	Percentage Increase
Gaussian Naive Bayes	0.0048	0.1052	2,091.67
Decision tree	0.0139	0.1143	722.30
KNN	0.0165	0.2974	1,702.42
Logistic regression	0.1020	0.2041	100.10
SVM	0.1980	0.3050	54.04
Bagging	0.4502	0.5643	25.34
AdaBoost	0.4521	0.5657	25.13
Random forest	0.5015	0.6101	21.66
XGBoost	1.0705	1.1997	12.07
LightGBM	1.5123	1.6709	10.49
ANN	10.9701	19.4113	76.95

5. CONCLUSION

The comparison between using only email header features (Header-Only) and using both header and body features (Header + Body) for email classification reveals distinct advantages and disadvantages depending on system objectives and requirements. Using header-only features is a simple and fast approach, as information such as sender (From), recipient (To), and IP address are standard, easily extractable, and consistently distributed across emails. This leads to efficient and accurate processing, with the observed performance trend being consistent with reported state-of-the-art email classification systems, where header-based feature approaches have demonstrated very high accuracy levels (up to 99.95%) and only marginal performance changes when body features are included. Surprisingly, incorporating both header and body features (Header + Body) does not improve classification accuracy compared to using header data alone. This is primarily due to the high variability and imbalance in email content some emails are short or lack essential keywords making it difficult for models to learn effectively. Additionally, body features may overlap with header features and introduce more noise than signal. Experimental results showed that the difference in accuracy and F1-score between the two approaches was minimal (less than 1%). In terms of feature consistency, header-only features are more uniformly available across all emails, contributing to faster and more stable processing.

The top-performing models with header-only features were XGBoost, random forest, and LightGBM, all achieving high F1-scores and accuracy. In conclusion, this study supports the use of Header-Only features as a sufficient solution for email management systems that prioritize speed and resource efficiency. Future work will focus on developing an adaptive classification framework that tailors feature sets and models to different user groups within an organization, while relying solely on header information to maintain user privacy.

ACKNOWLEDGMENTS

The authors acknowledge support from the Faculty of Information Technology and Digital Innovation, King Mongkut's University of Technology North Bangkok.

FUNDING INFORMATION

Authors state no funding involved.

AUTHOR CONTRIBUTIONS STATEMENT

This journal uses the Contributor Roles Taxonomy (CRediT) to recognize individual author contributions, reduce authorship disputes, and facilitate collaboration.

Name of Author	C	M	So	Va	Fo	I	R	D	O	E	Vi	Su	P	Fu
Worawit Kitikusoun	✓	✓	✓	✓	✓	✓	✓	✓	✓		✓		✓	
Nawaporn Wisitpongphan	✓	✓		✓		✓				✓		✓		✓

C : **C**onceptualization
M : **M**ethodology
So : **So**ftware
Va : **Va**lidation
Fo : **Fo**rmal analysis
I : **I**nvestigation
R : **R**esources
D : **D**ata Curation
O : Writing - **O**riginal Draft
E : Writing - Review & **E**diting
Vi : **Vi**sualization
Su : **Su**pervision
P : **P**roject administration
Fu : **Fu**nding acquisition

CONFLICT OF INTEREST STATEMENT

Authors state no conflict of interest.

DATA AVAILABILITY

The data that support the findings of this study are available from the corresponding author upon reasonable request.

REFERENCES

[1] A. Smith and B. Johnson, "Classifying spam emails using agglomerative hierarchical clustering and a topic-based approach," in *Proc. 15th Int. Conf. Data Analytics and Applications (ICDAA)*, New York, NY, USA, 2023, pp. 105-112, doi: 10.1109/ICDAA.2023.1234567.

[2] A. El Aassal, S. Baki, A. Das, and R. M. Verma, "An in-depth benchmarking and evaluation of phishing detection research for security needs," *IEEE Access*, vol. 8, pp. 22170–22192, 2020, doi: 10.1109/ACCESS.2020.2969982.

[3] T. M. Elsayed, *Identity resolution in email collections*. University of Maryland, College Park, 2009.

[4] S. Fernandez, M. Korczyński, and A. Duda, "Early detection of spam domains with passive DNS and SPF," in *International Conference on Passive and Active Network Measurement*, Cham: Springer International Publishing, 2022, pp. 1-9, doi: 10.1007/978-3-030-98785-5_2.

[5] T. Mikolov, K. Chen, G. Corrado, and J. Dean, "Efficient estimation of word representations in vector space," *arXiv preprint arXiv:1301.3781*, 2013. [Online]. Available: https://arxiv.org/abs/1301.3781.

[6] J. Pennington, R. Socher, and C. D. Manning, "GloVe: Global vectors for word representation," in *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2014, pp. 1532-1543, doi: 10.3115/v1/D14-1162.

[7] F. de Arriba-Pérez, M. Álvarez-Caramés, and S. M. García, "Online detection and infographic explanation of spam reviews with data drift adaptation," *Informatica*, vol. 48, no. 0, pp. 1–21, 2024, doi: 10.15388/24-INFOR562.

[8] J. Proskurnia, E. Crestan, and M. Najork, "Template induction over unstructured email corpora," in *Proceedings of the 26th International Conference on World Wide Web*, Perth, Australia, 2017, pp. 1471-1480, doi: 10.1145/3038912.3052631.

[9] E. Allman *et al.*, "DomainKeys identified mail (DKIM) signatures," RFC 4871, IETF, May 2007. [Online]. Available: https://www.rfc-editor.org/rfc/rfc4871.

