



# Advanced Phishing Detection: Leveraging t-SNE Feature Extraction and Machine Learning on a Comprehensive URL Dataset

Taha Etem<sup>1</sup> , Mustafa Teke<sup>2</sup> 

<sup>1</sup>Çankırı Karatekin University, Computer Engineering Department, Çankırı, Türkiye

<sup>2</sup>Çankırı Karatekin University, Department of Electrical-Electronics Engineering, Çankırı, Türkiye

**Corresponding author** : Taha Etem

**E-mail** : tahaetem@karatekin.edu.tr

## ABSTRACT

Phishing attacks continue to pose a major challenge in today's digital world; thus, sophisticated detection techniques are required to address constantly changing tactics. In this paper, we have proposed an innovative method to identify phishing attempts using the extensive PhiUSIIL dataset. The proposed dataset comprises 134,850 legitimate URLs and 100,945 phishing URLs, providing a robust foundation for analysis. We applied the t-SNE technique for feature extraction, condensing the original 51 features into only 2, while preserving high detection accuracy. We evaluated several machine learning algorithms on both full and reduced datasets, including Logistic Regression, Naive Bayes, k-Nearest Neighbors (kNN), Decision Trees, and Random Forest. The Decision Tree algorithm showed the best performance on the original dataset, achieving 99.7% accuracy. Interestingly, the proposed kNN demonstrated remarkable results on feature-extracted data, achieving 99.2% accuracy. We observed significant improvements in Logistic Regression and Random Forest performance when using the feature-extracted dataset. The proposed method offers substantial benefits in terms of computational efficiency. The feature-extracted dataset requires less processing power; thus, it is well-suited for systems with limited resources. These findings pave the way for developing more powerful and flexible phishing detection systems that can identify and neutralize emerging threats in real-time scenarios.

**Keywords:** Machine learning, cybersecurity, feature extraction, data mining

**Submitted** : 24.07.2024

**Revision Requested** : 16.09.2024

**Last Revision Received** : 19.09.2024

**Accepted** : 11.12.2024

**Published Online** : 13.12.2024



This article is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License (CC BY-NC 4.0)

## 1. INTRODUCTION

Phishing attacks are among the most common and harmful types of cybercrime. They pose significant risks to both individuals and organizations. These attacks usually trick victims into giving away sensitive information such as passwords, credit card numbers, and other personal data, through deceptive emails, websites, or messages. Due to the increasing sophistication and frequency of phishing attacks, effective and adaptive detection mechanisms are required. Traditional detection methods that rely on rule-based systems and block lists often fail to recognize new and evolving phishing techniques (Garera, Provov, Chew, & Rubin, 2007).

Recently, machine learning (ML) and deep learning (DL) have shown significant potential in improving phishing detection and prevention (Alhudhaif, Almaslukh, Aseeri, Guler & Polat, 2023; Buyrukoğlu & Savaş, 2023). These advanced computational techniques use large datasets to identify patterns and anomalies indicative of phishing activities. ML models, such as support vector machines (SVM) and random forests (RF), have been widely used to classify emails and URLs as either phishing or legitimate (Bergholz et al., 2010). More recently, DL models, particularly neural networks, have improved detection accuracy by capturing complex patterns in data that traditional ML models may miss (Adebowale, Lwin & Hossain, 2019; Türk, Lüy & Barışçı, 2020). One major advantage of ML and DL in phishing detection is their ability to generalize from training data to identify previously unseen phishing attempts. This ability is crucial given the constantly changing nature of phishing tactics. Techniques such as convolutional neural networks (CNN) and recurrent neural networks (RNN) have been applied to various aspects of phishing detection, including email content analysis, URL feature extraction, and web page layout analysis (Aburrous, Hossain, Dahal & Thabtah, 2010). However, applying ML and DL techniques to phishing detection faces several challenges (Etem & Teke, 2024). Issues like data imbalance, feature selection, model interpretability, and computational resource requirements must be carefully addressed to develop effective and efficient detection systems. In addition, adversaries continuously evolve tactics to evade detection, necessitating ongoing adaptation and refinement of these models (Jain & Gupta, 2022).

