NTU Journal of Engineering and Technology (2025) 4 (3): 9-14
DOI: https://doi.org/10.56286/ntujet.v4i3

IRAQI Academic Scientific Journals

P-ISSN: 2788-9971 E-ISSN: 2788-998X
NTU Journal of Engineering and Technology
Available online at: https://journals.ntu.edu.iq/index.php/NTU-JET/index

Comparative Analysis of Machine Learning Algorithms for Phishing Email Detection

Raweia S MohamedAli[1], Razan Abdulhammed[2]
[1]Computer Engineering Department, Technical Engineering College_Mosul, Northern Technical University, Mosul, 41001 , Iraq.
[2]Technical Engineering College for Computer and AI., Northern Technical University, Mosul- Iraq.
rawea.salem@ntu.edu.iq, rabdulhammed@ntu.edu.iq

Article Information

Received: 19-02- 2024,
Revised: 22-05-2024,
Accepted: 22-05-2024,
Published online: 28-09-2025

Corresponding author:
Name: Razan Abdulhammed
Affiliation: Northern Technical University
Email:rabdulhammed@ntu.edu.iq

Key Words:
Cybersecurity,
Phishing Detection,
Machine Learning,
Artificial Intelligence.

A B S T R A C T

Nowadays,The danger of cyberattacks grows as technology develops, requiring more advanced detection and prevention methods. With an emphasis on e-mail phishing detection, the study explores the use of machine learning (ML) to improve cybersecurity measures. Support Vector Machine (SVM), Random Forest (RF), Decision Tree (DT), Logistic Regression (LR), and CatBoost are among the ML models that are assessed to determine how well they can differentiate between secure and phishing e-mails. F1-score, recall, accuracy, and precision are among the evaluation measures. The results show that all models perform well, with SVM showing perfect accuracy. These findings highlight the importance of cutting-edge technologies in strengthening cybersecurity defenses against changing cyber threats.

1. Introduction

The rapid technological growth and digitalization has made more individuals rely on online platforms for daily requirements such as studying, shopping, transactions, and accessing services. It is now expected to open a smart device and quickly visit the websites of pharmacies, retailers, libraries, and educational institutions [1].Thus, E-services become more popular than in the last decade which lead to an increase of cyber attackers' dangers. Here, Attackers use weaknesses in the digital world to access and misuse sensitive user data, such as credit card information, names, phone numbers, and identifying details [2]. Phishing is a common technique used by cybercriminals through through various channels such as e-mail, SMS, or phone URLs [3].

Phishing attacks intened to compromise personal data and internet accounts credintials. Attackers (Hackers) utilize various strategies, such as deceiving customers with authentic URLs resembling retail or banking websites or breaking into company networks without authorization to carry out more nefarious acts [4]. One type of Phishing attack, URL phishing, manipulates e-mails and URLs to trick consumers into thinking they communicate electronically with a reliable source [5].

The development of intelligent technologies, particularly machine learning (ML) becomes apparent in the face of these obstacles as a critical component in improving cybersecurity. Because of its many features, ML is spanning pattern recognition of adaptive security measures [6].

Machine learning can reduce the requirement for human expertise in feature extraction and selection of phishing attempt detection ML models [7]. By using machine learning research were able to demonstrates the superior performance of these models, emphasizing its accuracy and efficiency [8].

The aggressive creation of anti-phishing technology based on machine learning, indicates an industry-wide effort to recognize and thwart new phishing attacks [9]. The combination of cutting-edge technology is essential in an ever-changing environment to protect users from malicious cyber activity and guarantee online interaction security [10].

2. Related Works

Research has been devoted to identifying and categorizing phishing e-mails that are extremely dangerous for digital Economy. This section take a look at related work regarding Email security. To start with, Bergholz et al. utilized feature selection in building phishing detection models. Bergholz et al. utlized a combination of feature set along side Random Forest and SVM algorithms [11]. One drawback of Bergholz et al. was emitting important detiles on the authors models making it is hard to carry on further future work.

