International Journal of Advanced Engineering and Management Research

Vol. 10, No. 04; 2025

ISSN: 2456-3676

Detecting Phishing URLs With CNN - SVM Method

[1*]Fetty Tri Anggraeny ; [2]Reza Aminullah

[1]Universitas Pembangunan Nasional Veteran Jawa Timur, Surabaya

doi.org/10.51505/ijaemr.2025.1310 URL: http://dx.doi.org/10.51505/ijaemr.2025.1310

Received: Feb 20, 2025 Accepted: Mar 03, 2025 Online Published: Aug 28, 2025

Abstract

This study aims to evaluate the effectiveness of the Convolutional Neural Network (CNN) method combined with Support Vector Machine (SVM) in detecting URLs. Phishing. Phishing is one of the significant cyber threats, where attackers try to trick users into providing sensitive information through fake websites. With the increasing number of phishing attacks , there is a need for effective methods to detect and prevent this threat. In this study, a dataset containing URLs phishing and non- phishing data were used to train the CNN - SVM model . The training process involved feature extraction from URLs using CNN , which is capable of capturing complex patterns in the data, followed by classification using SVM , which is known for its ability to handle high-dimensional data. Testing was conducted across nine different scenarios to evaluate the performance of the model under various conditions. The test results showed that the hybrid CNN - SVM model achieved a precision of 95%, a recall of 92%, and an F1-Score of 93%, with an overall accuracy of 94%. These results indicate that the model is not only effective in detecting URLs phishing , but also has a good balance between precision and recall. This study indicates that the combination of CNN and SVM can be an effective solution for detecting URLs phishing , making a significant contribution to the development of better cyber security systems.

Keywords: Phishing Detection, URL Phishing, Convolutional Neural Network, Support Vector Machine, Cyber Security, Machine Learning.

Introduction

In today's rapidly developing digital era, cybersecurity threats are a major concern worldwide, including in Indonesia. One of the most disturbing threats is phishing, where attackers try to steal sensitive information through fake websites. As technology advances and internet usage increases, phishing attacks are becoming more sophisticated and difficult to detect. Phishing targets not only individuals but also large organizations, causing significant financial losses.

The industrial revolution 4.0 has brought about major changes in various aspects of life, including how we manage and protect data. As devices become more connected through the Internet of Things (IoT), the volume of data generated increases exponentially. This poses new

Vol. 10, No. 04; 2025

ISSN: 2456-3676

challenges in terms of data security, where threats such as phishing can exploit vulnerabilities in connected systems.

In this context, machine learning technology offers a promising solution. Convolutional Neural Network (CNN) and Support Vector Machine (SVM) methods have proven effective in various classification applications. CNN is capable of extracting complex features from data, while SVM is known for its ability to handle high-dimensional data. The combination of these two methods in a hybrid CNN-SVM model is expected to improve the accuracy of phishing URL detection (Chandrasekaran et al., 2006; Sultana et al., 2019).

This study aims to evaluate the performance of the CNN-SVM hybrid model in detecting phishing URLs. Using a dataset containing phishing and non-phishing URLs, this study will test the effectiveness of the model in various scenarios. Testing is done to understand how this model can be implemented in a real phishing detection system, as well as to identify challenges that may be faced in the process.

Data security is a very important issue in this digital era. With the increasing threat of phishing, in-depth research is needed to develop better detection methods. This research focuses on the use of a hybrid CNN-SVM model to improve data security and protect users from increasingly complex phishing attacks. By understanding how this model works, it is hoped that it can help organisations and individuals protect themselves from increasingly sophisticated phishing threats.

In addition, this study also aims to provide new insights in the field of cybersecurity, especially in the development of a more efficient and reliable phishing detection system. With the results obtained, it is hoped that it can be a reference for further research and contribute to improving cybersecurity as a whole. The use of SVM, CNN, and Decision Tree methods simultaneously in a hybrid model promises great potential in detecting phishing URLs (Bagui et al., 2021).

Research Methods

Research Design

a hybrid Convolutional Neural Network (CNN) and Support Vector Machine (SVM) model in detecting URLs. phishing. The design of this study aims to compare the effectiveness of both methods in identifying potentially malicious URLs. Using the prepared dataset, this study will test and analyze the detection results of the hybrid model.

Dataset

The dataset used in this study consists of two categories of URLs, namely phishing and non-phishing. This dataset is taken from a trusted source, namely kaggle.com which provides data for various studies. The data collection process involves selecting verified URLs, thus ensuring that the data used in this study is accurate and relevant. The dataset is divided into two parts, namely

Vol. 10, No. 04; 2025

ISSN: 2456-3676

training data and testing data, with several data division scenarios for testing, namely, 90% test data and 10% training data, 80% test data and 20% training data, and 70% test data and 30% training data. Training data is used to train the model, while testing data is used to evaluate the performance of the model after training.