In a paper, Researchers provide a comprehensive overview of current state-of-the-art machine learning and deep learning phishing detection techniques. They discussed various techniques, highlighted their strengths and weaknesses, and explored future research directions in this critical area of cybersecurity using extreme learning machines (ELM) to detect phishing (Yang et al., 2021). Another review paper analyzed various ML and DL techniques used in phishing detection, focusing on the importance of combining multiple features and algorithms to enhance detection accuracy (Divakaran & Oest, 2022). A previous study compared the effectiveness of different ML algorithms in detecting phishing attacks, focusing on models such as SVM, RF, and neural networks, and evaluated their performance in terms of accuracy, precision, and recall (Jishnu & Arthi, 2023). A systematic review explores the use of DL techniques for phishing email detection, examining various models, including CNNs and RNNs, and their effectiveness in identifying phishing emails (Thakur, Ali, Obaidat & Kamruzzaman, 2023). Another study presented a model to detect phishing attacks using ML algorithms like RF and decision trees, emphasizing the importance of feature selection and engineering in terms of improving detection accuracy (Alam et al., 2020). In another study, the authors proposed a novel approach to phishing website detection using a combination of ML and DL models, highlighting the improvements in detection accuracy achieved through this multilayered approach (Bibi et al., 2024). Another study investigated the application of sequential DL models like Multi-Head Attention and Temporal Convolutional Networks in detecting phishing websites, and evaluated their performance in terms of accuracy and efficiency (Gopali, Namin, Abri & Jones, 2024). Another study proposed a DL model for phishing email detection, which was trained and tested on a comprehensive dataset, and demonstrated high accuracy in identifying phishing emails and discussed the implications of using DL for real-time phishing detection (Atawneh & Aljehani, 2023).

Datasets play a crucial role in training ML models to identify and counter sophisticated phishing attempts, making them a notable contribution to the field of cybersecurity. The PhiUSIIL dataset includes several data types that are essential for effective phishing detection, such as numerical, categorical, and text data (Prasad & Chandra, 2024). Numerical features include metrics such as URL lengths and email attribute frequencies, and categorical data include domain types and keyword presence. In addition, the dataset contains unstructured text data from emails and web pages, which require specialized techniques to detect phishing patterns. The diversity of data types allows the development of robust ML models that can accurately distinguish between legitimate and phishing communications. The PhiUSIIL dataset was used to train and evaluate various ML models, including SVMs, RFs, and Neural Networks, to enhance phishing detection capabilities. Security companies and software developers use this dataset to develop real-time phishing detection systems that can analyze incoming web traffic and emails to block phishing URLs before they reach end-users. The proposed dataset is invaluable for academic and industrial research, providing a benchmark for new algorithms and facilitating feature engineering and model optimization studies. Despite its benefits, the use of ML in

phishing detection faces significant challenges. The evolving tactics of cybercriminals, the need for high-quality and diverse datasets, and the requirement for substantial computational resources are major hurdles. Phishing detection systems must constantly adapt to new threats and ensure compliance with relevant privacy and ethical standards. The proposed PhiUSIIL dataset represents a significant step forward in the phishing fight, and it provides a powerful tool to develop more effective cybersecurity measures.

In examining all of these studies, it is evident that the most critical feature of phishing attacks is their ability to deceive people using constantly updated methods. Consequently, phishing attacks attempt to avoid detection by matching features in old datasets to fake websites. Continuous development of systems to detect phishing attacks and increase success rates is particularly important. To this end, we designed a phishing detection system based on the PhiUSIIL dataset. To achieve the best results and design a fast lightweight system, feature extraction was performed using the proven t-SNE method (Bibal, Delchevalerie & Frénay, 2023), and system evaluation was performed using different ML algorithms. The results show that the proposed method can play a significant role in detecting current phishing attacks.

## 2. PhiUSIIL PHISHING DATASET

Phishing involves creating unauthorized replicas of legitimate websites and emails, typically from financial institutions, to deceive individuals into divulging confidential information. These fraudulent communications often use legitimate company logos and slogans to appear credible, exploiting the HTML structure that allows easy copying of images or entire websites (Prasad & Chandra, 2024).