A phishing e-mail classification using structural features and various algorithms have were developed by Basnet et al. It include SVM, neural networks, Self-Organizing Maps (SOMs), and K-means clustering. The model achieved 90% accuracy using the k-means clustering approach [12].

Sarju and Thomas employed structural properties to detect spam e-mails, utilizing Naïve Bayes, Adaboost, and Random Forest algorithms. A majority voting algorithm achieved 99.8% accuracy showcasing the effectiveness of ensemble methods in spam detection [13].

In [16], authors utilized phishing term weighting. They employed Random Forest and J48 algorithms, achieving an accuracy of 99.1% with Random Forest and 98.4% with J48 [14].

Zhang et al. tackled phishing detection using semantic analysis with machine learning algorithms. Their approach achieved meager error rates, with a specific focus on identifying phishing websites, contributing to the broader landscape of anti-phishing strategies [15].

Rawal et al. conducted a comprehensive study comparer the affective of SVM, Random Forest, Logistic Regression, Naïve Bayes, and Voted Perceptron to distinguish among phishing and save e-mails. Their work demonstrated an impressive maximum accuracy of 99.87%, highlighting the robustness of multiple classifiers in tackling e-mail phishing threats [16].

Yang et al. employed 18 hybrid features and the SVM algorithm. The incorporating achieved 95% classification accuracy [17].

Fang et al. introduced a Recurrent Convolutional Neural Network (CNN) model with multilevel vectors for phishing e-mail detection using the Themis algorithm. Their model exhibited a very high classification accuracy of 99.84%, [18].

Verma et al. employed Natural Language Processing (NLP) to classify phishing e-mails to tackle phishing threats through language analysis [1].

Eckhardt and Bagui explored phishing e-mail classification employed Long Short-Term Memory (LSTM) and Convolutional Neural Network (CNN) architectures. The architectures demonstrated comparable performances [19].

3. Methodology

3.1 Model Block Diagram

Figure 1 shows proposed model block digram.The target of the proposed system id to for classify e-mails as either "Safe E-mail" or "Phishing E-mail." As highlited in Fig. 1, It begins by loading a splitted dataset of training and testing sets. Missing values are handled, and text data is transformed into numerical features using TF-IDF vectorization. The logistic regression classifier is then defined, trained, and used for predictions. The model's performance is evaluated using accuracy, a confusion matrix, and a classification report, visualizing results through a heatmap. The block diagram captures the workflow, representing the ML model's steps and performance evaluation.

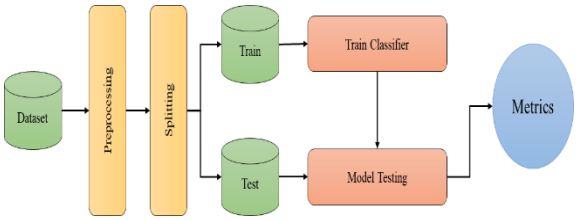


Fig. 1. Model Block Diagram

3.2 Dataset Description

The supplied dataset (https://www.kaggle.com/datasets/subhajournal/phishingemails/data) includes two primary features: "E-mail Text" and "E-mail Type." The former pertains to the e-mail body, while the latter designates whether the e-mail is classified as "Safe" or "Phishing." The dataset has 18,101 entries, with distribution detailed based on the word count ranges given in the "Label" column. Every entry has a word count range allocated to it, and word counts are associated with "safe" and "phishing" e-mails. The dataset is a valuable tool for analyzing and creating models that can differentiate between safe and phishing e-mails since it provides a thorough representation of e-mail text changes and the classifications that go along with them.

3.3 Dataset Preprocessing

A crucial step in machine learning is preprocessing, which converts unprocessed input into a format that can be used for model training and assessment. The first step is to import the dataset into a data structure (usually a Pandas Data Frame) from a structured file (like a CSV file).

The dataset is frequently divided into training and testing sets, making evaluating the model's performance on previously untested data easier. It is essential to handle missing values in order to provide a complete and consistent dataset. Tokenization, stemming, and lemmatization are preprocessing methods frequently used in textual data to standardize and break down words into their most basic forms. The dataset is further refined by eliminating special characters and stop words.

