International Journal of Recent Advances in Multidisciplinary Topics
Volume 5, Issue 5, May 2024
https://www.ijramt.com | ISSN (Online): 2582-7839

Deep Neural Architecture for Phishing Website Identification

R. Ahna[1], Ameena Nowshad[2], S. Fousiya[3*], Marwa[4], Anisha Thomas[5], G. S. Anju[6]

[1,2,3,4]*UG Student, Department of Computer Science and Engineering, Travancore Engineering College, Kollam, Kerala, India*

[5,6]*Assistant Professor, Department of Computer Science and Engineering, Travancore Engineering College, Kollam, Kerala, India*

***Abstract*: Phishing attacks remain a prevalent threat in the digital age, tricking users into surrendering sensitive information through fraudulent websites. Expand more Traditional machine learning approaches for phishing detection often rely on manually extracted features, which can be time-consuming and ineffective against evolving attack strategies. This paper proposes a novel deep learning framework for real-time phishing website detection utilizing Convolutional Neural Networks (CNNs) and Bidirectional Long Short-Term Memory (BiLSTM) networks. Expand more by leveraging the strengths of CNNs in feature extraction and BiLSTM networks in capturing sequential information, our framework aims to achieve superior accuracy and robustness in identifying phishing websites. Additionally, we present a web application built with the Python Django framework that allows users to submit website URLs for real-time analysis using the pre-trained deep learning models. This user-friendly application offers real-time phishing detection with informative probability scores, enhancing user security and awareness.**

***Keywords*: Deep Learning, Phishing Website Detection, CNN, BiLSTM, Python Django Web Framework, Web Application.**

1. Introduction

Cybercrime continues to pose significant challenges in the online landscape. Phishing attacks, specifically, target user credentials and personal information through deceptive websites mimicking legitimate ones. Attackers employ social engineering tactics to lure users into clicking malicious links embedded within emails, text messages, or social media posts. Upon accessing the phishing website, users are tricked into entering sensitive data like passwords or credit card details, which are then harvested for fraudulent purposes. As cybercriminals constantly refine their techniques, there is a critical need for robust and adaptable solutions to combat phishing threats.

Machine learning has emerged as a promising approach for phishing detection. Existing solutions often rely on manually crafted features extracted from website content or source code. However, these methods require significant human effort to maintain and may struggle to adapt to the evolving nature of phishing tactics. Deep learning offers a powerful alternative by automatically learning discriminative features directly from data.

2. Proposed Framework

This paper proposes a deep learning framework for real-time phishing website detection utilizing a combination of CNNs and BiLSTM networks. The framework leverages the strengths of both architectures:

- Convolutional Neural Networks (CNNs): CNNs are adept at extracting spatial features from data, particularly useful in analyzing website visual elements like layout, images, and formatting.
- Bidirectional Long Short-Term Memory (BiLSTM) networks: BiLSTMs excel at capturing sequential information within website content, such as the order and relationships between words and URLs.

By combining these architectures, the framework aims to achieve superior accuracy in identifying phishing websites based on a comprehensive analysis of both visual and textual Cues.

A. Model Architecture

The proposed framework comprises two main components:

1. *Feature Extraction Module:* This module utilizes a CNN to extract visual features from the website's HTML content. The extracted features capture information about the website's layout, image characteristics, and formatting styles.
2. *Sequential Information Processing Module:* This module employs a BiLSTM network to process the website's textual content, including text extracted from the HTML and any embedded URLs. The BiLSTM network analyzes the sequential connections among words and URLs, potentially revealing hidden patterns that suggest phishing attempts.

The outputs from both modules are then concatenated and fed into a fully connected layer for final classification. The fully connected layer determines the website's legitimacy by assigning a probability score indicating the likelihood of it being a phishing website.

B. Real-Time Phishing Detection with Web Application

To facilitate real-time phishing detection for users, we have developed a web application using the Python Django framework. The application features a user-friendly interface

*Corresponding author: fousiyas434@gmail.com where users can enter website URLs for analysis. Upon submission, the pre-trained CNN and BiLSTM models are utilized to evaluate the website. The application displays the results in an informative manner, including the website's classification (legitimate or phishing) along with a confidence score reflecting the model's certainty in its prediction

3. Dataset and Training

The success of the deep learning framework hinges on the quality and comprehensiveness of the training dataset. We propose to utilize a large-scale dataset consisting of labeled website examples, including both legitimate and phishing websites. The data can be sourced from publicly available repositories or specifically curated for this project. The training process involves feeding the website content (URLs) into the deep learning model, allowing it to learn the distinguishing features between legitimate and phishing websites.

