Privacy and Security With Deep Learning

1st MD Sayem
dept. of Electronics Engineering
Hochschule Hamm-Lippstadt
Lippstadt, Germany
md.sayem@stud.hshl.de

***Abstract*—In our rapidly developing digital world, privacy and security have become critical concerns. We cannot think of a single day without using digital technologies or platforms (e.g., IoT devices, social media platforms, e-commerce, and online payments). As we advance with these technologies, the privacy and security of users and providers are becoming more sensitive and critical. Increasing reliance on digital systems opens more doors for cyber-criminals to penetrate systems illegally and steal valuable information. This includes personal information, login credentials, databases, and financial data. A breach of privacy or security not only jeopardizes individual users but also organizations.Cyber-criminals have developed various methods, such as malware, DDoS, phishing attacks, adversarial attacks, ransomware, SQL injections, and spoofing attacks, to exploit vulnerabilities. This paper focuses on how privacy and security can be enhanced using deep learning techniques to detect phishing websites. More specifically, by implementing a Feed-Forward Neural Network (FNN), the proposed deep learning model analyzes patterns in website structures, URLs, and other content to improve detection and prediction accuracy. A dataset of phishing URLs was taken from Mendeley, and a well-performing deep learning model was selected and from GitHub and upgraded later. Google Colab was used for implementation, and the model achieved 91 percent accuracy.**

I. Introduction

In an increasingly digitalized world, which is highly depended on electronic technology, safegurding data from cyber-criminals and ensuring privacy and security become more challenging. The main purpose of cyber attacks is to break into a system or a device and steal valuable data and harm an user or an organization mostly financially [1].

One of the major sector where maintaining privacy and security is paramount is E-commerce. E-commerce systems allow customers and sellers to trade and manage payments digitally. It is a very attractive traget for cyber-criminals to exploit vulnerabilities and gain unauthorized access like stealing customers credit card information, often leading to a payment fraud [2].

IN 2017, Equifax, a credit reporting agency in United States paid around 700 million US dollars for settlement and fines because a group Hackers managed to breach Equifax's systems and stole personal data, including Social Security numbers, addresses, and credit card information of approximately 147 million US citizen [3] [4].On November 24,2014, Sony experienced a wiper attack from North Korean cyber-criminals and leaked confidential information, emails, and sensitive employee data. The estimated cost of this attack for Sony was 155 to 175 million US dollars [5] [6].

These examples underscore the importance of maintaining privacy and security to prevent personal data leaks to financial loss in our digitalized world.

Cyber-criminals have developed various types of attack to compromise victim's privacy and security.Some of the well known and effective types are : DDoS Attack : used to flood or jam a system with traffic and stop or block the system to respond to a legitimate request.MitM Attack: A Man-in-the-Middle (MitM) attack grants cyber criminal to breach into a system and intercepts communication between a client and a server. Attacker impersonates both side to steal or alter the information being exchanged. Password Attack: The password attack occurs when attacker try to obtain or decrypt password by illegal methods like accessing password database, using tools like Dictionary Attacks and Password sniffers. Malware Attack: it involves by installing malicious software on a user's computer without their consent. Most common type of malware are : Virus, Worms, Trojans and Ransomware. By injecting malware on victim's devices, attackers can gain access of the control of the device. Phishing Attack : it happens when an attacker sends fake emails of fake webpage links that creates a fake environment and pompt victim to input personal information or credentials [7] [8].

Phishing attack is one of the most attractive methods used by cyber-criminals to fool an user to elicit personal information. According to the report of APWG (Anti- Phishing Working Group), the rate of average phishing attack increased 5753 percent per month over the period of 12 years, from 2004 to 2016. In 2015, more than half a billion personal records were stolen by this attack [9].

Deep Learning techniques can be used To reduce the risk of this phishing attack. By training a deep learning model with a large number of datatset, containing phishing and legitimate URLs, the model can analyze their structure such as length, presence of special characters in the URL, presence of IP address, pop-up window, age of the domain etc and predict or detect the phishing sites.

[1]

II. Background and Related Work

The Phishing Attack was first introduced by a group of hackers in 1996, who stole America Online (AOL) accounts by tricking AOL user into giving their passwords [10].