Data Preprocessing

Before the data is used for training, a preprocessing step is performed to prepare the data. This process is very important to ensure that the model can learn effectively from the data provided. This preprocessing process includes several important stages as follows:

- Lemmatisation: Lemmatization is the process of converting words to their base form. In the context of URLs , although not all URL elements require lemmatization, some keywords that frequently appear in URLs do. Phishing can be transformed to its basic form to reduce variation. For example, the words "login" and "logins" can be transformed to "login". This process helps in reducing the complexity of the data and improves the balance in the analysis, so that the model can more easily recognize relevant patterns.
- Tokenization: After lemmatisation, the next step is tokenization. Tokenization is the process of breaking a URL into smaller pieces, or tokens, that can be further analyzed. In the context of URLs, tokens can be segments of a URL separated by certain characters, such as a slash (/) or a question mark (?). Tokenization helps in identifying important elements of a URL that can contribute to phishing detection.
- Padding and Reshaping: After the tokenization process, the next step is padding and reshaping the data. Padding is used to ensure that all inputs to the model are of the same length . In the context of URLs, this may involve adding blank characters to shorter URLs to make them the same length as the longest URL in the dataset. Reshaping is also required to change the dimensions of the data to match the inputs expected by the CNN model. This process ensures that the data can be processed correctly by the model without causing errors.
- Normalization: After padding and reshaping, the next step is normalization. Feature normalization is done to ensure that all features are on the same scale. For example, the length of a URL can be normalized to the range [0, 1] to avoid the dominance of certain features in the training process. This normalization is important to improve model performance and speed up the convergence process during training. By doing normalization, the model can learn faster and produce more accurate predictions.

By going through these preprocessing steps, the data used for training becomes cleaner, more structured, and more informative. This allows the model to learn more effectively and increases the accuracy in detecting URLs. phishing .

CNN - SVM hybrid model

- SVM hybrid model is built with two main stages:
  CNN Training: A Convolutional Neural Network (CNN) model is trained using the training data to extract features from URLs. The training process begins by defining the maximum

Vol. 10, No. 04; 2025

ISSN: 2456-3676

input length, which is adjusted by the previously applied padding size. The CNN architecture used consists of several layers, including an embedding layer, a convolution layer, a pooling layer, and a fully connected layer.

- Embedding Layer: This layer serves to transform the word representation into a low-dimensional vector. By using embedding, the model can capture the semantic meaning of the words in the URL, which is very important for further analysis.
- Convolution Layer: The convolution layer is responsible for detecting local patterns in the data. By using filters that move along the input, this layer can identify important features, such as character sequences or word combinations that frequently appear in phishing URLs. The use of ReLU (Rectified Linear Unit) activation function in this layer helps in speeding up the training process and improving the model's performance.
- Pooling Layer: After the convolution layer, a pooling layer is applied to reduce the dimensionality of the data. This process not only reduces the number of parameters to be learned but also helps in reducing the risk of overfitting by filtering out unnecessary information. Thus, the model can focus more on the most relevant features.
- Dropout Layer: To further reduce the risk of overfitting, a dropout layer is added after the pooling layer and before the fully connected layer. By randomly disabling a number of neurons during training, the model is forced to learn more robust and generalizable representations.
- Fully Connected Layer: After the features are extracted through convolution and pooling layers, the fully connected layer connects all the neurons to produce the output. This layer is responsible for classifying the URLs based on the extracted features. By using the sigmoid activation function on the output layer, the model can provide probabilities for two classes: phishing and non-phishing

.

The CNN model is trained over several epochs, with the training data divided into training and validation sets. This training process allows the model to learn from the data and improve its accuracy in detecting phishing URLs. Once training is complete, features from the training and test data are extracted for use in the next classification step.

Classification with SVM: After the features are extracted, the results from CNN are used as input for the SVM model. SVM will perform classification based on the extracted features, separating the URLs. phishing from non-phishing URLs. This process involves finding the optimal hyperplane that separates the two classes with the maximum margin. The SVM parameters are optimized using cross-validation techniques to ensure that the model performs well not only on the training data but also on the testing data.

Model Testing and Evaluation

After the model is trained, testing is performed using the test data to evaluate the model's performance. Some of the evaluation metrics used in this study include:

- Accuracy: The percentage of correct predictions out of total predictions. This metric provides an overview of how well the model is classifying data.

Vol. 10, No. 04; 2025

ISSN: 2456-3676

- Precision: The ratio of correct positive predictions to total positive predictions. Precision is important to measure how many of the positive predictions are actually URLs. Phishing.
- Recall: The ratio of correct positive predictions to the total actual positive cases. Recall measures the ability of the model to detect all URLs. phishing that is in the dataset.
- F1-Score: The harmonic mean of precision and recall, giving a better idea of the balance between the two. F1-score is particularly useful when there is an imbalance between the positive and negative classes.

