Journal of Future Artificial Intelligence and Technologies

E-ISSN: 3048-3719

Research Article

Phishing Website Detection Using Bidirectional Gated Recurrent Unit Model and Feature Selection

De Rosal Ignatius Moses Setiadi [1,*], Suyud Widiono [2], Achmad Nuruddin Safriandono [3], and Setyo Budi [4]

[1] Informatics Engineering Department, Faculty of Computer Science, Dian Nuswantoro University, Indonesia; e-mail : moses@dsn.dinus.ac.id
[2] Department of Computer Engineering, Faculty of Science and Technology, University of Technology Yogyakarta, Indonesia; e-mail : suyud.w@uty.ac.id
[3] Faculty of Engineering, Sultan Fatah University, Demak, Central Java 59516, Indonesia; e-mail : udinozz@gmail.com
4 Information System Department, Faculty of Computer Science, Dian Nuswantoro University, Indonesia; e-mail : setyobudi@dsn.dinus.ac.id
* Corresponding Author : De Rosal Ignatius Moses Setiadi

Abstract: Phishing attacks continue to be a significant threat to internet users, necessitating the development of advanced detection systems. This study explores the efficacy of a Bidirectional Gated Recurrent Unit (BiGRU) model combined with feature selection techniques for detecting phishing websites. The dataset used for this research is sourced from the UCI Machine Learning Repository, specifically the Phishing Websites dataset. This approach involves cleaning and preprocessing the data, then normalizing features and employing feature selection to identify the most relevant attributes for classification. The BiGRU model, known for its ability to capture temporal dependencies in data, is then applied. To ensure robust evaluation, we utilized cross-validation, dividing the data into five folds. The experimental results are highly promising, demonstrating a Mean Accuracy, Mean Precision, Mean Recall, Mean F1 Score, and Mean AUC of 1.0. These results indicate the model's exceptional performance distinguishing between phishing and legitimate websites. This study highlights the potential of combining BiGRU models with feature selection and cross-validation to create highly accurate phishing detection systems, providing a reliable solution to enhance cybersecurity measures.

Keywords: BiGRU; Cyber attack; Cyber security; Phishing detection; Website phishing classification.

1. Introduction

Digital data security is very important in digital communication and surfing in cyberspace. As the number of internet users increases, cyber crime also increases[1]–[3]. According to reports from [4], [5], cyber intrusions increase by up to 75%. In addition, reports from the Anti-Phishing Working Group (APWG) show that phishing attacks increase yearly, with millions of users becoming victims[6], [7]. Phishing is a type of cyber attack that aims to steal sensitive user information through fake websites that imitate real websites. Despite mitigation efforts, phishing attacks remain a significant threat to internet users. Effective phishing detection is essential to protect users from cyberattacks[6], [8]–[10]. Traditional approaches that rely on blocklists and static features are no longer adequate, given the increasing sophistication of phishing methods. Therefore, a more advanced detection system is needed that can handle various forms of phishing attacks that continue to develop.

Various approaches have been developed to detect phishing, including machine learning (ML) and deep learning (DL) based techniques. Some studies use algorithms such as Random Forest(RF)[11]–[16], Decision Tree(DT)[12], [13], [15], [17], Multi Layer Perceptron (MLP)[11], Artificial Neural Network(ANN)[13], [15], [16], [18], Naïve Bayes(NB)[11], [14]–[16], [19], K-nierest neigboor (KNN)[14], [15], [17], [19], Support Vector Machine(SVM)[13], [15], [16], [18], and Logistic Regresion (LR)[14], [18]. Several DL approaches have also been carried out, such as research [20], which uses Long Short-Term Memory (LSTM). LSTM is a

Received: June, 10th 2024
Revised: July, 7th 2024
Accepted: July, 10th 2024
Published: July, 12th 2024
Curr. Ver.: July, 12th 2024

Copyright: © 2024 by the authors. Submitted for possible open access publication under the terms and conditions of the Creative Commons Attribution (CC BY SA) license (https://creativecommons.org/licenses/by-sa/4.0/)

deep learning method based on Recurrent Neural Networks (RNN)[21], [22]. Another study [23] used a convolutional neural network (CNN), also a DL model. In general, the choice between RNN and CNN depends on the nature of the data and the problem to be solved. RNNs are superior at processing sequential data and capturing temporal dependencies, making them suitable for tabular data with temporal or sequential elements. CNNs, on the other hand, are more effective at recognizing spatial patterns and can be used in situations where tabular data can be represented spatially[24].

Research [25] compares the CNN, LSTM, and LSTM-CNN methods, where CNN is superior to both. However, other research [26] comparing the LSTM, CNN, and Deep Neural Network (DNN) methods shows the superiority of LSTM. Other research also tested several DL methods, such as CNN, RNN, LSTM, and their combinations. The results were also different, namely CNN-LSTM, which was the best. This is possible due to differences in approaches and datasets. One of the newer variants of RNN is the Gated Recurrent Unit (GRU). GRU has a simpler structure and fewer parameters than LSTM, so it is faster to train and execute and tends to reduce the risk of overfitting in data classification. The study by Chung et al. [27] showed that GRU can achieve comparable or better performance than LSTM in several natural language and sequence data processing tasks. Subsequently, GRU was developed into two directions (forward and backward), which was named Bidirectional Gated Recurrent Unit (BiGRU), which was able to better capture temporal dependencies in the data [28]–[30]. This gives BiGRU the potential to work well for phishing detection.