There are various types of phishing attacks that cybercriminals are using to trick the user.The types are [11]:

1) Algorithm-Based Phishing
2) Deceptive Phishing
3) URL Phishing
4) Hosts File Poisoning
5) Content-Injection Phishing
6) Clone Phishing
7) Spear phishing

Among all those attacks URL based phishing is hard to identify because attackers always change their techniques [12].

Numerous researches and developments were conducted on the phishing attack and a various types methods and techniques have been developed to prevent this. An organization, Anti-Phishing Working Group (APWG) was founded in 2003.

There are a number of effective phishing URL checker website available in online including CheckPhish by Bloster AI, Scam Detector and URLVoidetc.

Developers have been developing several machine learning based models to detect phishing URLs by analyzing various features of URLs including URL structures, domain attributes, and content patterns to predict whether an URLS is legitimate or malicious.Algorithms like Decision Trees, Support Vector Machines (SVMs) and ensemble methods were implemented and have shown effective result in identifying phishing URLs [13] [14] [15].

Modern web browsers like Chrome, Firefox, Edge use the blacklist of known fraud URLs and give warnings to the user while an user is trying to visit a potential phising URL [16].

The aim of this paper is to detect phishing URLs using a deep learning approach. The goal is to develop a DL model and with a large dataset, the model will be trained to determine URLs as legitimate or phishing .Ultimately the research intension is to contribute to maintain privacy and security by detecting phishing URLs with deep learning approach.

III. Methodology

The methodology for detecting phishing URL using Deep Learning techniques requires three main stages including Data Preprocessing, Model Design and Evaluation. Initially the dataset is analyzed by it's structural understanding. A quick checking for any null values in the datset and removing any irrelevant components is done to achieve higher performace.

The dataset is splitted into three parts for effective model development and evaluation. 80 percent of the data is utilized for training the model, 10 percent data is for validation and the remaining 10 percent is for testing.

[1]Dataset link : https://data.mendeley.com/datasets/6tm2d6sz7p/1

A. Dataset

The Dataset contains 247950 rows indicating the number of analyzed URLs of which, 128541 are from phishing URLs and 119409 are from legitimate URLs and 41 collums for each rows of 41 different kinds of attributes of the URLs.

SN	Feature	Description	Type
F0	Type	Indicating the type of the URL. It is a Boolean feature with 0 representing a legitimate URL and 1 representing a phishing URL.	Boolean
F1	url_length	Representing the number of characters in a URL, including the domain name, path, and any query parameters.	Numeric
F2	number_of_dots_in_url	Indicating the number of dots (".") in the UR	Numeric
F3	having_repeated_digits_in_url	A Boolean feature that denotes whether the URL has repeated digits (e.g., 2232)	Boolean
F4	number_of_digits_in_url	Representing the number of digits (0-9) in the URL.	Numeric
F5	number_of_special_char_in_url	Indicating the number of special characters (e.g., ", #, $, %, &, ~) in the URL.	Numeric
F6	number_of_hyphens_in_url	Representing the number of hyphens ("-") in the URL.	Numeric
F7	number_of_underline_in_url	Indicating the number of underscores ("_") in the URL.	Numeric
F8	number_of_slash_in_url	Representing the number of forward slashes ("/") or backward slashes ("\") in the URL.	Numeric
F9	number_of_questionmark_in_url	Indicating the number of question marks ("?") in the URL.	Numeric
F10	number_of_equal_in_url	Representing the number of equal signs ("=") in the URL. It is a numeric feature	Numeric
F11	number_of_at_in_url	Indicating the number of at symbols ("@") in the URL.	Numeric
F12	number_of_dollar_sign_in_url	Representing the number of dollar signs ("$") in the URL.	Numeric
F13	number_of_exclamation_in_url	Indicating the number of exclamation marks ("!") in the URL.	Numeric
F14	number_of_hashtag_in_url	Representing the number of hashtags ("#") in the URL.	Numeric
F15	number_of_percent_in_url	Indicating the number of percent signs (%) in the URL.	Numeric

Fig. 1. An Overview Of The Dataset

[2]

Fig :1 showing the first 15 types of attributes of the dataset. These are mainly patterns and characteristics of the URLs that can indicate an URL as Phishing Or Legitimate.