The PhiUSIIL dataset includes various data points that are essential for training and testing ML models to identify phishing attempts. This dataset includes distinct types of data such as numerical, categorical, and text data, each serving specific purposes in the analysis. Numerical data in the PhiUSIIL dataset include quantitative metrics such as the frequency of specific email attributes, URL length, and other measurable factors that could indicate phishing. These numerical features help quantify the critical aspects of detection algorithms. Categorical data include distinct categories or classes such as the type of domain used, the presence of certain keywords in the email content, and the classification of email sources as legitimate or suspicious. This categorical information is vital for creating classification models that distinguish between phishing and nonphishing activities. The dataset also includes unstructured text data, including email and web page contents, which require specialized extraction and analysis techniques. Text analysis helps identify common phrases or patterns used in phishing attempts, thereby enhancing the model's ability to detect subtle cues that might otherwise be overlooked.

The PhiUSIIL dataset is important in data-driven decision-making in cybersecurity. By training ML models on this dataset, researchers can build more robust phishing detection systems. The dataset's diverse range of features allows for comprehensive analysis and helps transform raw data into meaningful insights, which ultimately contributes to the more effective prevention of phishing attacks. The PhiUSIIL Phishing URL Dataset is an extensive collection of 134,850 legitimate URLs and 100,945 phishing URLs; thus, it is a substantial resource for developing and testing phishing detection algorithms. This dataset was meticulously curated by analyzing the source code of webpages and URLs to extract features such as CharContinuationRate, URLTitleMatchScore, URLCharProb, and TLDLegitimateProb. These features are essential in distinguishing legitimate and phishing URLs, and they offer numerous applications in research and practical fields.

## 3. MATERIALS AND METHODS

Phishing detection has become an essential component of cybersecurity strategies due to the evolving tactics employed by malicious actors, which deceive users and steal sensitive information. Traditional detection methods often fall short in countering sophisticated attacks, which necessitates advanced ML solutions. ML algorithms like Logistic Regression, Naïve Bayes, K-Nearest Neighbors, Decision Tree, and random forest, are used for phishing detection (Alsaç, Yenisey, Ganiz, Dagtekin & Ulusinan, 2023; Doğruel & Soner Kara, 2023; Efeoğlu, 2022; Tülay, 2023; Yaman & Tuncer, 2023).

Feature extraction involves identifying and extracting relevant features from raw data for machine learning and data analysis (Güler & Yücedağ, 2022). These features are then used to create an informative dataset for tasks such as classification, prediction, and clustering. The goal is to reduce data complexity or dimensionality while retaining as much relevant information as possible, thereby enhancing the performance and efficiency of machine learning algorithms. This process may include the creation of new features and data manipulation to separate meaningful and irrelevant features (Jiang, Shi, Liang & Liu, 2024).

t-SNE, or t-distributed Stochastic Neighbor Embedding, is a potent dimensionality reduction technique used for visualizing high-dimensional data. Developed by Laurens van der Maaten and Geoffrey Hinton in 2008, t-SNE has become a popular machine learning and data science tool for feature extraction and data visualization. The method functions by converting similarities between data points into joint probabilities and aims to minimize the Kullback-Leibler divergence between the joint probabilities of low-dimensional embedding and high-dimensional data. Unlike linear dimensionality reduction methods such as PCA, t-SNE is particularly adept at preserving local structures within the data, making it excellent for uncovering clusters and patterns that might be hidden in higher dimensions (Bibal et al., 2023). Pairwise similarities in high-dimensional space and Kullback-Leibler divergences can be found in Equation 1.

$$C = \sum_i KL(P_i || Q_i) = \sum_i \sum_j p_{i/j} \log_2 \frac{p_{i/j}}{q_{i/j}} \quad (1)$$

In the sample space, P represents the conditional probability distribution, and in the latent space, Qi represents the conditional probability distribution. In t-SNE, entropy is used to construct a cost function (Kullback-Leibler divergence) that measures the difference between the probability distributions in high-dimensional space (P) and low-dimensional space (Q). The proposed algorithm attempts to minimize this difference by effectively preserving the local structure of the data in lower-dimensional space. The units can be described as "bits per data point" or "bits per pairwise similarity." Pairwise similarities in low-dimensional space is shown in Equation 2.

$$Perp(P_i) = 2^{H(P_i)} \quad (2)$$

$$H(x) = -\sum(p_i) * \log(p_i) \quad (3)$$

Here, H introduces the Shannon entropy of the calculated P in Equation 3. t-SNE's nonlinear approach allows it to capture complex relationships in the data, often resulting in more intuitive and interpretable visualizations than linear methods. However, t-SNEs focus on preserving local structures, which means that they may not always maintain the global structure or distances between widely separated clusters. The output of the dataset after applying t-SNE is shown in Figure 1.