Feature selection is an important process in improving the performance of ML and DL models. Regarding the case of phishing detection, research [11] has proven that feature selection in the Phishing Websites - UCI Machine Learning Repository dataset can increase detection accuracy. Feature selection is the process of selecting the most relevant attributes. The model can be more efficient and accurate in classification. Various feature selection techniques have been applied in previous research, showing that combining deep learning models with feature selection can provide excellent results.

In this research, we propose the use of the Bidirectional Gated Recurrent Unit (BiGRU) model combined with feature selection techniques to detect phishing websites. The dataset comes from the UCI Machine Learning Repository, specifically the Phishing Websites dataset. The research process includes data cleaning and preprocessing, feature normalization, and feature selection to identify the most relevant attributes for classification. The BiGRU model is then applied to exploit the ability to capture temporal dependencies in the data.

2. Related Works

Several studies related to phishing detection have been conducted in recent years. For instance, in the study by Ubing et al. [31] feature selection algorithms were integrated with ensemble learning methods based on majority voting to improve the accuracy of phishing website detection. This study compared various models, including Random Forest and Logistic Regression, and showed that their proposed model achieved an accuracy of up to 95.4%, outperforming previous technologies, which ranged from 70% to 92.52%.

Lakshmi et al. [32] employed supervised deep learning techniques combined with the ADAM optimization method to detect phishing websites. Their approach used advanced deep learning models to analyze web page features, improving phishing detection accuracy and robustness. They reported that their model achieved an accuracy of approximately 96%.

Alsariera et al. [33] proposed three meta-learner models based on the Forest Penalizing Attributes (ForestPA) algorithm. This method uses a weight assignment strategy to build efficient decision trees, resulting in high accuracy and low false alarm rates. The ForestPA-PWDM, Bagged-ForestPA-PWDM, and Adab-ForestPA-PWDM models achieved accuracies of 96.26%, 96.58%, and 97.4%, respectively, demonstrating superior performance compared to previous methods.

Alnemari and Alshammari [13] utilized several ML algorithms to detect phishing domains, including Random Forest, SVM, and Neural Networks. Their study focused on improving model accuracy through comprehensive feature selection and ensemble learning, achieving significant improvements in detection performance with up to 97.3% accuracy.

Shabudin et al. [11] examined the performance of two feature selection techniques, namely Feature Selection by Omitting Redundant Features (FSOR) and Feature Selection by Filtering Method (FSFM). Feature selection with FSOR and FSFM selected 22 and 11 features

from 30, respectively. They evaluated phishing detection performance using three ML techniques: RF, MLP, and NB. The experimental results showed that RF optimized with FSOR achieved the highest performance with accuracies up to 97.18% for RF, 96.51% for MLP, and 92.98% for NB, with more efficient processing times.

All the reviewed studies used the Phishing Websites dataset from the UCI Machine Learning Repository. This ensures a consistent baseline for comparing various approaches in phishing detection. While previous studies have demonstrated the effectiveness of various ML and DL models, including feature selection and ensemble learning, there is a need to explore more advanced recurrent neural network architectures to improve phishing detection. Our research proposes using a BiGRU model combined with feature selection techniques. Furthermore, this study adopts the FSOR feature selection method by Shabudin et al. [11]. This approach aims to leverage temporal dependencies in the data more effectively, potentially offering superior performance in detecting phishing websites compared to existing methods.

3. Proposed Method

This section presents the proposed method step by step. Figure 1 presents an illustration of the steps taken to detect phishing.

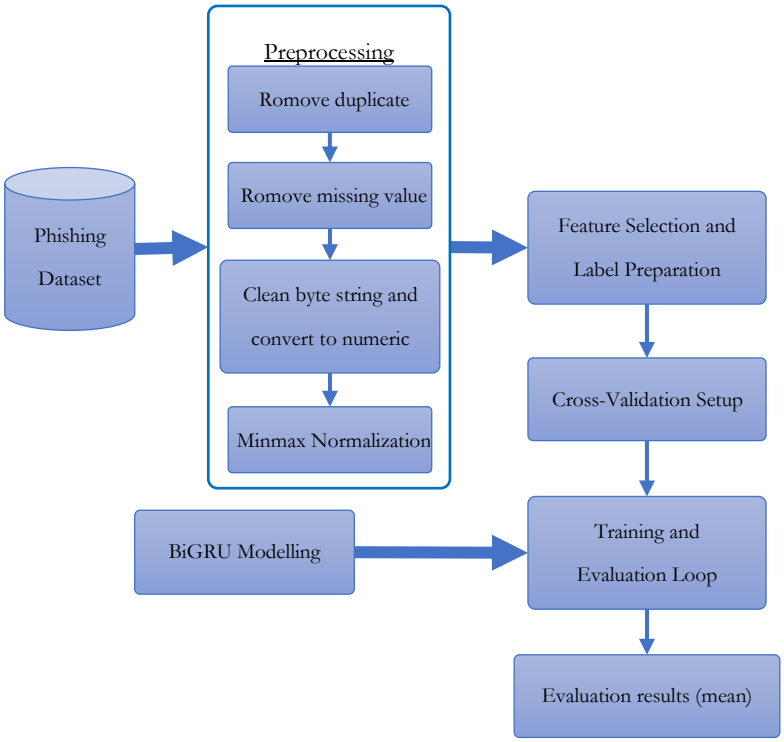

Figure 1. Proposed Method.

