A Comparative Analysis of Support Vector Machines, Decision Trees, and Long Short-Term Memory Networks in Phishing Website Detection

Madhav Adhikari, Saroj Pandey[b]
[b]*sarojpandey@kcc.edu.np*
[a b]*Kantipur City College, Kathmandu, 44600, Nepal*

Abstract

Phishing attacks pose a significant cybersecurity threat, with malicious actors continually evolving their tactics to deceive users and steal sensitive information. This research conducted a comparative analysis of machine learning (ML) algorithms, specifically Support Vector Machines (SVM) and Decision Trees, alongside deep learning Long Short-Term Memory Networks (LSTMs) for detecting phishing website URLs. Utilizing a dataset containing features extracted from URLs, we trained and evaluated the performance of these algorithms based on accuracy, precision, recall, and F1 score. The study also investigated the interpretability, computational complexity, and generalization capabilities of the models. Our findings indicated that LSTMs outperformed SVMs and Decision Trees, offering a balanced approach with low false positive and false negative rates. This balance was crucial in phishing detection, where the cost of missing a phishing attempt could be as severe as wrongly blocking legitimate access. The results underscored the importance of selecting a machine learning model based on the specific requirements and constraints of the phishing detection system, providing valuable guidance for cybersecurity practitioners and researchers in developing effective defense mechanisms against phishing attacks.

Keywords: Cybersecurity; Decision Trees; Deep learning, False negatives; Long Short-Term Memory Networks (LSTMs); Machine learning (ML); Phishing detection

1. INTRODUCTION

Phishing is a malicious practice in cybersecurity where attackers impersonate legitimate entities, such as banks or government agencies, to trick users into divulging sensitive information such as passwords, credit card numbers, or personal data. Typically, phishing attacks are carried out through deceptive emails, fake websites, or messages that appear authentic, luring victims into clicking on malicious links or providing confidential information. Phishing poses a significant threat to individuals, businesses, and organizations worldwide, leading to financial losses, identity theft, and compromised data security. Despite advancements in cybersecurity measures and awareness campaigns, phishing attacks continue to evolve, employing sophisticated techniques to exploit human vulnerabilities and bypass traditional security defenses.

Addressing the challenges posed by phishing attacks requires robust and adaptive detection mechanisms capable of distinguishing between legitimate and malicious URLs in real time. Traditional approaches, such as static rule-based methods or signature-based techniques, often struggle to keep up with the dynamic nature of phishing tactics. This research conducts a comprehensive analysis of traditional machine learning (ML) algorithms, specifically Support Vector Machines (SVM) and Decision Trees, compared to deep learning Long Short-Term Memory Networks (LSTMs) for detecting phishing website URLs. By utilizing a well-prepared dataset with features extracted from URLs and evaluating these models based on key performance metrics, this study aims to provide valuable insights into the strengths and limitations of ML and deep learning methods. This will guide cybersecurity practitioners and researchers in developing more effective defense mechanisms against phishing attacks, thereby enhancing the overall resilience of cybersecurity systems.

2. LITERATURE REVIEW

[1] showed a comprehensive exploration of phishing attacks and proposed enhancements to existing Machine Learning (ML)-based detection methods. The methodology involved detailed examinations of data mining techniques, phishing attack

deployment methods, and root causes of phishing, alongside an analysis of recent anti-phishing techniques. A review of various phishing detection solutions was conducted to inform the proposal for designing a more accurate and efficient ML model for identifying phishing URL patterns. Future implementation and performance comparison with existing ML models were planned, aiming to bolster online security against evolving threats.

[2] presented an investigation into phishing attack techniques and proposed a novel approach to phishing URL classification to address cybersecurity concerns. Unlike previous methods that only considered phishing URLs, this work incorporated both phishing and legitimate URLs to design a comprehensive data model for classification. Phishing and legitimate URLs were collected from the phish tank database and common sources, respectively, with 14 features calculated for classification using an Artificial Neural Network (ANN). Implementation utilized JAVA technology and the WEKA machine learning library, with the proposed model achieving a classification accuracy of 76.2% while minimizing resource consumption. Comparison with existing phishing detection models highlighted the effectiveness of the proposed approach, suggesting its potential for future extensions with deep learning and large-scale data analysis.

[3] proposed a taxonomy of deep learning algorithms for phishing detection based on a systematic literature review of 81 selected papers. It categorizes existing literature, introduces phishing and deep learning concepts, and reviews state-of-the-art techniques, highlighting their advantages and drawbacks. The paper discusses challenges faced by deep learning in phishing detection and suggests future research directions. An empirical analysis evaluates the performance of various deep learning techniques, revealing common issues such as manual parameter-tuning, lengthy training times, and suboptimal detection accuracy, underscoring the need for further research in this domain.