Vectorization is a fundamental process that turns text into numerical properties. Commonly used methods include TF-IDF (Term Frequency-Inverse Document Frequency), which weights terms according to their significance within the corpus. After that, machine learning models are successfully trained using this numerical representation.

3.4 Machine Learning Models

For the proposed system a SVM classifier is employed. The SVM classifier work wether the data can or cannot be separated linearly by using different kernels types that enable SVM to handle high-dimensional data effectively. Equation below describe the model [20].

$$\underset{w}{Minimize}\ \frac{1}{2}\|W\|^2 \quad (1)$$

Subject

$$y_i(\langle W, X_i\rangle + b) \geq 1 \quad (2)$$

$$Y.(Xw + b) \geq 1 \quad (3)$$

$$\begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{bmatrix} \cdot \left(\begin{bmatrix} x_{11} & x_{12} & \cdots & x_{1d} \\ x_{21} & x_{22} & \cdots & x_{2d} \\ \vdots & \vdots & \ddots & \vdots \\ x_{n1} & x_{n1} & \cdots & x_{nd} \end{bmatrix} \begin{bmatrix} W_1 \\ W_2 \\ \vdots \\ W_d \end{bmatrix} + b \right) \geq 1^n \quad (4)$$

The decision function is represented as w·x+b, where w denotes the weight vector, x corresponds to the input vector, and b signifies the bias term. The second classifier used is the Random Forest , during the training phase RF constructs

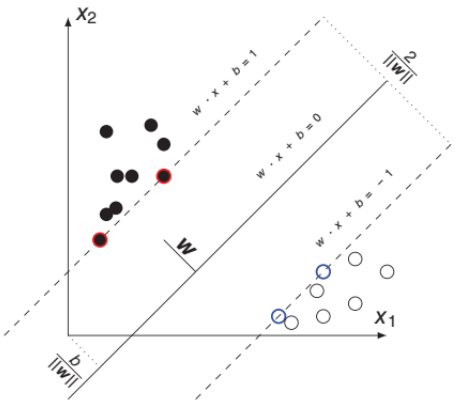


Fig. 2. SVM Model.

multiple decision trees to generates a class output reflecting the mode of the classes derived from the individual trees. Random Forests are effective in managing high-dimensional datasets and offer a gauge for feature relevance [21].

The third algortim utlized in the prposed system is Decision Tree classifier. DT generates a

tree-like structure by iteratively partitioning the data based on the most significant attributes. Decision trees facilitate an understanding of the decision-making process due to their ease of interpretation and visualization. [22].

The fourth algorthim used is the Logistic regression. The algorthim generates an output between 0 and 1 by applying the logistic function to the weighted sum of the input features. Equation (5) describe model [23].

$$P(Y-1) = \frac{1}{1+e^{-(\beta_0+\beta_1 x_1+\beta_2 x_2+\cdots\ldots\ldots+\beta_N x_N)}} \quad (5)$$

Here, P(Y=1) is the probability of the positive class, Xi are the input features, and βi are the coefficients.

The final classifier used is the CatBoost. The core idea for this classifier is similar to RF and DT, in which it creates a series of decision trees; each tree in the series learns from the mistakes of the previous one. To avoid the overfitting and one-hot encoding issues, order boosting alongside automatic categorial handling can be used during the implementation process [24].

3.5 Evaluation Metrics

To evaluate the performance of the proposed system, the study relied upon Accuracy, Precision, Recall, and F1. Moreover, a confusion matrix is utilized alongside these metrics [25]. Accuracy is the primary evaluation metric, representing the ratio of correctly classified instances to the total number of instances. Equation 6 represent Accuracy.

$$Accuracy = \frac{Tp+Tn}{TP+TN+FP+FN} \quad (6)$$

Precision is the ratio of true positives compared to the total of true negatives and false positives. The precision is calculated using equation 7

$$Precision = \frac{TP}{TP+FP} \quad (7)$$

Recall is the ratio of true positives to the combination of true positives and false negatives. Recall value is evaluated using equation 8

$$Recall = \frac{TP}{TP+FN} \quad (8)$$

The F1-Score is the harmonic mean of both precision and recall and estimated using equation 9.