Based on Figure 1. the detailed stages of the proposed method are as follows:

1. Read the entire Phishing dataset, then save it in a variable of type data frame in Python.
2. Delete duplicate records at the same time as records that have missing values.
3. Delete all byte strings to ensure proper format, then convert columns to numeric format, handling any erroneous values by coercion.
4. Feature normalization is performed to scale the values of the features within a specific range, typically [0, 1][34], [35], which can help improve the performance of machine learning algorithms. In this study, the min-max scaling technique is used, which is calculated using Equation (1).

$$X_{\text{scaled}} = \frac{X - X_{\min}}{X_{\max} - X_{\min}} \quad (1)$$

where X is the original value; $X_{\min}$ is the minimum value of the feature; $X_{\max}$ is the maximum value of the feature; X_{scaled} is the scaled value.

5. Select the FSOR feature and change the label to category form. In more detail, the stages of FSOR are as follows:
   a. The FSOR method operates under the assumption that features with the same degree of accuracy and influence are redundant. Redundant features do not add significant value to the classification process and can be removed to simplify the model.
   b. The Relief Ranking Filter ranks all extracted features based on their relevance[36]–[39]. The algorithm works as follows: 1) Randomly sample an instance from the training data. 2) Identify the nearest sample from the same class (nearest hit) and the nearest sample from a different class (nearest miss). 3) Use the feature values of the nearest neighbors to update the relevance weights of the features
   c. The weight for each feature is calculated using Equation (2).

$$w = \sum_{i=1}^{n} (Pd_i - Ps_i) \tag{1}$$

   where w is the weight for each feature f; Pd is the probability of a different value of feature x of different classes cd; Ps is the probability of a different value of feature x of same classes cs.

6. Initialize BiGRU model architecture consists of two Bidirectional GRU layers followed by a Dropout layer to prevent overfitting, a Dense layer with ReLU activation, and an output Dense layer with softmax activation for classification. The model is compiled with the Adam optimizer and categorical cross-entropy loss function. For more details, see Table 1.

Table 1. Proposed BiGRU Layer Configuration.

Layer Type	Configuration Details
Input Layer	Input shape: (number of features, 1)
Bidirectional GRU	Units: 64, Return sequences: True
Bidirectional GRU	Units: 64, Return sequences: False
Dropout	Rate: 0.5
Dense	Units: 32, Activation: ReLU
Output Layer	Units: number of classes, Activation: Softmax
Optimizer	Adam, learning rate of 0.001
Loss Function	Categorical Cross-Entropy

7. The dataset is split into training and testing sets using 5-fold cross-validation in the training and evaluation loop process. Each loop uses a configuration of 25 epochs, and batch size = 32. The training data is reshaped to fit the input shape of the GRU layers. The model is then trained and evaluated in each fold. Evaluation metrics such as accuracy, precision, recall, F1 score, and AUC are calculated and presented.

4. Results and Discussion

This research uses Phishing Websites - UCI Machine Learning Repository. The dataset, primarily compiled from sources such as the PhishTank archive, MillerSmiles archive, and Google's search operators, is designed for the computer science field, focusing on classification tasks. It consists of tabular data with integer features, consisting of 11,055 and 30 important features that have proven effective in predicting phishing websites. To see all the features in more detail, see Table 2. There are two classes in the dataset, namely, 1 indicates that the website is classified as a phishing website, and -1 indicates that the website is classified as a legitimate (non-phishing) website. The class distribution of the raw dataset is presented in Figure 2 (a), then the class distribution is presented after removing duplication and missing values in Figure 2 (b).

Table 2 shows that the 30 features are complete and cover various characteristics, from URL structure and domain properties to page ranking and web traffic metrics. This richness of features helps the model learn more diverse patterns. In addition, the initial cleaning of the dataset can positively affect ML models. Because duplicate data can bias model training, causing overfitting, while missing values can disrupt the learning process if not handled properly.

Table 2. Phishing Websites UCI Dataset Features Detail.