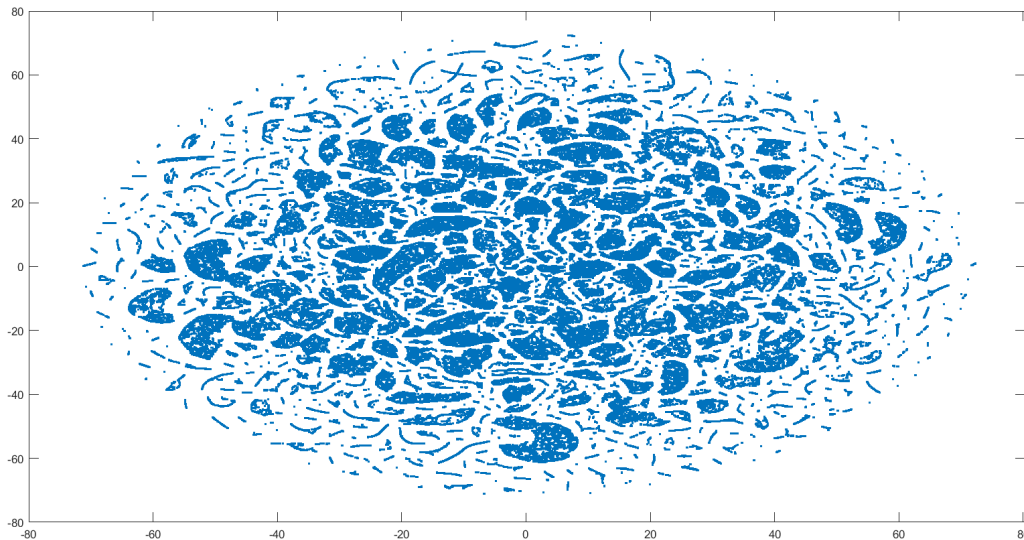


Figure 1. Outlook of Dataset after t-SNE

After all steps were applied to the MATLAB application, the dataset obtained and the code containing the applied methods were shared publicly via GitHub. The files can be accessed via the references ('GitHub - judger90/phishing\_detection\_tsne', n. d.).

#### 4. RESULTS AND DISCUSSIONS

After removing the labels and text from the dataset, 51 features containing numerical values are obtained. In addition, 51 features determined by applying the t-SNE method converted into 2 features by feature extraction. After this stage,

the original dataset and the dataset to which t-SNE feature extraction was applied are evaluated separately with the help of machine learning methods, and the results shown in Table 1 are obtained.

**Table 1.** Classification Accuracy of the Proposed Methods

Machine learning	Original Dataset	Feature Extracted Dataset
Logistic Regression	% 42,1	% 57,2
Naïve Bayes	% 99,4	% 77,2
kNN	% 99,6	% <b>99,2</b>
Decision Tree	% <b>99,7</b>	% 83,0
Random Forest	% 57,2	% 81,0

As seen in the table, the best results were obtained using the kNN algorithm for the t-SNE feature extraction method and the decision tree algorithm for the original dataset. Other metrics, Precision, Recall, F1-Score and AUC for the best methods are shown in Table 2.

**Table 2.** Performance Metrics of the Proposed Methods

Machine learning	kNN: Original Dataset	kNN-Feature Extracted Dataset	Decision Tree: Original Dataset	Decision Tree-Feature Extracted Dataset
Accuracy	% 99,6	% 99,2	% 99,7	% 83,0
Precision	% 99,6	% 99,4	% 99,7	% 83,0
Recall	% 99,6	% 99,3	% 99,2	% 82,9
F1-Score	% 99,5	% 99,2	% 98,6	% 82,8
AUC	% 99,9	% 99,9	% 99,6	% 86,1

Accuracy measures the overall model accuracy. All methods performed very well. Precision indicates the proportion of positive identifications that were correct. Results closely mirror the accuracy, with kNN and Decision Tree on Original Dataset performing the best (99.6% and 99.7%). Recall represents the proportion of actual positives that were identified correctly. kNN methods on both datasets demonstrated high recall (99.6% and 99.3%). The F1-score is the harmonic mean of precision and recall, providing a single score that balances both metrics. kNN methods on both datasets maintained high F1-scores (99.5% and 99.2%). AUC (Area Under the ROC Curve) measures the model's ability to distinguish between classes. kNN methods on both datasets demonstrated near-perfect AUC (99.9%) and Decision Tree on Original Dataset is also excellent (99.6%).