[4] addressed the escalating threat of phishing attacks, which have become a significant concern for internet users, governments, and businesses alike. It proposes a comprehensive overview of machine learning and common phishing techniques employed by cybercriminals to deceive unsuspecting users. Through a survey, the paper identifies phishing emails as particularly effective among targeted sectors and users. To mitigate this threat, the paper discusses the application of machine learning algorithms for more effective phishing detection. A detection model is proposed, leveraging machine learning techniques to classify emails as phishing or non-phishing based on inherent characteristics and other features. Results from comparing different datasets reveal that employing the greatest number of features yields the most accurate and efficient results.

Phishing attacks, employing sophisticated methods like content injection and social engineering, pose significant risks to confidentiality, prompting the development of various detection approaches. Deep learning algorithms have emerged as promising tools in this endeavor, yet existing studies lack a systematic overview of their application in phishing detection. To address this gap,[5] conducted a systematic literature review (SLR) was conducted, examining 43 selected journal articles to synthesize findings on deep learning approaches. Supervised deep learning algorithms were predominantly utilized, with deep neural networks (DNN) and hybrid models demonstrating superior performance. Data sources varied, with URL-related data being the most common. PhishTank emerged as the primary dataset, while Keras and TensorFlow were the preferred deep-learning frameworks. However, challenges persist, underscoring the need for further research in phishing detection methodologies.

The rise of internet usage has spurred e-commerce growth but also introduced security challenges, notably phishing attacks targeting personal and financial data. Detecting such attacks has become increasingly complex, prompting the exploration of anti-phishing machine-learning techniques.[6] conducted a comparative analysis of five machine learning approaches—Decision Tree, Random Forest, KNeighbors, Gaussian Naïve Bayes, and XGBoost—for website phishing detection. Essential features contributing to accuracy were selected, with results indicating that the Random Forest algorithm achieved the highest accuracy of 97.0%, outperforming other methods.

[7] utilized a phishing URL-based dataset extracted from a reputable repository, comprising attributes from over 11,000 websites. Various machine learning algorithms including decision tree, linear regression, random forest, naive Bayes, gradient boosting classifier, K-neighbors classifier, and support vector classifier are applied and designed to prevent phishing URLs. Additionally, a hybrid LSD model, combining logistic regression, support vector machine, and decision tree, is proposed for enhanced protection. Canopy feature selection and Grid Search Hyperparameter Optimization techniques are employed with the LSD model. Evaluation parameters such as precision, accuracy, recall, F1-score, and specificity are utilized to assess the effectiveness of the models. Comparative analyses demonstrate that the proposed approach outperforms other models, achieving superior results in defending against phishing attacks.

[8] investigated the escalating threat of online phishing frauds by assessing various machine learning (ML) algorithms for identifying malicious websites. Given the significant risks posed by phishing attacks, ML emerges as a promising solution due to its adaptability and capacity to analyze vast datasets. Previous research highlights Support Vector Machines (SVM) and Random Forests as effective tools in phishing detection. However, challenges persist in algorithm selection and feature prioritization. The proposed system integrates Gradient Boosting and Cat Boost alongside Random Forest, utilizing features from the UC Irvine Machine Learning Repository. The study's significance lies in its performance analysis, guiding the selection of the most effective algorithm for detecting phishing websites and addressing the evolving landscape of online threats with automated systems.

[9] addressed the significant threat of phishing to web security, where malignant web pages are disguised as genuine ones to steal sensitive information. Traditional techniques for phishing detection, such as Bayesian classification, have shown effectiveness with smaller datasets, achieving up to 90% accuracy. However, these methods struggle with larger datasets due to the exponential growth of web content. One innovative approach proposed the use of hyperlinks from the HTML source code, creating a feature vector with 30 parameters to detect phishing sites. This method involved training a supervised Deep Neural Network model with the Adam optimizer, employing a Listwise approach for classification. The results demonstrated that this deep learning approach outperformed traditional machine learning methods like SVM, Adaboost, and AdaRank, providing more accurate detection of phishing websites.

3. METHODOLOGY

The research attempts to address the problem of developers choosing the right memory size for the serverless function. This introduced an approach to choosing the most favorable memory size for serverless functions using a reference table with differing cost and time tradeoffs.