The first feature, Type is the target variable indication of whether the URL is Legitimate or Phishing. The Data Type for this attribute is Boolean, 0 = Legitimate and 1 = Phishing.

url length: Representing the number of characters in a URL, including the domain name, path, and any query parameters and the data type is numeric.

similarly all the 41 features will be used during training so that the model can identify the patterns of typical phishing URls. By analyzing and indetifying the pattern the model will be able to predict whether a new URL is phishing or legitimate.

```
data=pd.read_csv('Dataset.csv')
total_nulls = data.isnull().sum().sum()
print(f"There are {total_nulls} null values in the dataset.")
```

```
There are 0 null values in the dataset.
```

Fig. 2. Cheching For Null Values

Fig: 2 shows that the Dataset contains no null value and it is efficient for training.

B. Deep Learning Model

Primarly a Feed Forward Neural Network (FNN) model is selected from github .

The original model was configured to allocate 75 percent of the data for training and 25 percent for validation. But the model had no testing phase.

[2]Github link: https://github.com/shaheerAlam1/detecting-phishing-websites- using-deep-learning

Several updates have been made to enhance the model. The data is splitted as 80 percent for training, 10 percent for validation, and 10 percent for testing.

Additionally, I have used a different and a larger dataset compared to the original model for training, validation, and testing to evaluate the model's performance.
[3]

C. Framework

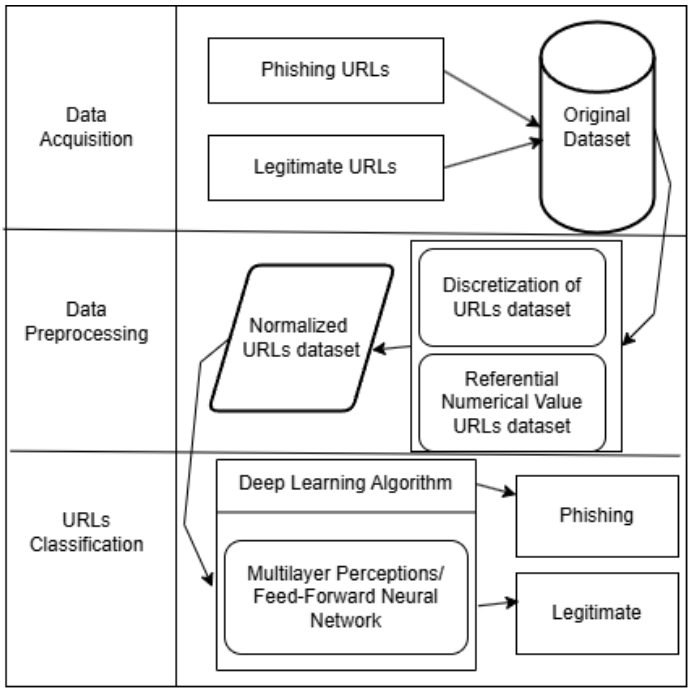

Fig. 3. The Framework For Phishing detection

Fig: 3 represent the main framework of the deep learning model to detect phishing URLs

- Data Acquisition involves complete process of collecting required data that will then be used to classify different URLs under two categories, Phishing or legitimate. This dataset consists of 41 attributes of the web pages, such as: URL length, number of dots in the URL, presence of repeated digits in the URL, number of digits in the URL, number of special characters in the URL, number of hyphens in the URL, number of underscores in the URL, number of slashes in the URL and etc, as for deep learning applications, is really critical because it affects the overgeneralization quality with which the model will be able to carry out its task. The original set is a raw site area that may consist of possible mixtures of the phishing sites, and legitimate sites. The results are appropriately labeled and balanced datasets to ensure deep learning model training.
- Data Preprocessing includes several key steps:

  Discretization: This refers to the process of conversion of raw features such as categorical data or string text into numerical values, thus making them interpretable to the deep learning model.
  Conversion to Referential Numerical Value: The attributes of these websites get mapped onto standard numerical values which are easy for both interpretation and consistency.
  Normalization: The process brings the data values into a uniform range- between say, 0 and 1. Teaching efficiency gets improved in this step and stability of the deep learning model improves in the end.
- URL Classification : The preprocessed data is then fed into a deep learning algorithm.
  Multilayer Perceptron (MLP), which is class of feed-forward neural network is used.
  In final step, the model is able to classify the URLs as phishing or legitimate.