Generally, the proposed kNN performed consistently well across both datasets. The decision tree method demonstrated a significant drop in performance when used with the Feature Extracted Dataset, that essential information may have been lost during the feature extraction process. The high AUC scores for all methods indicate excellent ability to distinguish between classes even when the other metrics are slightly lower.

**Table 3.** Classification Accuracy of the Proposed Methods

Machine learning	Prediction Speed (predict/second)	Training Time (seconds)	Model Size (bytes)	Selected Features
<b>Logistic Regression: Original Dataset</b>	486172,250	29,049	18217	51/51
<b>Logistic Regression: Feature Extracted Dataset</b>	2192167,345	40,134	11028	2/2
<b>Naïve Bayes: Original Dataset</b>	668,875	1802,257	5019572	51/51
<b>Naïve Bayes– Feature Extracted Dataset</b>	230,602	4335,852	15103075	2/2
<b>kNN: Original Dataset</b>	145,829	6873,399	101873948	51/51
<b>kNN–Feature Extracted Dataset</b>	83355,533	43,985	14090229	2/2
<b>Decision Tree: Original Dataset</b>	584450,559	33,817	10839	51/51
<b>Decision Tree– Feature Extracted Dataset</b>	2293163,544	9,204	8002	2/2
<b>Random Forest: Original Dataset</b>	238485,666	6953,444	261049	51/51
<b>Random Forest– Feature Extracted Dataset</b>	107519,752	356,531	9604	2/2

The prediction speed is shown in Table 3. The prediction speed indicates predictions can be made per second. Therefore, higher values indicate more efficient system designs. The feature-extracted dataset generally exhibited much higher prediction speeds than the original dataset. The logistic regression and decision tree models on feature extracted datasets demonstrated the highest prediction speeds (over two million predictions/second). kNN on the original dataset was the slowest (145,829 predictions/second).

The training time metric expresses the total time required for training in seconds. The feature extracted dataset often (but not always) results in faster training times. The decision tree on the feature extracted dataset had the fastest training time (9.204 seconds). Naïve Bayes and kNN methods on the original dataset had notably longer training times.

The model size metric expresses the total space held by the model in memory in bytes. Feature-extracted datasets generally result in smaller model sizes, with some exceptions. The logistic regression and decision tree models are consistently small. kNN on the original dataset had the largest model size (101,873,948 bytes).

The selected feature column shows the number of features used in the training process. All Original dataset models use all 51 available features, while feature-extracted dataset models use only 2 features, indicating significant dimensionality reduction.

Feature extraction significantly improves efficiency across most metrics:

1. the proposed t-SNE algorithm drastically increases the prediction speed.
2. t-SNE algorithm generally reduces the training time (except for Naïve Bayes).
3. t-SNE algorithms usually decrease the model size (except for Naïve Bayes).
4. t-SNE algorithm achieves high efficiency with only 2 features instead of 51 features.

The feature extraction process is highly effective in this case, as it captures the most valuable information in only two features.

The ROC curves obtained by these algorithms are shown in Figure 2.