A. Research Framework

The system flow diagram illustrates the key steps in the project, starting with data collection and preprocessing. In the "Data Collection" phase, datasets are acquired, cleaned, and prepared for analysis. Following this, the "Model Development" phase involves feature engineering to extract relevant features and select appropriate machine learning or deep learning algorithms.

Next, in the "Model Training" phase, the selected algorithms are trained on the preprocessed data to learn patterns and relationships. The "Model Evaluation" phase follows, where performance metrics such as accuracy, precision, and recall are calculated to assess the models. The project concludes with a "Comparison Analysis," comparing the performance of different models to identify the most effective approach for detecting phishing URLs. This ensures the final model is robust and accurate in distinguishing between legitimate and malicious URLs.

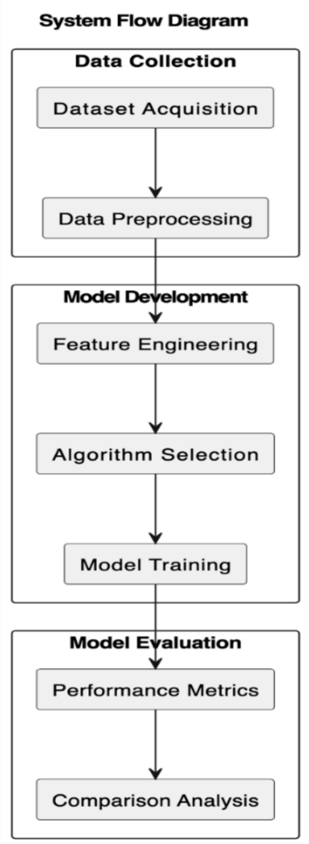


Fig. 1: Research Flow Diagram

B. Data collection

The dataset used in this thesis is a result of merging two separate datasets, each containing information related to phishing detection. The first dataset, named "Web page phishing detection," and the second dataset, named "Phishing Websites Dataset," were combined to create a unified dataset for analysis.[10][11]

The merging process involved retaining only those features that were common to both original datasets. This selective feature inclusion was done to focus on the most relevant data while avoiding redundancy and ensuring consistency across the merged dataset. Features that were not present in both datasets were excluded to maintain data integrity and coherence.

By merging these two datasets, the resulting dataset provides a comprehensive view of the shared characteristics between the original datasets, offering a consolidated and streamlined set of features for analysis. This unified dataset serves as the foundation for conducting comparative analyses of machine learning and deep learning approaches for phishing detection, enabling researchers to explore the effectiveness of different algorithms in identifying phishing URLs and addressing cybersecurity challenges.

C. Dataset cleaning

During the dataset preprocessing stage, two critical activities were conducted to ensure the integrity and quality of the data. Firstly, the dataset was scrutinized to identify the presence of null values or missing data. Detecting null values was essential as they could potentially distort the analysis and modeling outcomes. Any identified null values were addressed using appropriate techniques such as imputation or removal, depending on the nature and extent of the missing data. Secondly, the dataset was examined for duplicates to identify and eliminate redundant entries. Duplicates can lead to biased results and undermine the validity of the analysis; hence their detection and removal were imperative. By performing these preprocessing activities, the dataset was cleansed and prepared for subsequent analysis and modeling tasks. The absence of null values and duplicates ensured the reliability and accuracy of the dataset, laying a solid foundation for conducting robust analyses and evaluations of phishing detection methodologies using machine learning and deep learning approaches.

The dataset was bifurcated into training and testing subsets. The division was realized through the train_test_split method from the sklearn.model_selection suite. An 80:20 ratio was adhered to for the split, allocating 80% of the data to the training subset for model training, and the remaining 20% to the testing subset for model evaluation.

The next phase entailed the standardization of numerical features within the dataset. The StandardScaler from the sklearn.preprocessing library was utilized for this process. It standardized the features to a common scale by subtracting the mean and scaling to unit variance. This standardization was pivotal in ensuring that all features contributed equivalently to the model's predictions and that no single feature with a larger scale unduly influenced the model output.