D. FeedForward Neural Networks

The most general class of artificial neural network types is the Feedforward Neural Network (FNN). Feedforward Neural Network is the foundation of various advanced architectural systems in deep learning. Fig:4 is a graphical representation of FeedForward Neural Network It contains an input layer directing feed into one or more hidden layers, on to an output layer with no cycles or feedback between them. Therefore, it progresses in one direction from input to output, with data flowing through it. This simplicity makes the FNNs very easy to understand and use. They are very widely popular in most applications, including a range of machine learning tasks such as classification and regression. [17] [18]

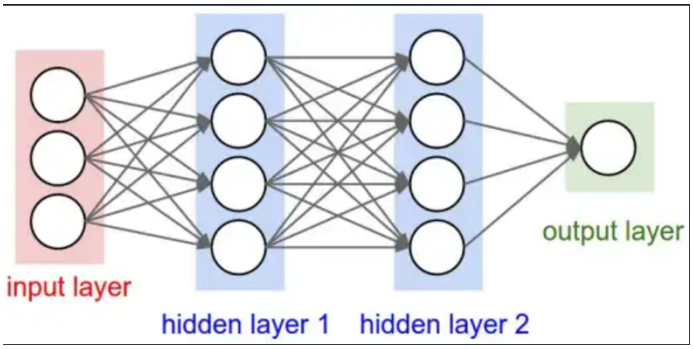

Fig. 4. Feed-Forward Neural Network

The dataset is consists of tabular data with numerical features, for example, the amount of special characters in the URL or the length of the URL. Feedforward Neural Networks are effective for such a highly structured dataset, since they would extract features independently, not processing what their location or sequential relationships are, unlike Convolutional Neural Networks (CNNs), which are suitable toward image data, or Recurrent Neural Networks (RNNs), which handles sequential data. .The main independence of these features in the phishing dataset also sets them apart from each other and

[3]This is the text of the footnote.

their dependencies are such that no complex dependencies exist. In such an environment, FNNs have been developed to learn the direct and indirect ways of relating the features with the different target labels. The architecture is simple enough to model such relationships without unnecessary complication. [17] [18] [19]

The three components of the FNN are the input layer, hidden layers, and finally the output layer. The input layer receives raw data features from the dataset which comprise URL lengths, number of dots in the URL, and some other attributes. Each neuron in the input layer corresponds to a specific feature, thereby passing that data into the network.

Hidden layers perform major computation. This weighted sum, bias term, and a non-linear activation function (for example, ReLU, Rectified Linear Unit) are applied to each neuron in a hidden layer. The transformation enables the network to model complex non-linearity in the data as successive transformations. In hidden layers, batch normalization and dropout are often used so as to improve training stability and to prevent overfitting.

The output layer has the final predictions. In binary classification problems such as identifying a phishing URL from a legitimate one, the output layer usually contains a single neuron. The activation function would be a sigmoid; hence it will produce a probability, which can be considered the chance for the input to be phishing or legitimate.

The selected model here is based on multilayer perceptron (MLP). Multilayer perceptron is a class of feedforward neural networks that has multiple layered of neurons between input and output layers. MLP is a very efficient supervised learning algorithm. Because it learns to map the input data to output labels by adjusting weights and biases during training. For phishing detection, MLP is useful as it will learn non-linear decision boundaries for the various attributes of the URLs in distinguishing legitimate from phishing websites. [20] [21]

The model uses backpropagation as the main process for training. The process of Backpropagation starts by calculating the error between the model's predictions and the actual results using a loss function (binary cross-entropy).Then The model runs backwards from layer to layer to determine the weights that has led to the error. The weights are then adjusted by the optimizer Adam to reduce the error and make the model more precise. It allows the model to learn from the mistake as well.

E. Validation

.

Best Validation Loss : In Fig: 5, Validation loss indicates how accurately model predictions correspond to actual target values during validation. The lesser the loss, lesser are errors incurred by the model. Here in this case, best validation loss shows the numeric value 0.188, thus indicating that model optimized itself to reduce errors without overfitting data.

Best Validation Accuracy: The validation accuracy represents how often the model makes correct predictions on the validation set. A high accuracy means the model is capable of separating phishing and legitimate websites. In this case, the accuracy of 92.5 percent indicates that the model has correctly predicted 92.5 percent of the URL samples of validation set.

```
--- Best Results ---
+--------------------------+---------------------+
|          Metric          |        Value        |
+--------------------------+---------------------+
|   Best Validation Loss   | 0.18840599060058594 |
| Best Validation Accuracy |  0.9254285097122192 |
|       Best Recall        |  0.9025207161903381 |
|      Best Precision      |  0.9493273496627808 |
+--------------------------+---------------------+
```

Fig. 5. Validation Result

Best Recall: In this case, Recall indicates the ability of the model to correctly identify phishing websites. A recall score of 90.2 percent means that out of all the actual phishing websites, the model successfully identified 90.2 percent of them.