$$F\_1 - Score = 2 \times \frac{percision \times recall}{Percison+recall} \quad (9)$$

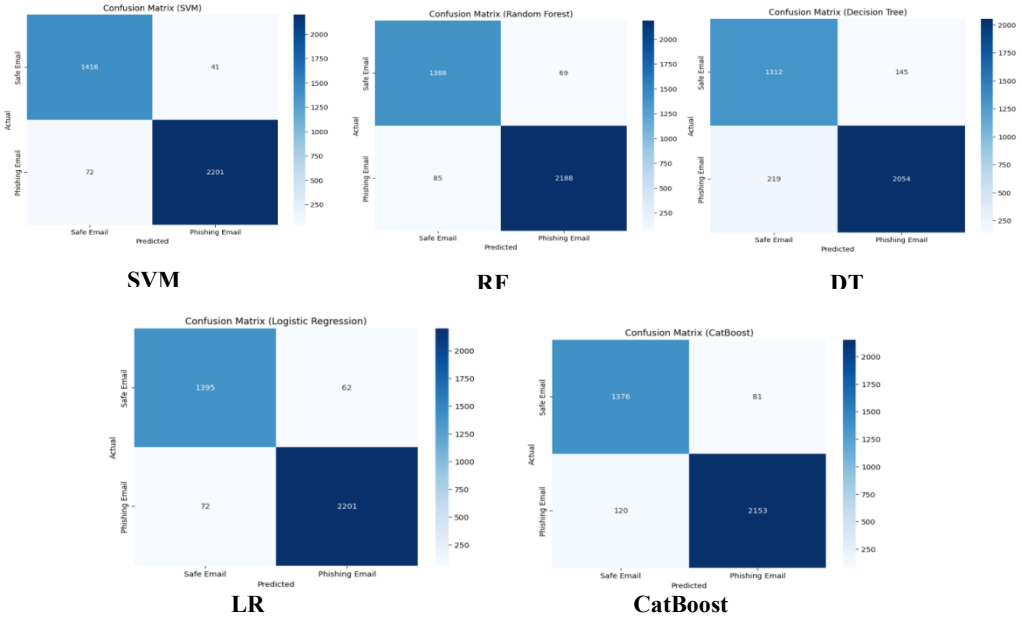
The confusion matrix organizes the model's predictions into four categories: true positives, true negatives, false positives, and false negatives.

4. Results

The evaluation results display the performance metrics of five machine learning algorithms – Support Vector Machine (SVM), Random Forest (RF), Decision Tree (DT), Logistic Regression (LR), and CatBoost. Fig. 2 displays confusion matrices for the performance of the proposed system.

Support Vector Machine (SVM) emerges as the top-performing algorithm with an outstanding accuracy of 97%, complemented by equally high precision, recall, and F1-score values, all standing at 0.97. Random Forest (RF) and Logistic Regression (LR) closely follow with accuracies of 96. While Decision Tree (DT) exhibits slightly lower metrics with an accuracy of 90.

CatBoost, achived accuracy of 95%. precision, recall, and F1-score metrics are consistently high, the F1-score were 92%.



SVM **RF** **DT**

LR **CatBoost**

Fig. 3. Models Confusion Matrix.

The confusion matrix results in Fig. 3 highlight performance of the proposed system such that SVM showed 1416 True Positives, 2201 True Negatives, 72 False Positives, and 41 False Negatives.

Random Forest demonstrates strong performance with 1388 True Positives, 2188 True Negatives, 85 False Positives, and 69 False Negatives.

Decision Tree exhibits competitive performance with 1312 True Positives, 2054 True Negatives, 219 False Positives, and 145 False Negatives.

Moreover, Logistic Regression performs well with 1395 True Positives, 2201 True Negatives, 62 False Positives, and 72 False Negatives.

Finally, CatBoost shows 1376 True Positives, 2153 True Negatives, 81 False Positives, and 120 False Negatives.