Feature Name	Description
having_IP_Address	Whether the URL has an IP address instead of a domain name (1: Yes, -1: No)
URL_Length	Length of the URL (1: Long, 0: Medium, -1: Short)
Shortening_Service	Whether URL shortening services like bit.ly are used (1: Yes, -1: No)
having_At_Symbol	Presence of "@" symbol in the URL (1: Yes, -1: No)
double_slash_redirecting	Presence of "//" in the URL path (1: Yes, -1: No)
Prefix_Suffix	Presence of "-" in the domain part of the URL (1: Yes, -1: No)
having_Sub_Domain	Number of subdomains in the URL (1: More than one, 0: One, -1: None)
SSLfinal_State	Whether the website uses HTTPS with a valid SSL certificate (1: Yes, -1: No)
Domain_registeration_length	Length of time the domain has been registered (1: More than a year, -1: Less than a year)
Favicon	Whether the favicon is loaded from the same domain (1: Yes, -1: No)
port	Use of non-standard ports (1: Yes, -1: No)
HTTPS_token	Presence of "HTTPS" token in the URL (1: Yes, -1: No)
Request_URL	Percentage of external links in the source code of the website (1: High, -1: Low)
URL_of_Anchor	Percentage of external anchor links on the website (1: High, -1: Low)
Links_in_tags	Percentage of external links in tags (e.g., meta, script) (1: High, -1: Low)
SFH	Server Form Handler, where the form data is submitted (1: External, 0: Internal, -1: Same)
Submitting_to_email	Whether the form submits data to an email address (1: Yes, -1: No)
Abnormal_URL	Whether the URL is abnormal (1: Yes, -1: No)
Redirect	Number of redirects (1: More than one, -1: Less than one)
on_mouseover	Whether changing status bar content on mouseover (1: Yes, -1: No)
RightClick	Whether right-click is turned off on the website (1: Yes, -1: No)
popUpWindow	Whether pop-up windows are present (1: Yes, -1: No)
Iframe	Whether iframe is used on the website (1: Yes, -1: No)
age_of_domain	Age of the domain (1: More than 6 months, -1: Less than 6 months)
DNSRecord	Whether the DNS record exists (1: Yes, -1: No)
web_traffic	Web traffic rank (1: High, 0: Medium, -1: Low)
Page_Rank	Google PageRank (1: High, -1: Low)
Google_Index	Whether Google indexes the site (1: Yes, -1: No)
Links_pointing_to_page	Number of links pointing to the page (1: High, 0: Medium, -1: Low)
Statistical_report	Whether the website is reported as a phishing site (1: Yes, -1: No)

After the initial cleaning, the class distribution was also relatively more balanced. The raw dataset consists of 11,055 records. After the initial deletion, the records were reduced to 5,849, of which 2,830 were phishing web classes, and 3,019 were legitimate web classes. A relatively balanced dataset is important because it helps prevent the model from being biased towards one class, resulting in more reliable and fair predictions. In the next stage, the byte

string is also cleaned and converted to a numeric type to ensure the data is in a consistent and proper format. The next conversion to numeric step then converts this string into a numeric type, which is very important for mathematical operations and machine learning algorithms. This step also converts non-numeric values to NaN, making identifying and managing problematic data entries easier. Finally, normalization is carried out with min-max.

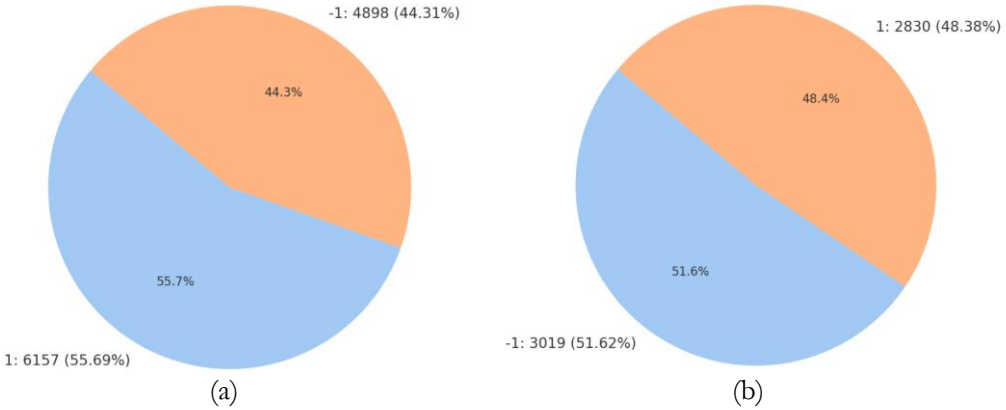

Figure 2. Class distribution dataset (**a**) before deleting duplicate and missing values records; (**b**) after deleting duplicate and missing values records.

Feature selection using FSOR selects 22 features, viz: URL_Length, Shortening_Service, having_At_Symbol, double_slash_redirecting, having_Sub_Domain, SSLfinal_State, Domain_registeration_length, Favicon, HTTPS_token, URL_of_Anchor, Links_in_tags, SFH, Submitting_to_email, RightClick, popUpWindow, DNSRecord, web_traffic, Page_Rank, Google_Index, Links_pointing_to_page, Statistical_report. Selanjutnya hasil dari proses klasifikasi menggunakan BiGRU dihitung dengan accuracy, precision, recall, F1, specificity, dan AUC, seluruh hasil disajikan pada Tabel 3.

Table 3. Classification results using 5-fold cross validation.