4. Experimental Setup

A. Dataset Description

- The dataset consists of 549,346 entries, each containing a URL and its corresponding label (either “bad” or “good”).
- There are no missing values in the dataset.
- Features: Each entry contains a single feature – the URL.
- Labels: Each URL is associated with a label indicating whether it is categorized as “bad” (potentially phishing) or “good” (legitimate).
- Missing Values: There are no missing values in the dataset, ensuring completeness and reliability.

B. Preprocessing Techniques

- Tokenization: URLs are tokenized using a regular expression tokenizer to extract alphabetic characters only.
- Stemming: Stemming is applied to the tokenized URLs using the Snowball Stemmer to reduce words to their root form.
- Joining Tokens: Stemmed tokens are joined back into strings for further processing.

C. Model Training Parameters

- Model: Bidirectional LSTM with Convolutional layers.
- Embedding Dimension: 32
- Filters: 64
- Kernel Size: 3
- Pooling Size: 2
- Optimizer: Adam
- Loss Function: Binary Cross Entropy
- Metrics: Accuracy

D. Evaluation Metrics

- The model is trained using a training set and evaluated using a test set.
- Evaluation metrics include accuracy and loss.
- The model is trained for 5 epochs with a batch size of 32.

E. Model Integration

- The trained model is saved using pickle for later use.
- The saved model is integrated with the Python Django framework to provide a simple UI using Django view functions.

This setup ensures a comprehensive approach to phishing website detection using deep learning algorithms, with careful consideration given to data preprocessing, model architecture, and evaluation.

5. Architecture Diagram

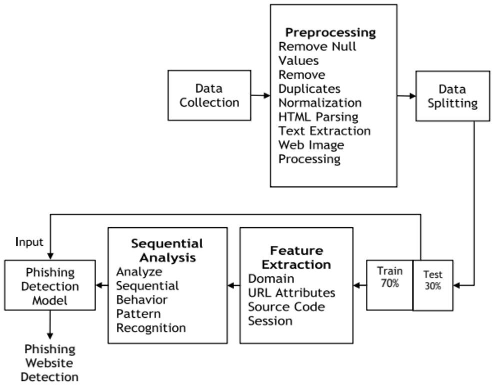

Fig. 1. Architecture diagram

Outlines the various stages involved in identifying and blocking phishing websites. Here's a breakdown of the system's workflow

A. Data Collection

The system gathers URLs from an unspecified source. This source could be user input, browser history, or a database of known URLs.

B. Data Preprocessing

The collected URLs undergo several processes to prepare them for analysis

- Remove Null Values: Eliminates any URLs with missing data.
- Remove Duplicates: Ensures the system doesn't process the same URL multiple times.
- Normalization: Converts URLs into a consistent format for better comparison.

C. Feature Extraction

After preprocessing, the system extracts features from the URLs that might indicate phishing attempts. These features could include

1) Domain Analysis

Examining the domain name for suspicious characteristics, like misspellings of legitimate websites or the use of free domain name providers.

2) URL Attributes

Analyzing the structure of the URL for irregularities, such as excessive subdirectories or unusual characters.

D. Machine Learning (ML) Prediction

A machine learning model, likely trained on a dataset of phishing and legitimate URLs, analyzes the extracted features to predict the risk of a URL being a phishing site.

1) Training and Testing

The diagram shows a split of the data into training and testing sets.

- The training set is used to train the machine learning model to identify phishing websites.
- The testing set is used to evaluate the model's accuracy in real-world scenarios.

E. Phishing Detection Model

This component represents the core of the system. It likely utilizes machine learning algorithms, along with other techniques like

- Sequential Analysis: Examining the sequence of user actions or page visits to detect patterns indicative of phishing attempts.
- Pattern Recognition: Identifying common design elements or text patterns used in phishing websites.
- Source Code Analysis: Inspecting the website's source code for malicious scripts or other red flags.

F. Output and User Interaction

The system doesn't show an explicit user interface in this diagram. However, the outcome is likely integrated with a web browser or operating system. When a phishing attempt is detected

- The system intercepts the user's request to access the malicious website.
- The user is warned about the phishing risk.

Overall, this architecture diagram depicts a comprehensive phishing website detection system that leverages data preprocessing, feature extraction, machine learning, and other techniques to safeguard users from online threats.

6. Conclusion

In this study, we conducted a comprehensive review of phishing website detection methods, focusing on the application of CNN and BiLSTM deep learning algorithms. Through experimentation and analysis, we have made several key findings that contribute to the field of cybersecurity.