The results of this study demonstrate the effectiveness of the proposed system across various performance metrics. Thus, it provides valuable insights into the strengths of the chosen classifiers and areas where they excel in predictive modeling tasks. Based on the results, this study recommends a trade-off between precision and recall, alongside specific customer requirements, to adopt the Anti-Phishing Tool classifier model. Table 1 summarizes these results

Table 1. Models Evaluation Metrics

ALGORITHM	ACCURACY	PRECISION	RECALL	F1-SCORE
SVM	0.97	0.97	0.97	0.97
RF	0.96	0.96	0.96	0.96
DT	0.9	0.9	0.9	0.9
LR	0.96	0.96	0.96	0.96
CAT BOOST	0.95	0.95	0.95	0.92

5. Conclusions

In conclusion, the study underscores the pivotal role of AI, specifically ML and DL, in strengthening cybersecurity measures, particularly against phishing attacks. The evaluated models demonstrate commendable performance, with SVM emerging as a robust choice for accurate email classification. While the research body extensively discusses the capabilities of various algorithms, The SVM algorithm exhibited an accuracy, precision, recall, and F1-score of 0.97, showcasing its effectiveness in countering phishing threats. These findings emphasize the necessity for thorough and accurate reporting in research conclusions to ensure a comprehensive understanding of the contributions made to the field of cybersecurity. As the cybersecurity landscape evolves, continuous research and innovation remain crucial for developing effective anti-phishing technologies to counteract the ever-evolving nature of cyber threats.

References

[1] P. Verma, A. Goyal, and Y. Gigras, "E-mail phishing: text classification using natural language processing," *Comput. Sci. Inf. Technol.*, vol. 1, no. 1, pp. 1–12, 2020, doi: 10.11591/csit.v1i1.p1-12.

[2] Y. A. Alsariera, V. E. Adeyemo, A. O. Balogun, and A. K. Alazzawi, "AI Meta-Learners and Extra-Trees Algorithm for the Detection of Phishing Websites," *IEEE Access*, vol. 8, pp. 142532–142542, 2020, doi: 10.1109/ACCESS.2020.3013699.

[3] Y. S. Murti and P. Naveen, "Machine Learning Algorithms for Phishing E-mail Detection," *J. Logist. Informatics Serv. Sci.*, vol. 10, no. 2, pp. 249–261, 2023, doi: 10.33168/JLISS.2023.0217.

[4] N. A. Unnithan, N. B. Harikrishnan, S. Akarsh, R. Vinayakumar, and K. P. Soman, "Machine learning based phishing E-mail detection Security-CEN@Amrita," *CEUR Workshop Proc.*, vol. 2124, no. March, pp. 64–68, 2018.

[5] H. F. Atlam and O. Oluwatimilehin, "Business E-mail Compromise Phishing Detection Based on Machine Learning: A Systematic Literature Review," *Electron.*, vol. 12, no. 1, pp. 1–28, 2023, doi: 10.3390/electronics12010042.

[6] Z. Yang, C. Qiao, W. Kan, and J. Qiu, "Phishing E-mail Detection Based on Hybrid Features," *IOP Conf. Ser. Earth Environ. Sci.*, vol. 252, no. 4, 2019, doi: 10.1088/1755-1315/252/4/042051.

[7] Z. Alshingiti, R. Alaqel, J. Al-Muhtadi, Q. E. U. Haq, K. Saleem, and M. H. Faheem, "A Deep Learning-Based Phishing Detection System Using CNN, LSTM, and LSTM-CNN," *Electron.*, vol. 12, no. 1, pp. 1–18, 2023, doi: 10.3390/electronics12010232.

[8] S. Atawneh and H. Aljehani, "Phishing E-mail Detection Model Using Deep Learning," *Electron.*, vol. 12, no. 20, 2023, doi: 10.3390/electronics12204261.

[9] K. Evans *et al.*, "RAIDER: Reinforcement-Aided Spear Phishing Detector," *Lect. Notes Comput. Sci.*, vol. 13787, no. 1, pp. 23–50, 2022, doi: 10.1007/978-3-031-23020-2_2.