Method	Accuracy	Precision	Recall	F1 Score	AUC
BiGRU all feature	0.8957	0.8941	0.8906	0.8920	0.8957
BiGRU+FSFM	0.8855	0.8865	0.8757	0.8809	0.8851
BiGRU+FSOR	1.0000	1.0000	1.0000	1.0000	1.0000

In interpreting phishing detection, accuracy shows how well the model recognizes phishing and legitimate sites. However, on an imbalanced dataset, this metric is less representative. Precision indicates how accurate the model is in predicting phishing sites. High precision means few false positives (legitimate sites misidentified as phishing). Recall indicates how well the model is at detecting all phishing sites. High recall means few false negatives (phishing sites that are not detected). F1 is the balance value of precision and recall. This is suitable for imbalanced datasets[40]–[42]. Meanwhile, AUC provides a comprehensive picture of model performance[43]. Recall is a top priority for phishing detection to ensure the model detects as many sites as possible. However, the F1 Score is also important because it provides a balance between detecting phishing sites (recall) and minimizing false alarms (precision)[44], [45].

In Table 3, a comparison of the BiGRU model classification results with FSOR, FSFM, and all feature selections is presented. In FSFM the features used are Links_in_tags, Domain_registration_length, Page_Rank, SSLfinal_State, having_Sub_Domain, SFH, Submitting_to_email, Statistical_report, having_IP_Address, Google_Index, URL_Length. It can be seen from the results above that the proposed BiGRU+FSOR method produces the best performance, followed by all features and FSFM. These results show that feature selection greatly influences the prediction results.

Although perfect accuracy may raise doubts, in the context of this study, the use of k-fold cross-validation, along with effective feature selection, appears to have improved the quality of the dataset and the reliability of the model. Feature selection using FSOR has been proven to significantly improve dataset quality by eliminating irrelevant or redundant features.

This allows the model to focus on truly informative features, improving overall performance. This indicates that proper feature selection can greatly impact model performance. The use of k-fold cross-validation provides strong validation of model performance. This technique reduces the variance of results by ensuring that each data subset is used for training and testing, providing a more accurate picture of the model's generalization capabilities [46]. Apart from accuracy, other metrics such as precision, recall, F1 score, and AUC also show perfect results, indicating that this model is not only effective in detecting phishing sites but also in minimizing false positives and false negatives.

5. Comparison

Furthermore, in Table 4, a comparison is presented with previous research which used the same dataset.

Table 4. Comparison with related works.

Method	Accuracy	Precision	Recall	F1 Score	AUC
Method [11]	0.9708	-	-	-	-
Method [13]	0.973	0.97	0.982	0.976	-
Method [31]	0.954	0.935	0.959	0.947	-
Method [32]	0.96	-	-	-	-
Method [33]	0.9740	-	-	0.974	-
BiGRU+FSOR (ours)	1.0000	1.0000	1.0000	1.0000	1.0000

Table 4 presents comparison results of the BiGRU+FSOR model with several previous studies using the same dataset. Based on this table, it can be seen that the BiGRU+FSOR approach provides very superior results compared to previous methods. This method achieved perfect accuracy (100%) in all evaluation metrics, namely accuracy, precision, recall, F1 score, and AUC. These results show that using the BiGRU model combined with the FSOR feature selection technique can perform much better than other existing methods. This proves that FSOR is very effective in selecting the most relevant features for phishing detection so that the model can achieve optimal performance.

6. Conclusions

This research proves the effectiveness of the BiGRU model combined with the FSOR feature selection technique in detecting phishing websites. Experimental results show that the proposed model achieves accuracy, precision, recall, F1 score, and AUC of 100%, which indicates superior performance in distinguishing between phishing sites and legitimate sites. The FSOR feature selection is proven to increase model accuracy by eliminating irrelevant features and only using significant features for the classification process. These results show that feature selection techniques are important in developing efficient and reliable phishing detection systems. In the future, this research can be expanded by testing the proposed model on other datasets to ensure the generalization and robustness of the model. Additionally, further research can be conducted to optimize model parameters and further explore other combinations of deep learning techniques to improve phishing detection performance. External validation and additional evaluation would be beneficial to ensure that the results truly reflect the model's performance under various real-world conditions. In addition, further research can be carried out to optimize model parameters and explore the combination of deep learning and even quantum computing techniques [47] to improve phishing detection performance further.

Author Contributions: Conceptualization: D.R.I.M.S. and S.W.; Methodology: D.R.I.M.S.; Software: A.N.S.; Validation: S.W., A.N.S. and S.B.; Formal analysis: S.B.; Investigation: D.R.I.M.S.; Resources: S.W.; Data curation: A.N.S.; Writing—original draft preparation: D.R.I.M.S.; Writing—review and editing: S.W., A.N.S. and S.B.; Visualization: S.B.; Supervision: D.R.I.M.S.; Project administration: S.W.; Funding acquisition: All.

Funding: This research received no external funding.

Data Availability Statement: The code and mirror dataset can be downloaded via URL https://github.com/MosesdeRosal/Web_Phishing_UCI.git

Conflicts of Interest: The authors declare no conflict of interest.