D. Dataset Analysis

Correlation is a statistical measure that quantifies the degree to which two variables change together, providing insights into their linear relationship. The Pearson correlation coefficient (r) is the most commonly used measure of correlation, ranging from -1 to 1. A value of r = 1 indicates a perfect positive correlation, r = -1 signifies a perfect negative correlation, and r = 0 denotes no linear correlation. The formula for calculating the Pearson correlation coefficient is:

$$r = \frac{n(xy)-(y)(x)}{\sqrt{(nx^2-(x)^2)(ny^2-(y)^2)}}$$

where,

r= Pearson correlation coefficient, which measures the strength and direction of the linear relationship between the variables x and y

n = number of observations

xy = Sum of the products of corresponding values of x and y

y = sum of all y values

x = sum of all x values

x^2 = sum of the squares of all x values

y^2 = sum of the squares of all y values

Box plots divide data into quartiles, highlighting the median, interquartile range (IQR), and potential outliers. The median is marked by a line within the box, the IQR is represented by the box itself, and the whiskers extend to the rest of the distribution, excluding outliers, which are plotted as individual points. For each feature, the box plots contrast phishing and non-phishing categories, revealing insights into central tendency, variability, and skewness. The central tendency shows whether phishing URLs have higher or lower median values compared to non-phishing URLs. Variability is illustrated by the spread of the box and whiskers, with a wider IQR indicating more variability within a feature for a given class. Skewness and outliers suggest asymmetry in the data or the presence of anomalies, which are crucial for identifying phishing URLs that deviate from typical URL patterns.

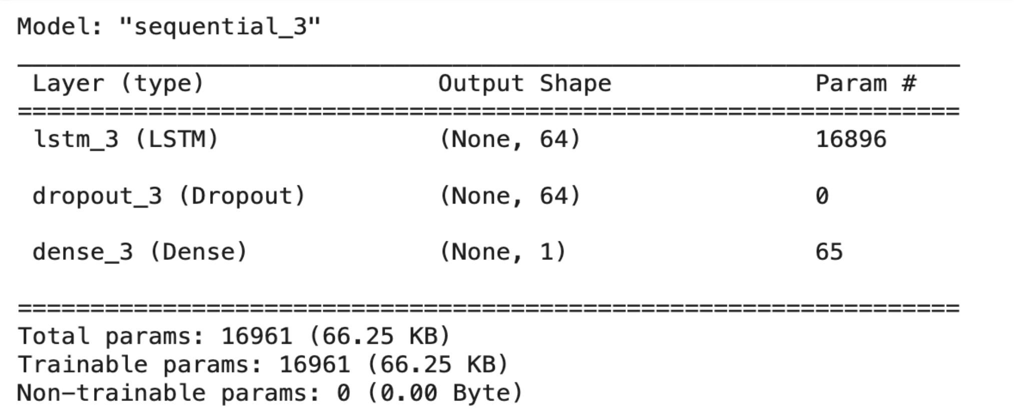
E. Machine Learning Algorithms

Decision Trees are intuitive models that partition the feature space based on attribute values, making them easy to understand and interpret. They recursively split data into branches to form a tree structure, where each node represents a decision based on a feature, and each leaf node represents an outcome. While quick and interpretable, Decision Trees can suffer from overfitting, especially with complex datasets.

Support Vector Machines are powerful algorithms that construct an optimal hyperplane to separate data points into different classes, maximizing the margin between classes. SVMs are particularly effective in high-dimensional spaces and can handle non-linear boundaries through kernel functions. They excel in minimizing false positives but can be conservative, potentially missing some phishing attempts.

Long Short-Term Memory Networks are a type of recurrent neural network (RNN) designed to capture long-term dependencies and temporal patterns in sequential data. LSTMs are well-suited for tasks involving sequential data with variable-length sequences, such as phishing URL detection. They can discern subtle variations and irregularities in data, making them highly effective for detecting complex patterns associated with phishing URLs.

The deep learning model architecture utilized in this study comprised three main layers, each playing a distinct role in the model's computational process.



```
Model: "sequential_3"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 lstm_3 (LSTM)               (None, 64)                16896

 dropout_3 (Dropout)         (None, 64)                0

 dense_3 (Dense)             (None, 1)                 65

=================================================================
Total params: 16961 (66.25 KB)
Trainable params: 16961 (66.25 KB)
Non-trainable params: 0 (0.00 Byte)
```

Fig. 2: LSTM Model Layers and Architecture

4. EVALUATION AND RESULTS

A. Correlation Plot

The bar chart presents the correlation of each feature with the phishing label. The majority of the features exhibit a positive correlation with the likelihood of a URL being phishing, as indicated by the SkyBlue bars. Notably, the feature with the highest positive correlation (significantly taller SkyBlue bar) suggests that as its value increases, the likelihood of the URL being identified as phishing significantly increases. Conversely, the single bar (n_redirection) at the far-right end of the chart signifies a feature that is negatively correlated with the phishing label.