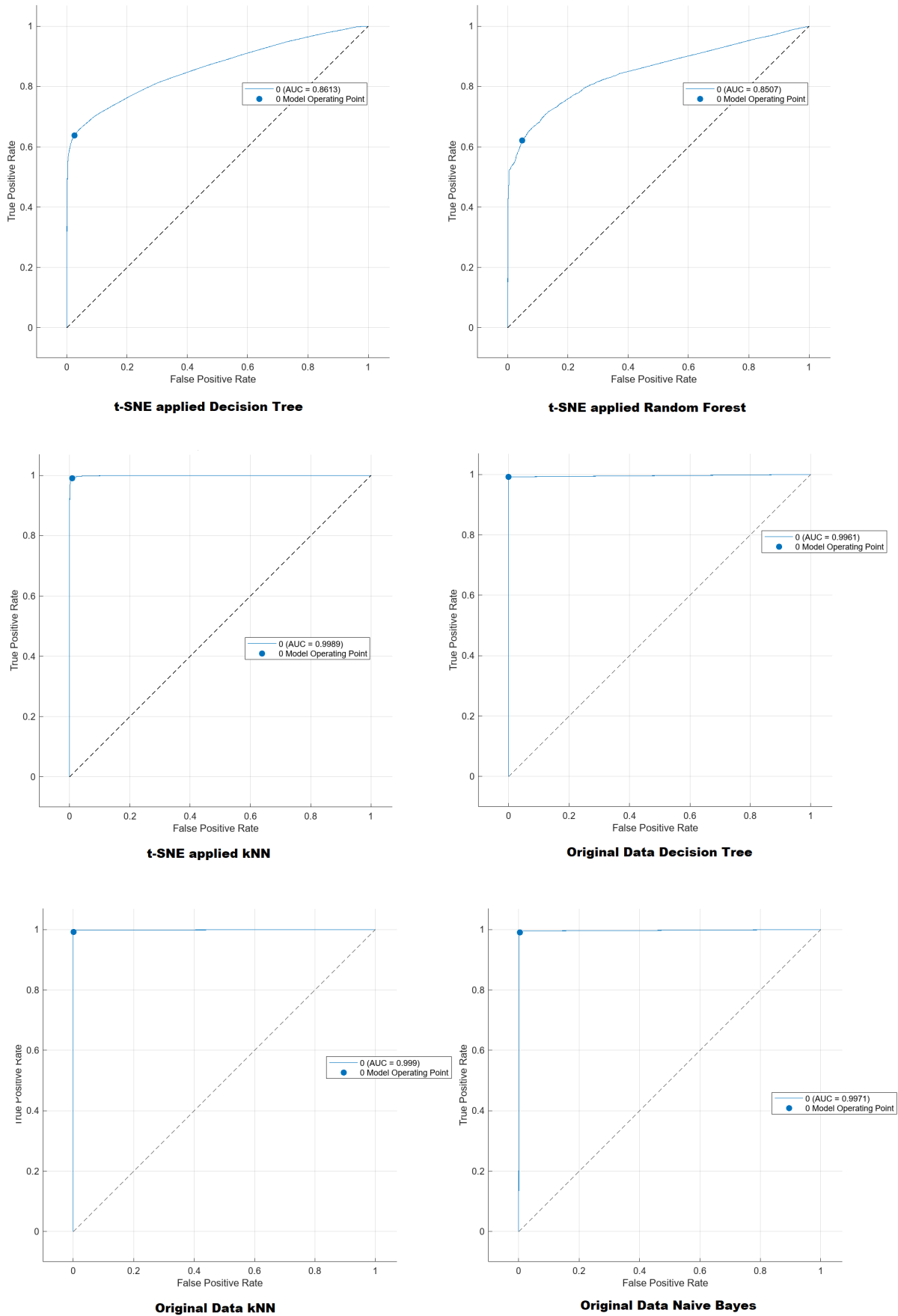


Figure 2. ROC Curves of the Machine Learning Algorithms

In general, the best result obtained by the decision tree in the original dataset. However, in the feature extraction method, the kNN algorithm again demonstrated high performance 99.2%. Another remarkable result was obtained with feature extraction (Table 1, where the success of the Logistic Regression classifier and random forest algorithms significantly increased. The fact that the algorithms use 51 features in the original dataset and only 2 features in the feature extraction dataset indicates a reduction in the amount of memory and training time.

## 5. CONCLUSIONS

The increasing number of phishing attacks each day supports the continuous conduct of up-to-date studies. In this study, it is aimed to detect websites containing phishing attacks using the PhiUSIIL dataset, which is an up-to-date phishing detection dataset. The dataset has demonstrated high performance in machine learning algorithms both in its original form and with feature extraction using the t-SNE method. The most important advantage of applying inference with the t-SNE method is that the method itself has a lightweight structure, operates fast, and offers high performance in machine learning algorithms with the help of only 2 features obtained. With these characteristics, the proposed method can be used successfully to detect phishing attacks and can also be applied to structures with low system requirements.