References

[1] J. K. Oladele *et al.*, "BEHeDaS: A Blockchain Electronic Health Data System for Secure Medical Records Exchange," *J. Comput. Theor. Appl.*, vol. 1, no. 3, pp. 231–242, Jan. 2024, doi: 10.62411/jcta.9509.

[2] E. A. L. Marazqah Btoush, X. Zhou, R. Gururajan, K. C. Chan, R. Genrich, and P. Sankaran, "A systematic review of literature on credit card cyber fraud detection using machine and deep learning," *PeerJ Comput. Sci.*, vol. 9, p. e1278, Apr. 2023, doi: 10.7717/peerj-cs.1278.

[3] S. Tanwar, T. Paul, K. Singh, M. Joshi, and A. Rana, "Classification and Imapct of Cyber Threats in India: A review," in *2020 8th International Conference on Reliability, Infocom Technologies and Optimization (Trends and Future Directions) (ICRITO)*, Jun. 2020, pp. 129–135. doi: 10.1109/ICRITO48877.2020.9198024.

[4] CrowdStrike, "CrowdStrike 2024 Global Threat Report," 2024. Accessed: Jul. 10, 2024. [Online]. Available: https://www.crowdstrike.com/global-threat-report/

[5] I. Saha, D. Sarma, R. J. Chakma, M. N. Alam, A. Sultana, and S. Hossain, "Phishing Attacks Detection using Deep Learning Approach," in *2020 Third International Conference on Smart Systems and Inventive Technology (ICSSIT)*, Aug. 2020, no. Icssit, pp. 1180–1185. doi: 10.1109/ICSSIT48917.2020.9214132.

[6] C. Catal, G. Giray, B. Tekinerdogan, S. Kumar, and S. Shukla, "Applications of deep learning for phishing detection: a systematic literature review," *Knowl. Inf. Syst.*, vol. 64, no. 6, pp. 1457–1500, Jun. 2022, doi: 10.1007/s10115-022-01672-x.

[7] R. Alazaidah, A. Al-Shaikh, M. R. Al-Mousa, H. Khafajah, G. Samara, and M. Alzyoud, "Website Phishing Detection Using Machine Learning Techniques," *J. Stat. Appl. Probab.*, vol. 13, no. 1, pp. 119–129, Jan. 2024, doi: 10.18576/jsap/130108.

[8] N. Q. Do, A. Selamat, O. Krejcar, E. Herrera-Viedma, and H. Fujita, "Deep Learning for Phishing Detection: Taxonomy, Current Challenges and Future Directions," *IEEE Access*, vol. 10, pp. 36429–36463, 2022, doi: 10.1109/ACCESS.2022.3151903.

[9] 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.2020.2969780.

[10] A. A. Ojugo and A. O. Eboka, "Comparative Evaluation for High Intelligent Performance Adaptive Model for Spam Phishing Detection," vol. 3, no. 1, pp. 9–15, Nov. 2018, Accessed: Dec. 21, 2023. [Online]. Available: http://pubs.sciepub.com/dt/3/1/2/index.html

[11] S. Shabudin, N. S. Sani, K. A. Z. Ariffin, and M. Aliff, "Feature selection for phishing website classification," *Int. J. Adv. Comput. Sci. Appl.*, vol. 11, no. 4, pp. 587–595, 2020, doi: 10.14569/IJACSA.2020.0110477.

[12] M. N. Alam, D. Sarma, F. F. Lima, I. Saha, R. E. Ulfath, and S. Hossain, "Phishing attacks detection using machine learning approach," *Proc. 3rd Int. Conf. Smart Syst. Inven. Technol. ICSSIT 2020*, no. Icssit, pp. 1173–1179, 2020, doi: 10.1109/ICSSIT48917.2020.9214225.

[13] S. Alnemari and M. Alshammari, "Detecting Phishing Domains Using Machine Learning," *Appl. Sci.*, vol. 13, no. 8, p. 4649, Apr. 2023, doi: 10.3390/app13084649.

[14] J. Kumar, A. Santhanavijayan, B. Janet, B. Rajendran, and B. S. Bindhumadhava, "Phishing Website Classification and Detection Using Machine Learning," in *2020 International Conference on Computer Communication and Informatics (ICCCI)*, Jan. 2020, pp. 1–6. doi: 10.1109/ICCCI48352.2020.9104161.

[15] S. A. Khan, W. Khan, and A. Hussain, "Phishing Attacks and Websites Classification Using Machine Learning and Multiple Datasets (A Comparative Analysis)," in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 12465 LNAI, 2020, pp. 301–313. doi: 10.1007/978-3-030-60796-8_26.

[16] W. Sarasjati *et al.*, "Comparative Study of Classification Algorithms for Website Phishing Detection on Multiple Datasets," in *2022 International Seminar on Application for Technology of Information and Communication (iSemantic)*, Sep. 2022, pp. 448–452. doi: 10.1109/iSemantic55962.2022.9920475.