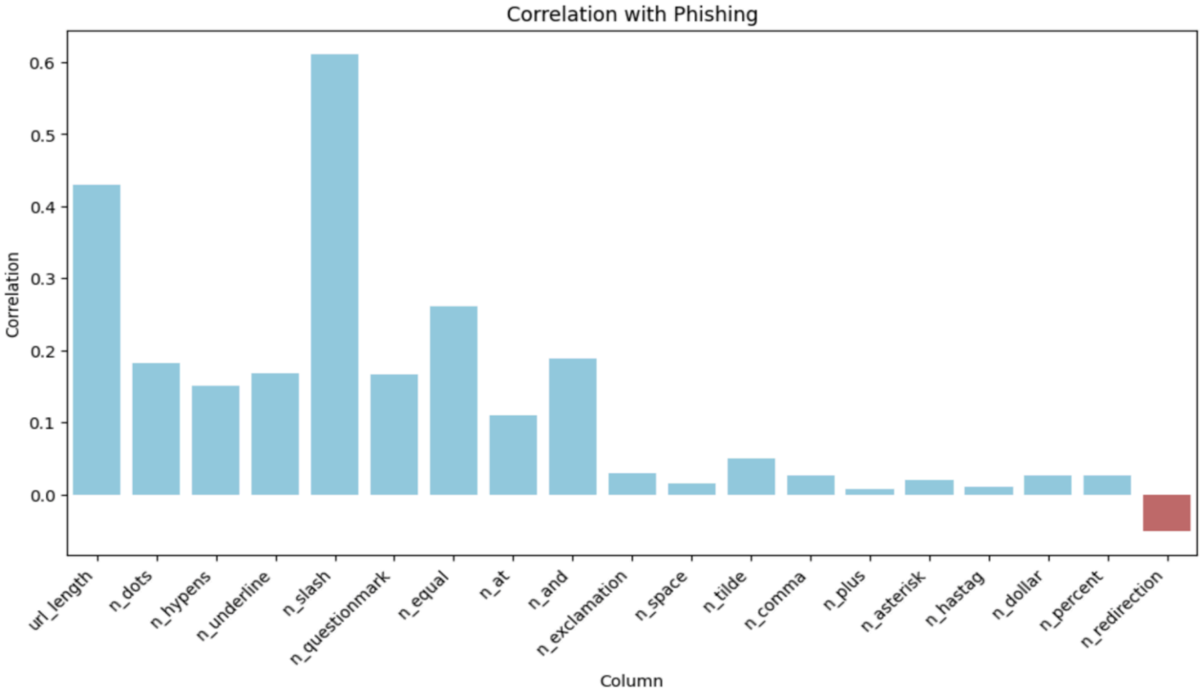


Fig. 3: Correlation Plot for different features

B. Box Plot

For further analysis beyond correlation coefficients, horizontal box plots for each feature were constructed to visualize their distributions across the two phishing categories: phishing (1) and non-phishing (0). This analytical approach provided a visual representation of the central tendency and dispersion for each feature and offered insight into their distributional characteristics contingent upon the URL's class.