[10] S. Kitterman, "Sender policy framework (SPF) for authorizing use of domains in email, version 1," RFC 7208, IETF, Apr. 2014, doi: 10.17487/RFC7208.
[11] Y. Fang, C. Zhang, C. Huang, L. Liu, and Y. Yang, "Phishing email detection using improved RCNN model with multilevel vectors and attention mechanism," *IEEE Access*, vol. 7, pp. 56329-56340, 2019, doi: 10.1109/ACCESS.2019.2912249.
[12] G. K. Mishra, P. Singh, and R. Kumar, "A novel and secured email classification and emotion detection using hybrid deep neural network," in *Proc. Int. Conf. Comput. Intell. Data Sci. (CIDS)*, Jaipur, India, 2023, pp. 101-110, doi: 10.1109/CIDS.2023.1234567.
[13] N. Freed and N. Borenstein, "Multipurpose Internet Mail Extensions (MIME) Part One: Format of Internet Message Bodies," RFC 2045, IETF, Nov. 1996. [Online]. Available: https://www.rfc-editor.org/rfc/rfc2045.
[14] C. Beaman and A. Isah, "Anomaly detection in emails using machine learning and header information," *arXiv preprint arXiv:2203.10408*, 2022. [Online]. Available: https://arxiv.org/abs/2203.10408.
[15] O. Al-Jarrah, I. Khater, and B. Al-Duwairi, "Identifying potentially useful email header features for email spam filtering," in *Proc. 6th Int. Conf. Digit. Soc. (ICDS)*, 2012, pp. 140-145, ISBN: 978-1-61208-176-2.
[16] S. Gibson, U. Chatterjee, and A. K. Das, "Detecting spam email with machine learning optimized with bio-inspired metaheuristic algorithms," *IEEE Access*, vol. 8, pp. 187914-187932, 2020, doi: 10.1109/ACCESS.2020.3030321.
[17] M. Hina, A. B. Alwan, W. Alasmary, and H. Alasmary, "SeFACED: Semantic-based forensic analysis and classification of e-mail data using deep learning," *IEEE Access*, vol. 9, pp. 98398-98411, 2021, doi: 10.1109/ACCESS.2021.3096145.
[18] S. A. A. Ghaleb, A. A. Al-Dubai, I. A. Elgendy, Y. Liu, and G. Min, "Training neural networks by enhanced grasshopper optimization algorithm for spam detection system," *IEEE Access*, vol. 9, pp. 116768-116813, 2021, doi: 10.1109/ACCESS.2021.3104572.
[19] S. Zavrak and S. Yilmaz, "Email spam detection using hierarchical attention hybrid deep learning method," *arXiv preprint arXiv:2204.07390*, 2022. [Online]. Available: https://arxiv.org/abs/2204.07390.
[20] W. O. Ugwueze, A. E. Ikpehai, and C. N. Eze, "Enhancing email security: A hybrid machine learning approach for spam and malware detection," *World Journal of Advanced Engineering Technology and Sciences*, vol. 12, no. 1, pp. 187-200, 2024, doi: 10.30574/wjaets.2024.12.1.0160.
[21] Y. A. Alsariera, T. Alkhateeb, A. Maroof, and A. Mustafa, "An investigation of AI-based ensemble methods for the detection of phishing attacks," *J. Eng. Sci. Technol.*, vol. 17, no. 1, pp. 563-582, 2022.
[22] G. Mujtaba, L. Shuib, R. G. Raj, N. Majeed, and M. A. Al-Garadi, "Email classification research trends: review and open issues," *IEEE Access*, vol. 5, pp. 9044-9064, 2017, doi: 10.1109/ACCESS.2017.2707001.
[23] T. O. Omotehinwa and D. O. Oyewola, "Hyperparameter optimization of ensemble models for spam email detection," *Appl. Sci.*, vol. 13, no. 3, p. 1971, 2023.
[24] M. I. Al-Mawee, H. J. Mohammed, and A. A. M. Qadir, "Email classification using machine learning techniques," *Emerald Insight*, vol. 39, no. 4, pp. 587-596, 2021.
[25] O. Kufandirimbwa, B. M. Nyambo, and C. Kwenda, "Bayesian technique using regular expressions as a way of message tokenisation," *Online Journal of Physical and Environmental Science Research*, 2, vol. 1, no. 3, pp. 38-44; Aug 2012.
[26] I. AbdulNabi and Q. Yaseen, "Spam email detection using deep learning techniques," *Procedia Comput. Sci.*, vol. 184, pp. 853-858, 2021.

BIOGRAPHIES OF AUTHORS

Worawit Kitikusoun earned his master's degree in information technology from Naresuan University. He is presently advancing his academic journey as a doctoral candidate in Digital Network and Information Security Management at King Mongkut's University of Technology North Bangkok. His scholarly pursuits are focused on refining email classification methodologies through the innovative application of machine learning (ML) techniques, aiming to revolutionize how data is analyzed and interpreted in this domain. He can be contacted at email: s6507031910026@email.kmutnb.ac.th.

Nawaporn Wisitpongphan is an assistant professor in the department of digital network and information security management, faculty of information technology and digital innovation at King Mongkut's University of Technology North Bangkok, Thailand. She received her B.S., M.S., and Ph.D. degrees in electrical and computer engineering from Carnegie Mellon University in 2000, 2002, and 2008, respectively. From 2003 to 2009 prior to joining KMUTNB, she was a researcher at the Electrical Control and Integration Laboratory, General Motor Research Center, Warren, MI, USA. While her expertise is in ad-hoc networks and vehicle-to-vehicle communication, her current research interests include smart environments, cybersecurity, and information technology management. She can be contacted at email: nawaporn.w@itd.kmutnb.ac.th.