[10] S. Bagui, D. Nandi, S. Bagui, and R. J. White, "Machine Learning and Deep Learning for Phishing E-mail Classification using One-Hot Encoding," *J. Comput. Sci.*, vol. 17, no. 7, pp. 610–623, 2021, doi: 10.3844/jcssp.2021.610.623.

[11] Y. Fang, C. Zhang, C. Huang, L. Liu, and Y. Yang, "Phishing E-mail Detection Using Improved RCNN Model With Multilevel Vectors and Attention Mechanism," *IEEE Access*, vol. 7, pp. 56329–56340, 2019, doi: 10.1109/ACCESS.2019.2913705.

[12] S. Rawal, B. Rawal, A. Shaheen, and S. Malik, "Phishing Detection in E-mails using Machine Learning," *Int. J. Appl. Inf. Syst.*, vol. 12, no. 7, pp. 21–24, 2017.

[13] S. S, R. Thomas, and E. Shyni C, "Spam E-mail Detection using Structural Features," *Int. J. Comput. Appl.*, vol. 89, no. 3, pp. 38–41, 2014, doi: 10.5120/15485-4265.

[14] A. Yasin and A. Abuhasan, "An Intelligent Classification Model for Phishing E-mail Detection," *Int. J. Netw. Secur. Its Appl.*, vol. 8, no. 4, pp. 55–72, 2016, doi: 10.5121/ijnsa.2016.8405.

[15] X. Zhang, Y. Zeng, X.-B. Jin, Z.-W. Yan, and G.-G. Geng, "Boosting the phishing detection performance by semantic analysis," in *Proc. IEEE Int. Conf. Big Data*, 2017, doi: 10.1109/BigData.2017.8258030.

[16] R. Basnet, S. Mukkamala, and A. H. Sung, "Detection of phishing attacks: A machine learning approach," *Stud. Fuzziness Soft Comput.*, vol. 226, pp. 373–383, 2008, doi: 10.1007/978-3-540-77465-5_19.

[17] A. Bergholz, G. Paaß, F. Reichartz, S. Strobel, and J. H. Chang, "Improved phishing detection using model-based features," in *Proc. 5th Conf. E-mail Anti-Spam (CEAS)*, 2008.

[18] R. Eckhardt and S. Bagui, "Convolutional Neural Networks and Long Short Term Memory for Phishing E-mail Classification," *Int. J. Comput. Sci. Inf. Secur.*, vol. 19, no. 5, pp. 27–35, 2021.

[19] H. Syahputra and A. Wibowo, "Comparison of Support Vector Machine (SVM) and Random Forest Algorithm for Detection of Negative Content on Websites," *J. Ilm. Tek. Elektro Komput. dan Inform.*, vol. 9, no. 1, pp. 165–173, 2023, doi: 10.26555/jiteki.v9i1.25861.

[20] G. Blandini *et al.*, "A Random Forest approach to quality-checking automatic snow-depth sensor measurements," 2023. [Online]. Available: https://doi.org/10.5194/egusphere-2023-656

[21] Y. Liu and S. Yang, "Application of Decision Tree-Based Classification Algorithm on Content Marketing," *J. Math.*, vol. 2022, 2022, doi: 10.1155/2022/6469054.

[22] J. C. Stoltzfus, "Logistic regression: A brief primer," *Acad. Emerg. Med.*, vol. 18, no. 10, pp. 1099–1104, 2011, doi: 10.1111/j.1553-2712.2011.01185.x.

[23] J. T. Hancock and T. M. Khoshgoftaar, "CatBoost for big data: an interdisciplinary review," *J. Big Data*, vol. 7, no. 1, 2020, doi: 10.1186/s40537-020-00369-8.

[24] Ž. Vujović, "Classification Model Evaluation Metrics," *Int. J. Adv. Comput. Sci. Appl.*, vol. 12, no. 6, pp. 599–606, 2021, doi: 10.14569/IJACSA.2021.0120670.