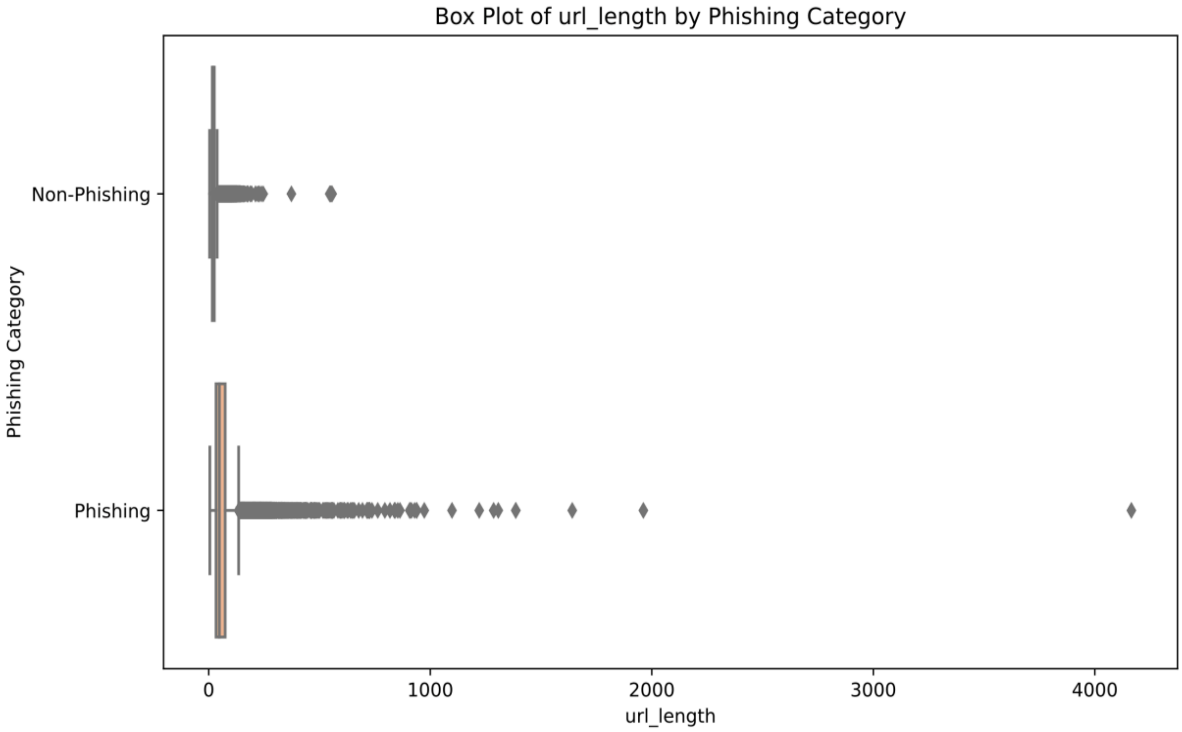


Fig. 4: Box Plot for URL length features

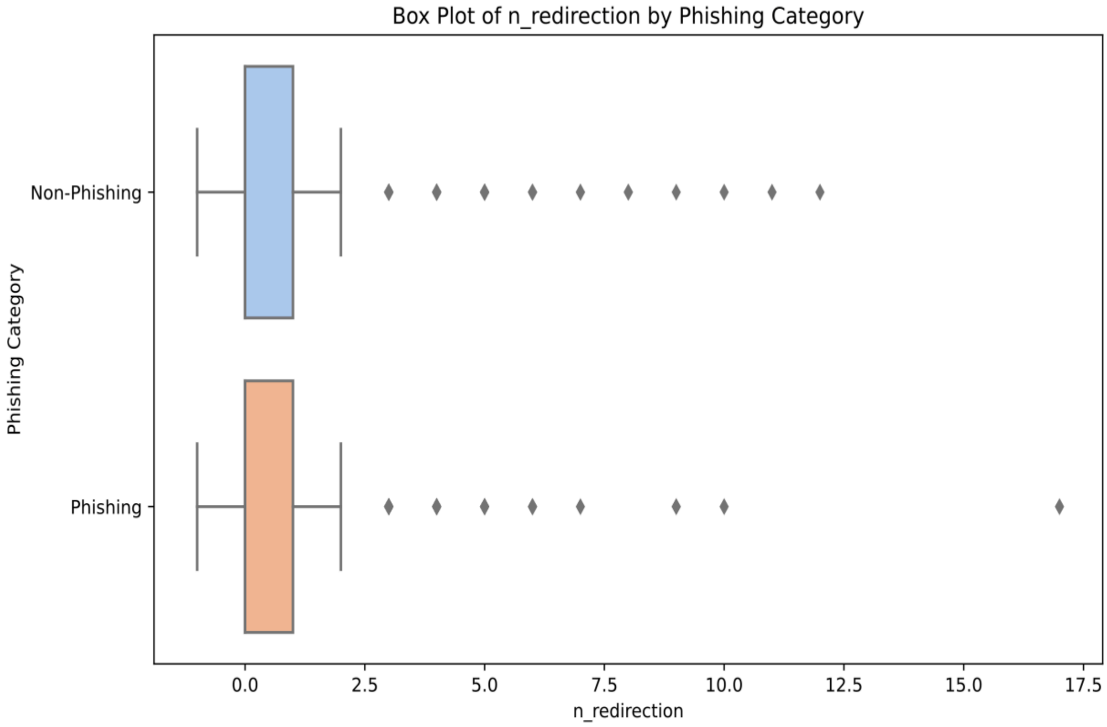


Fig. 5: Box Plot for n of redirection feature

C. Results from Different Algorithm

Decision Tree (DT):

The Decision Tree model was evaluated using a dataset with a focus on phishing URL detection. The best hyperparameters were determined as {'criterion': 'gini', 'max_depth': 7, 'min_samples_leaf': 1, 'min_samples_split': 5}, achieving an accuracy of 0.8785. The classification report revealed a precision of 0.88 for non-phishing URLs and 0.75 for phishing URLs, with corresponding recall values of 0.85 and 0.80. The F1 scores were 0.86 for non-phishing and 0.77 for phishing, highlighting a balanced performance. The confusion matrix indicated 5854 true positives, 10730 true negatives, 1968 false positives, and 1464 false negatives, showing effective identification of both phishing and non-phishing URLs.