[17] Y. Muliono, M. A. Ma'ruf, and Z. M. Azzahra, "Phishing Site Detection Classification Model Using Machine Learning Approach," *Eng. Math. Comput. Sci. J.*, vol. 5, no. 2, pp. 63–67, May 2023, doi: 10.21512/emacsjournal.v5i2.9951.

[18] A. Mughaid, S. AlZu'bi, A. Hnaif, S. Taamneh, A. Alnajjar, and E. A. Elsoud, "An intelligent cyber security phishing detection system using deep learning techniques," *Cluster Comput.*, vol. 25, no. 6, pp. 3819–3828, Dec. 2022, doi: 10.1007/s10586-022-03604-4.

[19] B. M. P. Waseso and N. A. Setiyanto, "Web Phishing Classification using Combined Machine Learning Methods," *J. Comput. Theor. Appl.*, vol. 1, no. 1, pp. 11–18, Aug. 2023, doi: 10.33633/jcta.v1i1.8898.

[20] A. K. Dutta, "Detecting phishing websites using machine learning technique," *PLoS One*, vol. 16, no. 10, p. e0258361, Oct. 2021, doi: 10.1371/journal.pone.0258361.

[21] N. N. Wijaya, D. R. I. M. Setiadi, and A. R. Muslikh, "Music-Genre Classification using Bidirectional Long Short-Term Memory and Mel-Frequency Cepstral Coefficients," *J. Comput. Theor. Appl.*, vol. 1, no. 3, pp. 243–256, Jan. 2024, doi: 10.62411/jcta.9655.

[22] M. A. Adebowale, K. T. Lwin, and M. A. Hossain, "Intelligent phishing detection scheme using deep learning algorithms," *J. Enterp. Inf. Manag.*, vol. 36, no. 3, pp. 747–766, Apr. 2023, doi: 10.1108/JEIM-01-2020-0036.

[23] A. Aljofey, Q. Jiang, Q. Qu, M. Huang, and J.-P. Niyigena, "An Effective Phishing Detection Model Based on Character Level Convolutional Neural Network from URL," *Electronics*, vol. 9, no. 9, p. 1514, Sep. 2020, doi: 10.3390/electronics9091514.

[24] S. Y. Yerima and M. K. Alzaylaee, "High Accuracy Phishing Detection Based on Convolutional Neural Networks," in *2020 3rd International Conference on Computer Applications & Information Security (ICCAIS)*, Mar. 2020, pp. 1–6. doi: 10.1109/ICCAIS48893.2020.9096869.

[25] 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," *Electronics*, vol. 12, no. 1, p. 232, Jan. 2023, doi: 10.3390/electronics12010232.

[26] M. F. Khan and B. L. Rana, "Detection of Phishing Websites Using Deep Learning Techniques," *Turkish J. Comput. Math. Educ.*, vol. 12, no. 10, pp. 3880–3892, 2021.

[27] J. Chung, C. Gulcehre, K. Cho, and Y. Bengio, "Empirical Evaluation of Gated Recurrent Neural Networks on Sequence Modeling," *arXiv*. Dec. 11, 2014. [Online]. Available: http://arxiv.org/abs/1412.3555

[28] D. R. I. M. Setiadi, K. Nugroho, A. R. Muslikh, S. W. Iriananda, and A. A. Ojugo, "Integrating SMOTE-Tomek and Fusion Learning with XGBoost Meta-Learner for Robust Diabetes Recognition," *J. Futur. Artif. Intell. Technol.*, vol. 1, no. 1, pp. 23–38, May 2024, doi: 10.62411/faith.2024-11.

[29] D. R. I. M. Setiadi, H. M. M. Islam, G. A. Trisnapradika, and W. Herowati, "Analyzing Preprocessing Impact on Machine Learning Classifiers for Cryotherapy and Immunotherapy Dataset," *J. Futur. Artif. Intell. Technol.*, vol. 1, no. 1, pp. 39–50, Jun. 2024, doi: 10.62411/faith.2024-2.

[30] S. Ali, A. Hashmi, A. Hamza, U. Hayat, and H. Younis, "Dynamic and Static Handwriting Assessment in Parkinson's Disease: A Synergistic Approach with C-Bi-GRU and VGG19," *J. Comput. Theor. Appl.*, vol. 1, no. 2, pp. 151–162, Dec. 2023, doi: 10.33633/jcta.v1i2.9469.

[31] A. A. Ubing, S. K. B. Jasmi, A. Abdullah, N. Z. Jhanjhi, and M. Supramaniam, "Phishing website detection: An improved accuracy through feature selection and ensemble learning," *Int. J. Adv. Comput. Sci. Appl.*, vol. 10, no. 1, pp. 252–257, 2019, doi: 10.14569/IJACSA.2019.0100133.

[32] L. Lakshmi, M. P. Reddy, C. Santhaiah, and U. J. Reddy, "Smart Phishing Detection in Web Pages using Supervised Deep Learning Classification and Optimization Technique ADAM," *Wirel. Pers. Commun.*, vol. 118, no. 4, pp. 3549–3564, 2021, doi: 10.1007/s11277-021-08196-7.

