DOI:

Journal of Computers and Intelligent Systems

Vol. 3, No. 1, 2025, pages 58 - 80

Journal Homepage:

https://journals.iub.edu.pk/index.php/JCIS/

Detection of Phishing Attack by using LightGBM&Xgbost

Ansar Munir Shah[1], Muhammad Noman[1], Talha Farooq Khan[1]

[1]Department of Computer Science, University of Southern Punjab, Multan, Pakistan

ARTICLE INFO

Article History:

Received:	March	21, 2025
Revised:	March	22, 2025
Accepted:	March	26, 2025
Available Online:	March	27, 2025

Keywords:

Cyber Attack
Phishing
LightGBM
Xgbost

Classification Codes:

Funding:

This research received no specific grant from any funding agency in the public or not-for-profit sector.

ABSTRACT

Phishing attacks provide a significant security risk to both individuals and organizations. To steal sensitive information, these assaults are typically carried out by creating phony websites that substantially resemble actual ones. This research employs these phishing attacks, by using AI techniques that have lately been used to look at the URLs of these phony websites. We have proposed the increasing sophistication and frequency of phishing attacks, highlighting the need for an enhanced AI-based model to detect such attacks effectively. Involves LightGBM, Xgbost, and using a hybrid model of a LightGBM, Xgboost classifier to train and test data for detecting phishing attacks on URLs. There are several feature extraction techniques used to detect URL phishing attacks. To identify if a website is a phishing assault or not, these attributes are then given to a LightGBM and Xgboost classifier. As compared to the previous research model's accuracy was 93%, Hence The current proposed results of combining training and testing datasets on LightGBM and XgBoost give a 96% accuracy and improve the quality-of-evaluation metrics of the feature of the URLs to detecting phishing attack detection.

Corresponding Author's Email: talhafarooqkhan@gmail.com

Citation: Talha Farooq Khan

1. Introduction

In the current digital age, phishing attempts have grown to be a significant security threat. In these assaults, people are tricked into supplying personal information or financial data by means of phishing websites or emails. Recognizing and stopping phishing attempts on URLs has become more crucial than ever with the growth of e-commerce and online banking.

IoT requires cyber security since a specific attack or series of attacks could destroy the network or, worst, provide a cybercriminal full access to the entire system – cyber security is critical in the IoT. Internet of things (IoTs) devices are used to hold extremely sensitive data in many areas such as military defense operations.

Hackers have the ability to inspect data or perform network maintenance if they get access to the IoT through a faulty network point or vulnerable device. Cyber-attacks are a constant threat to us. Vulnerabilities, cyber-attacks, data theft, and other threats associated with IoT devices exacerbate the need for IoT security solutions.

DL is the process used in artificial intelligence. A subset of machine learning is called deep learning. DL used Artificial Neural Networks (ANNs) that are worked as function and connectivity of neurons like a human. Artificial neural networks and Conventional Neural Networks (CNN) is mostly use in computer vision tasks. Deep learning is significance of cybersecurity and how it may help handle cybersecurity risks. There are many cybersecurity attacks and threats that are detected with the help of Deep Learning. Social engineering is a form of Phishing and most common cybersecurity attack.

We can observe recent developments in sensing, processing, and connectivity in numerous smart grid applications in today's globe. As a result, communication has become increasingly reliant on network-based sensors, and in the meantime, the energy grid is vulnerable to fake data injection (FDI) assaults, which can circumvent bad data mechanisms and cause real-time problems. Dealing with cyber-attacks does not appear to be faultless in this cyber environment. An AI architecture is employed in the suggested strategy to detect the injected erroneous data measurement. To successfully estimate system variables, this time-series anomaly detector uses a LightGBM, XgBoost, as a hybrid model detect URL-based phishing detection on achieving high accuracy to make a user familiar and confident. Network security concerns are becoming more crucial as the Internet and computer technologies advance continuously. Phishing attacks cost Internet users, financial institutions, and e-commerce businesses a lot of money because they trick consumers into giving over their private information by utilizing bogus websites (Novel Phishing Website). Due to our increased usage of social networks and the Internet, face-to-face connection has largely been displaced by online communication in our daily lives. Due to its accessibility, dependability, and speed, Email is one of the most widely used methods of communication in business and government. As the number of people using email increased, spam emails one or more unwanted messages that appear to be advertising or promotional materials for debt relief programmers, get-rich-quick schemes, online dating, health-related products, etc. were quickly increasing. (Abdul Nabi & Yaseen, 2021).