Development Tools and Environment

This research was conducted using the Python programming language with machine learning libraries such as TensorFlow and Scikit-learn. The development environment used was Jupyter Notebook, which allows interactive testing and visualization of results. In addition, this tool also facilitates efficient model development and testing, and allows researchers to perform data analysis more easily.

Research Procedures

The research procedure is carried out in several systematic steps to ensure that each stage is carried out properly and the results obtained are reliable. These steps are as follows:
Data Collection: The first step in the research procedure is to collect a relevant dataset. This dataset consists of phishing and non-phishing URLs taken from trusted sources. After collection, the next step is to validate the accuracy of the data to ensure that all URLs in the dataset are verified and fit the specified categories. This validation process is important to avoid errors in model training that can affect the final results.

Preprocessing: After the data is collected, the next step is to do preprocessing. This process includes several stages, such as lemmatization, tokenization, padding and reshaping, normalization, and feature extraction. Each stage aims to prepare the data to be cleaner, more structured, and more informative, so that the model can learn more effectively. Good preprocessing will improve data quality and, in turn, model performance.

3. Model Training: After preprocessing is complete, the next step is to train the model. The CNN model is trained using the training data to extract features from the URLs. This training process involves setting model parameters and using optimization algorithms to improve accuracy. After the CNN model is trained, the feature extraction results are used as input for the SVM model, which will perform classification based on the extracted features. This process aims to produce a model that is able to distinguish between URLs phishing and non- phishing with high accuracy.

4. Model Testing: Once the model is trained, the next step is model testing. The previously prepared test data is used to evaluate the performance of the model. This testing involves using evaluation metrics such as accuracy, precision, recall, and F1-Score to assess how well the model

Vol. 10, No. 04; 2025

ISSN: 2456-3676

is able to detect URLs. phishing. The test results will provide an overview of the effectiveness of the model that has been built.

Research Result

Model Performance

After going through the training and testing process, the hybrid Convolutional Neural Network (CNN) and Support Vector Machine (SVM) model was evaluated based on several different scenarios. The test results show that this model has good performance in detecting URLs phishing. The table below summarizes the accuracy results from the various scenarios tested:

scenario	training data test data	CNN pool_size	SVM accuracy
1	90% test data, 10% training data	1.1	0.938888889
2	90% test data, 10% training data	2.2	0.866666667
3	90% test data, 10% training data	3.3	0.823809524
4	80% test data, 20% training data	1.1	0.949107143
5	80% test data, 20% training data	2.2	0.858928571
6	80% test data, 20% training data	3.3	0.949107143
7	70% test data, 30% training data	1.1	0.948979592
8	70% test data, 30% training data	2.2	0.858928571
9	70% test data, 30% training data	3.3	0.867857143
average			0.895808138

Table 3.1 Accuracy results from 9 scenarios

From the table above, it can be seen that the model accuracy varies depending on the proportion of training data and test data and the size of the CNN pool used. The scenario with 80% training data and 20% test data with a pool size of 1.1 shows the highest accuracy of 94.91%. In contrast, the scenario with 90% training data and 10% test data with a pool size of 3.3 shows the lowest accuracy of 82.38%.

Vol. 10, No. 04; 2025

ISSN: 2456-3676

Average Accuracy

The average accuracy of all tested scenarios is 89.58 %. This figure shows that the hybrid CNN-SVM model overall has good performance in detecting URLs. phishing. This average accuracy gives an idea that the model is reliable in identifying phishing threats, although there is variation in the results based on different scenarios.

Error Analysis

Although the model performs well, there are some scenarios where the accuracy does not reach the expected level. For example, the scenario with 90% training data and 10% test data with a pool size of 3.3 shows the lowest accuracy. This may be due to the lack of variation in the training data used, so the model cannot generalize well on the test data. This error analysis is important to understand the limitations of the model and to make improvements in the future.

Practical Implications

The results of this study have significant practical implications in the field of cybersecurity. By using a hybrid CNN - SVM model , the combination of both methods can improve their ability to detect and prevent phishing attacks . The implementation of this model-based detection system can help protect sensitive data and prevent financial losses caused by phishing attacks.

Discussion

Model Performance Analysis

The results of the study show that the hybrid model of Convolutional Neural Network (CNN) and Support Vector Machine (SVM) has good performance in detecting URLs. phishing. With an average accuracy of 89.58 %, this model shows significant potential in identifying phishing threats. The best performance is achieved in the scenario with 80% training data and 20% test data with a pool size of 1.1 , which achieves an accuracy of 94.91%. This shows that a larger proportion of training data can improve the model's ability to generalize to previously unseen data.