Support Vector Machine (SVM):

The SVM model, tuned with a regularization parameter (C) of 100, an 'rbf' kernel, and a tolerance level (tol) of 0.1, performed well in classifying phishing URLs. It achieved a precision of 0.85 for non-phishing and 0.87 for phishing, with recall values of 0.94 and 0.72, respectively. The F1 scores were 0.89 for non-phishing and 0.79 for phishing. The confusion matrix showed 5290 true positives, 11878 true negatives, 820 false positives, and 2028 false negatives, indicating strong performance but with a higher false-negative rate compared to the Decision Tree.

Long Short-Term Memory (LSTM):

The LSTM model's performance was evaluated using training and validation loss over multiple epochs. The convergence of training and validation losses indicated good generalization without overfitting. The confusion matrix for the LSTM model revealed 6357 true positives, 11329 true negatives, 1369 false positives, and 961 false negatives, demonstrating superior performance in detecting phishing URLs with fewer false negatives and false positives compared to the other models. The LSTM's ability to capture sequential patterns in URL data contributed to its effectiveness in phishing detection.

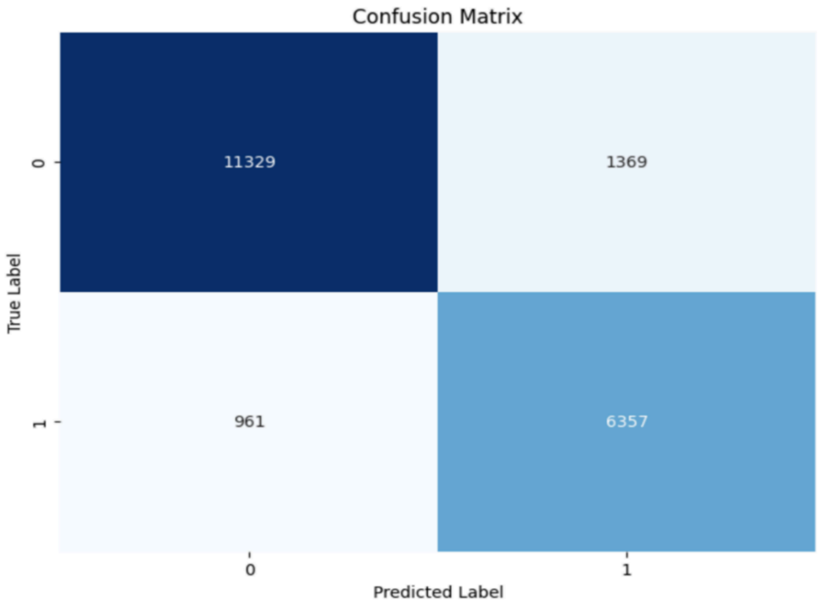


Fig. 6: Confusion Matrix for best model (LSTM)

A. Findings and discussion

This study explored and evaluated the effectiveness of three machine learning models—Decision Trees (DT), Support Vector Machines (SVM), and Long Short-Term Memory (LSTM) networks—in detecting phishing URLs. The comparative analysis revealed that each model has distinct strengths and weaknesses, influencing their suitability for different operational environments.

The Decision Tree model, optimized through hyperparameter tuning, achieved a high accuracy of 0.8785. It showed balanced precision and recall rates, with an F1-score of 0.86 for non-phishing URLs and 0.77 for phishing URLs. The confusion matrix indicated that while the Decision Tree model effectively identified a substantial number of phishing URLs, it also had a moderate rate of false positives and false negatives. This suggests that while the Decision Tree model is competent, it may not be the optimal choice in highly sensitive applications where the cost of misclassification is significant.

The SVM model demonstrated strong precision and recall rates, particularly excelling in identifying non-phishing URLs with a recall of 0.94. Its precision for phishing URLs was slightly higher than that of the Decision Tree, reflecting its robustness in making correct positive predictions. However, the SVM's higher false-negative rate indicates a limitation in consistently identifying phishing URLs. This could be problematic in scenarios where missing a phishing URL could lead to significant security breaches.