Best Precision: Precision measures how many of the predicted phishing websites are originally phishing.A score of 94.9 percent 1 ensures that nearly 95 percent of the websites the model detects as phishing are indeed malicious. This makes the model both reliable and trustworthy for practical use.

F. Evaluation Metrics

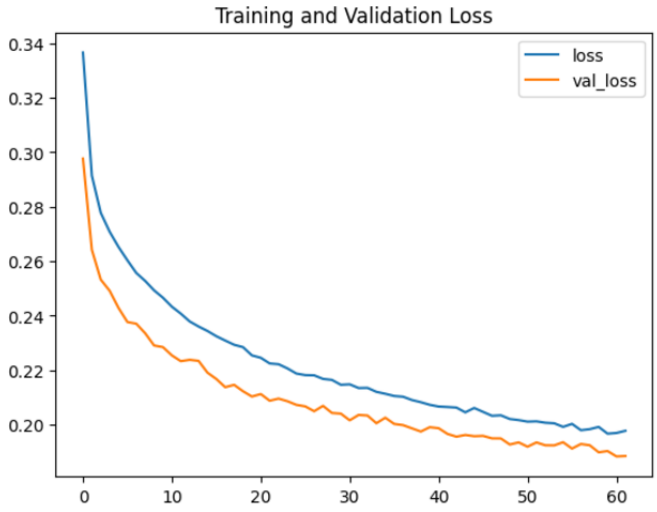

Fig. 6. Training And Validation Loss

Training And Validation Loss:The graph in Fig: 6 shows the loss values for the training and validation data. In this case, loss is implication of the distance between the prediction and the actual results produced by the model. In the initial stages, loss is very high, followed by a rather consistent decrease as time progresses and the model learns. The training loss (blue line) and validation loss (orange line) smoothly decrease and are very close as well; this indicates a very good model learning that can predict unseen validation data without overfitting.

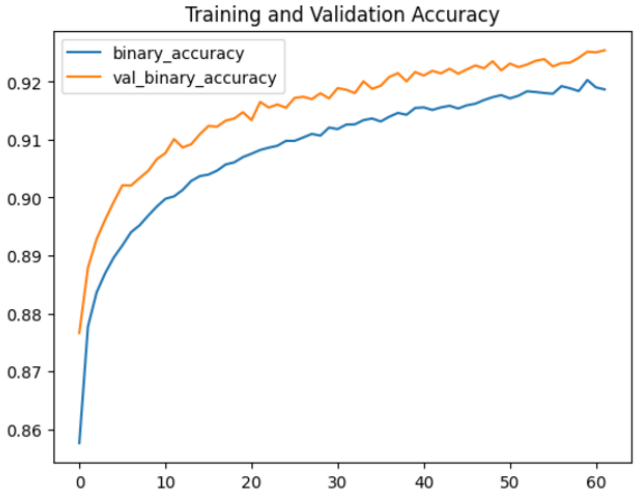

Fig. 7. Training And Validation Accuracy

Training and Validation Accuracy: This second graph in Fig:7 represent the accuracy of the model regarding both training as well as validation data. Accuracy describes the percentage of correct predictions. The training accuracy (blue line) continues to improve but does so slowly over time. Validation accuracy (orange line) increases and stays slightly above the improvement. This indicates that the model is performing quite well both on the known data (training) and the new data (validation), showing that it can be trusted to detect phishing websites

G. Test Results

The result shown in Fig: 8 is from the last 10 percent of the data set that was reserved for test purposes after model training and validation. Thus, the testing is performed using unseen data, providing clarity in how well the model generalizes to new examples.

```
--- Test Results ---
+----------------+----------------------+
|     Metric     |        Value         |
+----------------+----------------------+
|   Test Loss    | 0.19609327614307404  |
| Test Accuracy  |  0.919338583946228   |
| Test Precision |  0.9351308941841125  |
|  Test Recall   |  0.8945649266242981  |
+----------------+----------------------+
```

Fig. 8. Test Result

The test loss is at 0.196, meaning the model has predictions very close to the actual results, reflecting good fit. Less values of loss show better performance of the model.