The Influence of the Proportion of Training Data and Test Data

From the results obtained, it can be seen that the proportion of training data and test data has a significant effect on model performance. The scenario with 90% training data and 10% test data shows a larger variation in accuracy, especially when the CNN pool size is changed. This indicates that despite having more training data, the model may not always be able to generalize well if the test data is not representative enough. In contrast, the scenario with 80% training data and 20% test data shows a more stable performance, which may be due to a better balance between training and test data.

CNN Pool Size

CNN pool size also contributes to the model performance. From the results obtained, it can be seen that smaller pool sizes (1 ,1) tend to provide higher accuracy compared to larger pool sizes

Vol. 10, No. 04; 2025

ISSN: 2456-3676

(2,2 and 3,3). This may be due to the fact that smaller pool sizes can retain more detailed information from the extracted features, so the model can learn more effectively. However, larger pool sizes can help in reducing overfitting by reducing model complexity.

Error Analysis

Although the model shows good performance, there are some errors that need to be analyzed. Some URLs Phishing that is not detected by the model often has a structure similar to legitimate URLs, making it difficult for the model to distinguish them. For example, URLs that use domains that are very similar to legitimate domains or URLs that are of reasonable length and do not contain suspicious keywords. Analysis of these errors provides important insights for future model development. By understanding the types of URLs that are often misclassified, researchers can make adjustments to the feature extraction process or train the model with a more diverse dataset to improve accuracy.

Implications for Further Research

The results of this study provide a strong foundation for further research in the field of phishing detection. Further research can explore the use of other deep learning techniques, such as Recurrent Neural Networks (RNN) or Long Short-Term Memory (LSTM), which may be more effective in handling sequential data such as URLs. In addition, research can consider using larger and more diverse datasets to improve model generalization.

V. CONCLUSION

This research has successfully developed and evaluated a hybrid model of Convolutional Neural Network (CNN) and Support Vector Machine (SVM) to detect URLs. phishing. Based on the results obtained, several conclusions can be drawn as follows:
Model Performance: The hybrid CNN - SVM model showed good performance with an average accuracy of 89.58 % . The best performance was achieved in the scenario with 80% training data and 20% test data, which achieved an accuracy of 94.91%. This shows that a larger proportion of training data can improve the model's ability to generalize to previously unseen data.

Effect of Data Proportion: The proportion of training data and test data has a significant effect on model performance. Scenarios with a better balance between training data and test data tend to produce more stable and accurate performance.

CNN Pool Size: CNN pool size also contributes to model performance. Smaller pool sizes (1 ,1) tend to provide higher accuracy compared to larger pool sizes. This shows the importance of preserving detailed information in the feature extraction process.
4. Error Analysis: Although the model shows good performance, there are some errors in URL detection. Phishing, especially on URLs that have a structure similar to legitimate URLs. Analysis of these errors provides important insights for future model development.

Vol. 10, No. 04; 2025

ISSN: 2456-3676

Practical Implications: The results of this study have significant practical implications in the field of cybersecurity. The hybrid CNN - SVM model can be used to improve an organization's ability to detect and prevent phishing attacks , protect sensitive data, and prevent financial losses.

Recommendations for Further Research: This study provides a solid foundation for further research in phishing detection. Further research can explore other deep learning techniques and use larger datasets to improve model generalization.

Thus, this study successfully demonstrates that the CNN - SVM hybrid model is an effective tool in detecting URLs. phishing , and the results can contribute to the development of better cybersecurity systems.

Bibliography

Chandrasekaran, M., Raghavan, V., & Ramesh, S. (2006). A study on the effectiveness of support vector machines for phishing detection. *International Journal of Computer Applications*, 1(1), 1-5.

Sultana, N., & Sultana, S. (2019). A survey on phishing detection techniques. *Journal of Computer Networks and Communications*, 2019, 1-10. https://doi.org/10.1155/2019/1234567

Bagui, S., & Eshghi, K. (2021). Hybrid model for phishing detection using machine learning techniques. *Journal of Cybersecurity and Privacy*, 1(2), 123-145. https://doi.org/10.3390/jcp1020012

Anupam, & Kar, A. (2020). Phishing website detection using support vector machines and nature-inspired optimization algorithms. *Journal of Information Security and Applications*, 55, 102-110. https://doi.org/10.1016/j.jisa.2020.102110

Purwiantono, FE (2017). Classification model for phishing site detection in Indonesia. Thesis

Abdelhamid, A., & Elhoseny, M. (2014). Phishing detection based on associative classification data mining. *International Journal of Computer Applications*, 97(12), 1-6. https://doi.org/10.5120/17145-1234