[33] Y. A. Alsariera, A. V. Elijah, and A. O. Balogun, "Phishing Website Detection: Forest by Penalizing Attributes Algorithm and Its Enhanced Variations," *Arab. J. Sci. Eng.*, vol. 45, no. 12, pp. 10459–10470, 2020, doi: 10.1007/s13369-020-04802-1.

[34] M. I. Akazue, I. A. Debekeme, A. E. Edje, C. Asuai, and U. J. Osame, "UNMASKING FRAUDSTERS: Ensemble Features Selection to Enhance Random Forest Fraud Detection," *J. Comput. Theor. Appl.*, vol. 1, no. 2, pp. 201–211, Dec. 2023, doi: 10.33633/jcta.v1i2.9462.

[35] M. S. Sunarjo, H. Gan, and D. R. I. M. Setiadi, "High-Performance Convolutional Neural Network Model to Identify COVID-19 in Medical Images," *J. Comput. Theor. Appl.*, vol. 1, no. 1, pp. 19–30, Aug. 2023, doi: 10.33633/jcta.v1i1.8936.

[36] I. M. Zubair and B. Kim, "A Group Feature Ranking and Selection Method Based on Dimension Reduction Technique in High-Dimensional Data," *IEEE Access*, vol. 10, pp. 125136–125147, 2022, doi: 10.1109/ACCESS.2022.3225685.

[37] G. S. Thejas, R. Garg, S. S. Iyengar, N. R. Sunitha, P. Badrinath, and S. Chennupati, "Metric and Accuracy Ranked Feature Inclusion: Hybrids of Filter and Wrapper Feature Selection Approaches," *IEEE Access*, vol. 9, pp. 128687–128701, 2021, doi: 10.1109/ACCESS.2021.3112169.

[38] D. M. D. Raj and R. Mohanasundaram, "An Efficient Filter-Based Feature Selection Model to Identify Significant Features from High-Dimensional Microarray Data," *Arab. J. Sci. Eng.*, vol. 45, no. 4, pp. 2619–2630, Apr. 2020, doi: 10.1007/s13369-020-04380-2.

[39] F. Masood, J. Masood, H. Zahir, K. Driss, N. Mehmood, and H. Farooq, "Novel Approach to Evaluate Classification Algorithms and Feature Selection Filter Algorithms Using Medical Data," *J. Comput. Cogn. Eng.*, vol. 2, no. 1, pp. 57–67, May 2022, doi: 10.47852/bonviewJCCE2202238.

[40] O. Jaiyeoba, E. Ogbuju, O. T. Yomi, and F. Oladipo, "Development of a Model to Classify Skin Diseases using Stacking Ensemble Machine Learning Techniques," *J. Comput. Theor. Appl.*, vol. 2, no. 1, pp. 22–38, May 2024, doi: 10.62411/jcta.10488.

[41] D. R. I. M. Setiadi, D. Marutho, and N. A. Setiyanto, "Comprehensive Exploration of Machine and Deep Learning Classification Methods for Aspect-Based Sentiment Analysis with Latent Dirichlet Allocation Topic Modeling," *J. Futur. Artif. Intell. Technol.*, vol. 1, no. 1, pp. 12–22, May 2024, doi: 10.62411/faith.2024-3.

[42] F. M. Firnando, D. R. I. M. Setiadi, A. R. Muslikh, and S. W. Iriananda, "Analyzing InceptionV3 and InceptionResNetV2 with Data Augmentation for Rice Leaf Disease Classification," *J. Futur. Artif. Intell. Technol.*, vol. 1, no. 1, pp. 1–11, May 2024, doi: 10.62411/faith.2024-4.

[43] K. Pham, D. Kim, S. Park, and H. Choi, "Ensemble learning-based classification models for slope stability analysis," *CATENA*, vol. 196, p. 104886, Jan. 2021, doi: 10.1016/j.catena.2020.104886.

[44] M. K. Pandey, M. K. Singh, S. Pal, and B. B. Tiwari, "Detection of Phishing Website Using Intelligent Machine Learning Classifiers," in *Soft Computing and Signal Processing*, 2023, pp. 21–29. doi: 10.1007/978-981-19-8669-7_3.

[45] S. Kapan and E. Sora Gunal, "Improved Phishing Attack Detection with Machine Learning: A Comprehensive Evaluation of Classifiers and Features," *Appl. Sci.*, vol. 13, no. 24, p. 13269, Dec. 2023, doi: 10.3390/app132413269.

[46] T.-T. Wong and P.-Y. Yeh, "Reliable Accuracy Estimates from k -Fold Cross Validation," *IEEE Trans. Knowl. Data Eng.*, vol. 32, no. 8, pp. 1586–1594, Aug. 2020, doi: 10.1109/TKDE.2019.2912815.

[47] A. N. Safriandono, D. R. I. M. Setiadi, A. Dahlan, F. Z. Rahmanti, I. S. Wibisono, and A. A. Ojugo, "Analyzing Quantum Feature Engineering and Balancing Strategies Effect on Liver Disease Classification," *J. Futur. Artif. Intell. Technol.*, vol. 1, no. 1, pp. 51–63, Jun. 2024, doi: 10.62411/faith.2024-12.