The Test Accuracy is 91.93 percent meaning that the model correctly identifies phishing and legitimate websites from about 92 percent of the test data. The high accuracy here indicates that the model is very reliable in detecting differences between phishing and legitimate websites.

The Test Precision is at 93.51 percent, which means that such test results reflect how many phishing websites this model has identified out of all phishing websites predicted. Therefore, it has very high precision itself since a smaller false positive ratio means higher chances for legitimate websites not being wrong-labeled as phishing.

The Test Recall is standing at 89.45 percent, which indicates that most phishing websites can be detected by the model out from the test data. Although it is slightly lower than precision, this value confirms that the model has a high capturing ability for actual phishing websites.

H. Confusion Matrix For Test Data

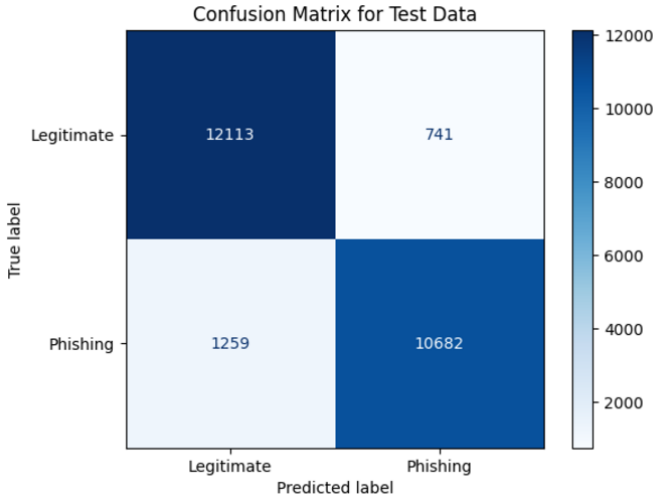

Fig. 9. Confusion Matrix

This confusion matrix in Fig:9 evaluates the model's performance. It shows that the model correctly identified 12,113 legitimate cases (True Negatives) and 10,682 phishing cases (True Positives). It also illustrates that the model has misclassified 741 legitimate cases as phishing (False Positives) and failed to identify 1,259 phishing cases (False Negatives). The overall accuracy is 91.65 percent, with a precision of 93.51 Percent. It indicates that the model is effective at minimizing false alarms. The recall of 89.45 percent, showing it detects most phishing cases but misses a few.

I. Important features (Based On Model Weights)

This graph in Fig:10 illustrates the top 10 most significant features of URLs identified by the Feedforward Neural Network (FNN) for detecting phishing websites.The ranking is based on their contribution to model predictions. The scores of importance derived from model weights. The feature "urllength" is the most influential, longer URLs are indicating strong connection with phishing attempts. Other critical features include "domainlength" and "numberofsubdomains," which implies with these characteristics, phishing URLs are

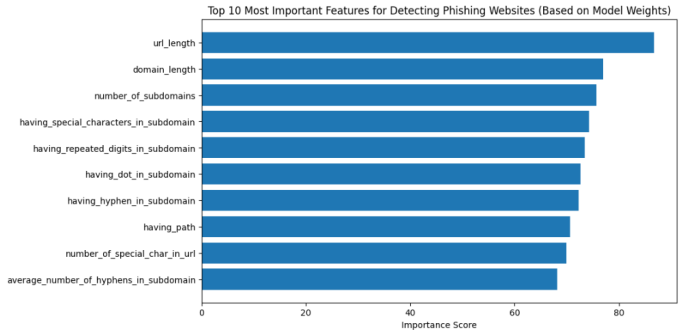

Fig. 10. Important Features Of the URLs

often structured. Features like "havingspecialcharactersinsubdomain" and "havingrepeateddigitsinsubdomain" reflect the tendency of phishing URLs.

IV. Conclusion

In our digitalized world maintaining privacy and security is becoming a challenge. Cyber criminals are creating a lot of ways to breach in to someone's privacy and security and phishing attack is very common of them. The paper explores how deep learning techniques can be utilized for enhancing privacy and security by identifying phishing URLs from URLs attributes using a Feed Forward Neural Network (FNN). The model has pleasantly acquired 91 percent of accuracy in testing. This research focuses on contributing to privacy and security by preventing criminals from stealing personal information and doing online fraud. The result implies the deep learning techniques as a powerful tool in protecting users from cyber threats.In future, the model could be more improved and expanded to acquire more accuracy. A dedicated software can be developed that integrates with the web browser, email systems or other digital platforms to provide realtime phishing detection for the user.The use of more complex models like Convolutional Neural Networks (CNNs) or Recurrent Neural Networks (RNNs) could further enhance the performance by capturing more complex patterns in the data.