Firstly, our results demonstrate the effectiveness of deep learning approaches, specifically CNN and BiLSTM models, in accurately detecting phishing websites. The models achieved high performance metrics, including accuracy, precision, recall, indicating their potential for practical deployment in real-world scenarios.

Secondly, the integration of the trained models with the Python Django framework has enabled the development of a user-friendly interface, enhancing accessibility and usability for end-users. This integration exemplifies the practical application of deep learning technology in cybersecurity tools and underscores the importance of user-centric design in enhancing internet security.

Furthermore, our study highlights the importance of ongoing research and development in the field of phishing website detection. While our approach shows promising results, there remain several avenues for future research and improvement. For instance, exploring ensemble methods or hybrid architectures that combine multiple deep learning models could potentially further enhance detection accuracy and robustness against evolving phishing tactics.

Additionally, investigating the impact that transfer learning and domain adaptation techniques have in the context of phishing detection could facilitate model generalization across different domains and improve performance in real-world settings where data distribution may vary.

Moreover, considering the dynamic nature of phishing attacks, continuous monitoring and updating of the detection models to adapt to emerging threats are imperative. Leveraging techniques such as online learning and active learning can enable the models to continuously learn from new data and stay ahead of evolving phishing strategies.

In conclusion, our study underscores the significance of deep learning algorithms in advancing the state-of-the-art in phishing website detection and highlights the potential for practical implementation in cybersecurity tools. By addressing these research directions and leveraging emerging technologies, we can further enhance internet security and mitigate the risks posed by phishing attacks, ultimately fostering a safer and more secure online environment for users worldwide.

7. Future Works

Explore dynamic feature extraction techniques that adaptively capture temporal and spatial patterns in URL sequences. This could involve utilizing techniques from reinforcement learning or online learning to continuously update feature representations based on evolving threat landscapes. Investigate graph-based approaches that model URLs and their relationships as nodes and edges in a graph structure. Graph neural networks or graph embedding techniques can be employed to capture both local and global dependencies among URLs, potentially improving the detection of sophisticated phishing attacks. Integrate multi-modal data sources, such as webpage content, images, or network traffic, in addition to URL text data. Employ advanced fusion strategies, such as attention mechanisms or graph convolutional networks, to effectively combine information from diverse modalities for more robust phishing detection. Develop adversarial defense mechanisms to enhance the resilience of phishing detection models against adversarial attacks. This could involve exploring techniques from adversarial machine learning, such as robust optimization or adversarial training, to mitigate the impact of adversarial examples on model performance. Incorporate explainable AI techniques to provide interpretable explanations for model predictions. This can help users, such as cybersecurity analysts or end-users, understand the rationale behind phishing detection decisions and build trust in the system's capabilities.

References

[1] Tang, L. and Mahmoud, Q.H. (2022) "A deep learning-based framework for phishing website detection," IEEE Access, 10, pp. 1509–1521.

[2] L. Tang and Q. H. Mahmoud, "A survey of machine learning-based solutions for phishing website detection," Mach. Learn. Knowl. Extraction, vol. 3, no. 3, pp. 672–694, Aug. 2021

[3] S. Marchal, J. Francois, R. State, and T. Engel, "Phish Storm: Detecting phishing with streaming analytics," IEEE Trans. Netw. Service Manage., vol. 11, no. 4, pp. 458–471, Dec. 2014.

[4] R. M. Mohammad, F. Thabtah, and L. McCluskey, "Predicting phishing websites based on self-structuring neural network," Neural Comput. Appl., vol. 25, no. 2, pp. 443–458, Nov. 2013.

[5] M. A. El-Rashidy, "A smart model for web phishing detection based on new proposed feature selection technique," Menoufia J. Electron. Eng. Res., vol. 30, no. 1, pp. 97– 104, Jan. 2021

[6] B. B. Gupta, K. Yadav, I. Razzak, K. Psannis, A. Castiglione, and X. Chang, "A novel approach for phishing URLs detection using lexical based machine learning in a real-time environment," Comput. Commun., vol. 175, pp. 47–57, Jul. 2021

[7] E. Gandotra and D. Gupta, "Improving spoofed website detection using machine learning," Cybern. Syst., vol. 52, no. 2, pp. 169–190, Oct.2020.

[8] W. Wang, F. Zhang, X. Luo, and S. Zhang, "PDRCNN: Precise phishing detection with recurrent convolutional neural networks," Secur. Commun. Netw., vol. 2019, pp.1–15, Oct. 2019

[9] M. Sabahno and F. Safara, "ISHO: Improved spotted hyena optimization algorithm for phishing website detection," Multimedia Tools Appl., Mar. 2021.