**Peer Review:** Externally peer-reviewed.

**Author Contributions:** Conception/Design of Study- T.E.; Data Acquisition- T.E.; Data Analysis/Interpretation- T.E., M.T.; Drafting Manuscript- T.E.; Critical Revision of Manuscript- M.T.; Final Approval and Accountability- T.E., M.T.; Technical or Material Support – M.T.; Supervision- M.T.

**Conflict of Interest:** The authors have no conflict of interest to declare.

**Grant Support:** The authors declared that this study has received no financial support.

## ORCID IDs of the authors

Taha Etem 0000-0003-1419-5008

Mustafa Teke 0000-0002-7262-4918

## REFERENCES

- Aburrous, M., Hossain, M. A., Dahal, K., & Thabtah, F. (2010). Intelligent phishing detection system for e-banking using fuzzy data mining. *Expert Systems with Applications*, 37(12), 7913–7921. doi:10.1016/J.ESWA.2010.04.044
- Adebowale, M. A., Lwin, K. T., & Hossain, M. A. (2019). Deep learning with convolutional neural network and long short-term memory for phishing detection. *2019 13th International Conference on Software, Knowledge, Information Management and Applications, SKIMA 2019*. doi:10.1109/SKIMA47702.2019.8982427
- Alam, M. N., Sarma, D., Lima, F. F., Saha, I., Ulfath, R. E., & Hossain, S. (2020). Phishing Attacks Detection using Machine Learning Approach. *2020 Third International Conference on Smart Systems and Inventive Technology (ICSSIT)*, 1173–1179. doi:10.1109/ICSSIT48917.2020.9214225
- Alhudaif, A., Almaslukh, B., Aseeri, A. O., Guler, O., & Polat, K. (2023). A novel nonlinear automated multi-class skin lesion detection system using soft-attention based convolutional neural networks. *Chaos, Solitons & Fractals*, 170, 113409. doi:10.1016/J.CHAOS.2023.113409
- Alsaç, A., Yenisey, M. M., Ganiz, M., Dagtekin, M., & Ulusunan, T. (2023). The Efficiency of Regularization Method on Model Success in Issue Type Prediction Problem. *Acta Infologica*, 7(2), 360–383. doi:10.26650/ACIN.1394019
- Atawneh, S., & Aljehani, H. (2023). Phishing Email Detection Model Using Deep Learning. *Electronics 2023, Vol. 12*, Page 4261, 12(20), 4261. doi:10.3390/ELECTRONICS12204261
- Bergholz, A., De Beer, J., Glahn, S., Moens, M. F., Paaß, G., & Strobel, S. (2010). New filtering approaches for phishing email. *Journal of Computer Security*, 18(1), 7–35. doi:10.3233/JCS-2010-0371
- Bibal, A., Delchevalerie, V., & Frénay, B. (2023). DT-SNE: t-SNE discrete visualizations as decision tree structures. *Neurocomputing*, 529, 101–112. doi:10.1016/J.NEUCOM.2023.01.073
- Bibi, H., Shah, S. R., Baig, M. M., Sharif, M. I., Mehmood, M., Akhtar, Z., & Siddique, K. (2024). Phishing Website Detection Using Improved Multilayered Convolutional Neural Networks. *Journal of Computer Science*, 20(9), 1069–1079. doi:10.3844/JCSSP.2024.1069.1079
- Buyrukoğlu, S., & Savaş, S. (2023). Stacked-Based Ensemble Machine Learning Model for Positioning Footballer. *Arabian Journal for Science and Engineering*, 48(2), 1371–1383. doi:10.1007/s13369-022-06857-8
- Divakaran, D. M., & Oest, A. (2022). Phishing Detection Leveraging Machine Learning and Deep Learning: A Review. *IEEE Security and Privacy*, 20(5), 86–95. doi:10.1109/MSEC.2022.3175225
- Doğruel, M., & Soner Kara, S. (2023). Determining the Happiness Class of Countries with Tree-Based Algorithms in Machine Learning. *Acta Infologica*, 7(2), 0–0. doi:10.26650/ACIN.1251650