References

[1] Y. Li and Q. Liu, "A comprehensive review study of cyber-attacks and cyber security; Emerging trends and recent developments," Energy Reports, vol. 7, pp. 8176–8186, 2021.

[2] G. J. Udo, "Privacy and security concerns as major barriers for e-commerce: a survey study," Information Management Computer Security, vol. 9, no. 4, pp. 165-174, 2001.

[3] J. Craig and J. Kovacic, "Equifax Data Breach 2017," 2024.

[4] M. Bond, K. Human, and N. Kwon, "Analysis and implications for Equifax data breach," 2022.

[5] L. Chen, E. Drannbauer, A. Ademoroti, and D. H. Williams, "How plausible is North Korea's involvement in the Sony Pictures hack? A comprehensive analysis,".

[6] C. Sullivan, "The 2014 Sony hack and the role of international law," Journal of National Security Law Policy, vol. 8, p. 437, 2015

[7] J. M. Biju, N. Gopal, and A. J. Prakash, "Cyber attacks and its different types," Int. Res. J. Eng. Technol., vol. 6, no. 3, pp. 4849–4852, 2019.

[8] M. Uma and G. Padmavathi, "A survey on various cyber attacks and their classification," Int. J. Netw. Secur., vol. 15, no. 5, pp. 390–396, 2013.

[9] K. L. Chiew, K. S. C. Yong, and C. L. Tan, "A survey of phishing attacks: Their types, vectors and technical approaches," Expert Syst. Appl., vol. 106, pp. 1–20, 2018.

[10] B. B. Gupta, A. Tewari, A. K. Jain, and D. P. Agrawal, "Fighting against phishing attacks: state of the art and future challenges," Neural Comput. Appl., vol. 28, pp. 3629–3654, 2017.

[11] M. N. Alam, D. Sarma, F. F. Lima, I. Saha, S. Hossain, et al., "Phishing attacks detection using machine learning approach," in Proc. 2020 Third Int. Conf. Smart Syst. Inventive Technol. (ICSSIT), 2020, pp. 1173–1179.

[12] S. Mishra and D. Soni, "Smishing Detector: A security model to detect smishing through SMS content analysis and URL behavior analysis," Future Gener. Comput. Syst., vol. 108, pp. 803–815, 2020.

[13] R. Basnet, S. Mukkamala, and A. H. Sung, "Detection of phishing attacks: A machine learning approach," in Soft Computing Applications in Industry, Springer, 2008.

[14] G. Mohamed, J. Visumathi, M. Mahdal, J. Anand, and M. Elangovan, "An effective and secure mechanism for phishing attacks using a machine learning approach," Processes, vol. 10, no. 7, p. 1356, 2022.

[15] S. Hossain, D. Sarma, and R. J. Chakma, "Machine learning-based phishing attack detection," Int. J. Adv. Comput. Sci. Appl., vol. 11, no. 9, 2020.

[16] K. Chatzidrosos, "Security and privacy in the web through browser technology," 2024.

[17] G. Bebis and M. Georgiopoulos, "Feed-forward neural networks," IEEE Potentials, vol. 13, no. 4, pp. 27–31, 1994.

[18] D. Svozil, V. Kvasnicka, and J. Pospichal, "Introduction to multi-layer feed-forward neural networks," Chemometrics and Intelligent Laboratory Systems, vol. 39, no. 1, pp. 43–62, 1997.

[19] M. H. Sazlı, "A brief review of feed-forward neural networks," Communications Faculty of Sciences University of Ankara Series A2-A3 Physical Sciences and Engineering, vol. 50, no. 1, 2006.

[20] A. C. Cinar, "Training feed-forward multi-layer perceptron artificial neural networks with a tree-seed algorithm," Arabian Journal for Science and Engineering, vol. 45, no. 12, pp. 10915–10938, 2020.

[21] S. R. S. Tabib and A. A. Jalali, "Modelling and prediction of internet time-delay by feed-forward multi-layer perceptron neural network," in Tenth International Conference on Computer Modeling and Simulation (UKSim 2008), 2008, pp. 611–616.