Among the models evaluated, the LSTM network emerged as the most effective for phishing URL detection. The LSTM model's ability to capture and learn from the sequential patterns in URL data contributed to its superior performance. It achieved a balance between high precision and recall rates, minimizing both false positives and false negatives. The confusion matrix for the LSTM model showed fewer errors compared to the other models, highlighting its robustness and reliability in distinguishing between legitimate and malicious URLs. This makes the LSTM model particularly well-suited for real-world applications where the detection of phishing URLs is critical to maintaining cybersecurity.

Overall, the findings underscore the importance of selecting the appropriate model based on the specific requirements of the task. While Decision Trees and SVMs offer valuable insights and reasonable performance, LSTM networks provide a significant advantage in handling the complexities of phishing URL detection. This study's comparative analysis not only highlights the strengths of each model but also emphasizes the need for continuous optimization and adaptation of machine learning techniques to keep pace with evolving cyber threats. Future work could explore hybrid models and advanced feature engineering to further enhance the efficacy of phishing detection systems.

5. CONCLUSION

This research has provided comprehensive insights into the capabilities and performance of three distinct machine learning models—Decision Trees, Support Vector Machines (SVM), and Long Short-Term Memory Networks (LSTM)—in the context of phishing URL detection. Among the evaluated models, LSTMs have demonstrated superior performance, offering a balanced approach with low rates of both false positives and false negatives. This balance is crucial in phishing detection, where the cost of missing a phishing attempt can be as severe as wrongly blocking legitimate access.

The study underscored the importance of selecting a machine learning model based on specific requirements and constraints of the phishing detection system. Each model has shown distinct strengths and limitations, influencing their suitability for different cybersecurity scenarios. Decision Trees provide quick and interpretable results but may suffer from overfitting. SVMs excel in high-dimensional spaces with fewer false positives but tend to miss more phishing attempts due to their conservative nature. LSTMs, capable of capturing sequential and contextual information in URL data, have emerged as the most effective in understanding the complex patterns associated with phishing URLs.

Acknowledgments

I am profoundly grateful to my supervisor, Sr. Asst. Prof. Saroj Pandey, for his consistent feedback and invaluable insights, which were instrumental in the completion of this paper. I also extend my sincere thanks to the Department of IT at Kantipur City College for their steadfast support and encouragement throughout this study.

References

[1] Sonam Malviya, “Investigating the Solutions of Phishing Detection Using ML Algorithm,” 2022.

[2] Sonam Malviya, “Phishing Detection Using ML Based URL Classification,” 2022.

[3] Nguyet Quang Do et al., “Deep Learning for Phishing Detection: Taxonomy, Current Challenges and Future Directions,” IEEE Access, pp. 1–1, Jan. 2022, doi: 10.1109/access.2022.3151903.

[4] Ala Mughaid et al., “An intelligent cyber security phishing detection system using deep learning techniques,” Cluster Computing, May 2022, doi: 10.1007/s10586-022-03604-4.

[5] D. Rodriguez et al., “Applications of deep learning for phishing detection: a systematic literature review,” Knowledge and Information Systems, May 2022, doi: 10.1007/s10115-022-01672-x.

[6] Md. Milon Uddin et al., “A Comparative Analysis of Machine Learning-Based Website Phishing Detection Using URL Information,” 2022 5th International Conference on Pattern Recognition and Artificial Intelligence (PRAI), Aug. 2022, doi: 10.1109/prai55851.2022.9904055.

[7] Abdul Gaffar Karim, Mobeen Shahroz, Khabib Mustofa, Samir Brahim Belhaouari, and S Ramana Kumar Joga, “Phishing Detection System Through Hybrid Machine Learning Based on URL,” IEEE Access, vol. 11, pp. 36805–36822, Jan. 2023, doi: 10.1109/access.2023.3252366.

[8] Aman Anand, Aayush Gupta, Abinash Dubey, and K. Chakradhar Naidu, “Phishing Site Detection Using ML Algorithms,” International Journal For Multidisciplinary Research, 2024, doi: 10.36948/ijfmr.2024.v06i02.14744.

[9] L. Lakshmi et al., “Smart Phishing Detection in Web Pages using Supervised Deep Learning Classification and Optimization Technique ADAM,” Wireless Personal Communications, vol. 118, no. 4, pp. 3549–3564, 2021, doi: 10.1007/s11277-021-08196-7.

[10] G. Vrbančič, “Phishing Websites Dataset,” vol. 1, Sep. 2020, doi: 10.17632/72ptz43s9v.1.

[11] A. Hannousse and S. Yahiouche, “Web page phishing detection,” vol. 3, Jun. 2021, doi: 10.17632/c2gw7fy2j4.3.

[12] “Box Plot,” Definitions, doi: 10.32388/lfuqqu.