- Efeoğlu, E. (2022). Kablosuz Sinyal Gücünü Kullanarak İç Mekan Kullanıcı Lokalizasyonu için Karar Ağacı Algoritmalarının Karşılaştırılması. *Acta Infologica*, 0(0), 0–0. doi:10.26650/ACIN.1076352
- Etem, T., & Teke, M. (2024). Enhanced deep learning based decision support system for kidney tumour detection. *BenchCouncil Transactions on Benchmarks, Standards and Evaluations*, 4(2), 100174. doi:10.1016/J.TBENCH.2024.100174
- Garera, S., Provos, N., Chew, M., & Rubin, A. D. (2007). A framework for detection and measurement of phishing attacks. *WORM'07 - Proceedings of the 2007 ACM Workshop on Recurring Malcode*, 1–8. doi:10.1145/1314389.1314391
- GitHub - judger90/phishing\_detection\_tsne. (n.d.). Retrieved 19 September 2024, from [https://github.com/judger90/phishing\\_detection\\_tsne](https://github.com/judger90/phishing_detection_tsne)
- Gopali, S., Namin, A. S., Abri, F., & Jones, K. S. (2024). *The Performance of Sequential Deep Learning Models in Detecting Phishing Websites Using Contextual Features of URLs*. In *SAC '24: Proceedings of the 39th ACM/SIGAPP Symposium on Applied Computing* (pp. 1064–1066). Association for Computing Machinery (ACM). doi:10.1145/3605098.3636164
- Güler, O., & Yücedağ, İ. (2022). Hand Gesture Recognition from 2D Images by Using Convolutional Capsule Neural Networks. *Arabian Journal for Science and Engineering*, 47(2), 1211–1225. doi:10.1007/S13369-021-05867-2/TABLES/8
- Jain, A. K., & Gupta, B. B. (2022). A survey of phishing attack techniques, defence mechanisms and open research challenges. *Enterprise Information Systems*, 16(4), 527–565. doi:10.1080/17517575.2021.1896786
- Jiang, D., Shi, X., Liang, Y., & Liu, H. (2024). Feature extraction technique based on Shapley value method and improved mRMR algorithm. *Measurement*, 237, 115190. doi:10.1016/J.MEASUREMENT.2024.115190
- Jishnu, K. S., & Arthi, B. (2023). Review of the effectiveness of machine learning based phishing prevention systems. *AIP Conference Proceedings*, 2917(1). doi:10.1063/5.0175593/2919402
- Prasad, A., & Chandra, S. (2024). PhiUSIIL: A diverse security profile empowered phishing URL detection framework based on similarity index and incremental learning. *Computers & Security*, 136, 103545. doi:10.1016/J.COSE.2023.103545
- Thakur, K., Ali, M. L., Obaidat, M. A., & Kamruzzaman, A. (2023). A Systematic Review on Deep-Learning-Based Phishing Email Detection. *Electronics* 2023, Vol. 12, Page 4545, 12(21), 4545. doi:10.3390/ELECTRONICS12214545
- Tülay, E. (2023). Detection of Orienting Response to Novel Sounds in Healthy Elderly Subjects: A Machine Learning Approach Using EEG Features. *Acta Infologica*, 0(0), 0–0. doi:10.26650/ACIN.1234106
- Türk, F., Lüy, M., & Barışçı, N. (2020). Kidney and Renal Tumor Segmentation Using a Hybrid V-Net-Based Model. *Mathematics* 2020, Vol. 8, Page 1772, 8(10), 1772. doi:10.3390/MATH8101772
- Yaman, O., & Tuncer, T. (2023). Plant Classification Method Using Histogram and Machine Learning for Smart Agriculture Applications. *Acta Infologica*, 0(0), 0–0. doi:10.26650/ACIN.1070261
- Yang, L., Zhang, J., Wang, X., Li, Z., Li, Z., & He, Y. (2021). An improved ELM-based and data preprocessing integrated approach for phishing detection considering comprehensive features. *Expert Systems with Applications*, 165, 113863. doi:10.1016/J.ESWA.2020.113863

### How cite this article

Etem, T., & Teke, M. (2024). Advanced Phishing Detection: Leveraging t-SNE Feature Extraction and Machine Learning on a Comprehensive URL Dataset. *Acta Infologica*, 8(2), 213-221. <https://doi.org/10.26650/acin.1521835>