There are several methods for spotting and avoiding bogus websites. They are divided into two categories by search and categorization methods. When conducting online transactions, the lookup techniques primarily keep blacklisted known fake website URLs. If the transaction URL and the blacklists are the same, the transaction is canceled. The drawback of this method is that it may be challenging to obtain the most accurate findings. If the blacklist has a significant latency time and fraudulent websites have a short lifespan (Lakshmi et al., 2021).

Phishing:

Phishing is a sort of cyberattack that seeks to get sensitive data by impersonating a reliable institution, such as credit card details, login passwords, and other private data. Attackers frequently persuade victims to browse a malicious site or install a malicious file via email, instant messaging, or social media.

Types of Phishing:

Phishing can take many different forms, including:

Email phishing the attacker sends an email posing as a trustworthy entity and tricking the victim into malicious attachment getting clicking or downloaded on a link of malicious, Spear phishing, when an attacker directly targets a person or a small people of groups it is known as targeted phishing, Whaling, a style of spear phishing that targets important decision-makers and top-level business leaders, Clone phishing to deceive the victim into clicking on a dangerous link, the attacker crafts an email that is an exact clone of an official one, SMS phishing in an attempt to deceive the victim into clicking on a dangerous link, the attacker sends a text message pretending to be a reliable source, Detection Methods there are several types of phishing detection methods, including, Technical Methods: these include using firewalls, intrusion detection systems, and anti-phishing software, User education an essential component of phishing detection is teaching users how to recognize and prevent phishing attempts, Domain-based Message Authentication, Reporting & Conformance (DMARC).

A procedure that checks the legitimacy of communications aids in preventing email-based phishing attempts, Brand protection services these services keep an eye out for fraudulent or harmful websites online and notify businesses when their brands are being utilized in phishing scams, URL scanning tools that examine URLs to identify whether or not they are dangerous.

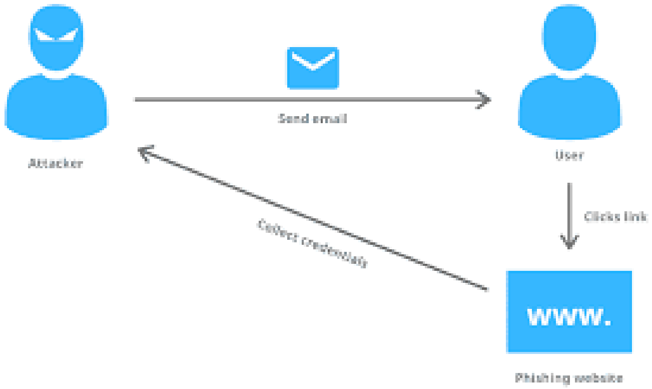


Figure 1 (Threat Finding)

Internet of Things and Its Attacks (IoT):

The "Internet of Things" is a network of interconnected, Internet-connected gadgets that gather and transmit data across a wireless network without human assistance (IoT). No matter where they are, real-world things may be integrated and used thanks to IoT. The steps an attacker takes from early reconnaissance and identification through mission completion are described by the cyber-attack lifecycle. This assists us in comprehending and thwarting malicious actors, ransomware, and other threats.

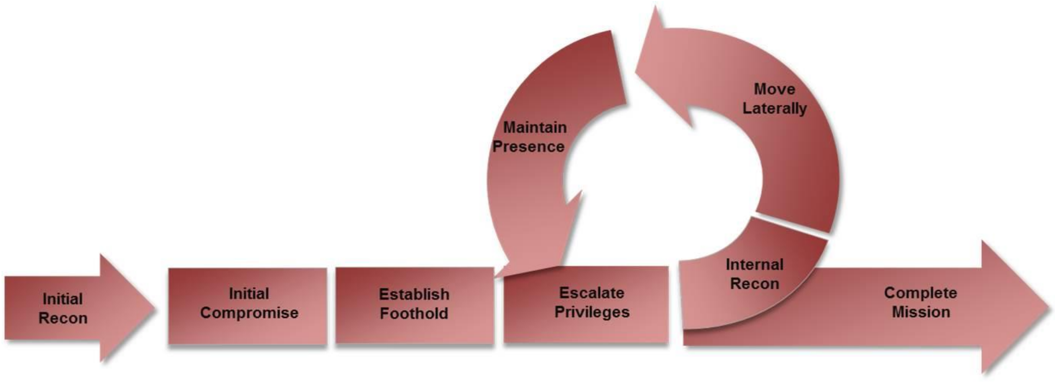


Figure 2 (Cycle of IOT)

For network administration and performance monitoring in such a situation, privacy and security approaches are extremely important and difficult.

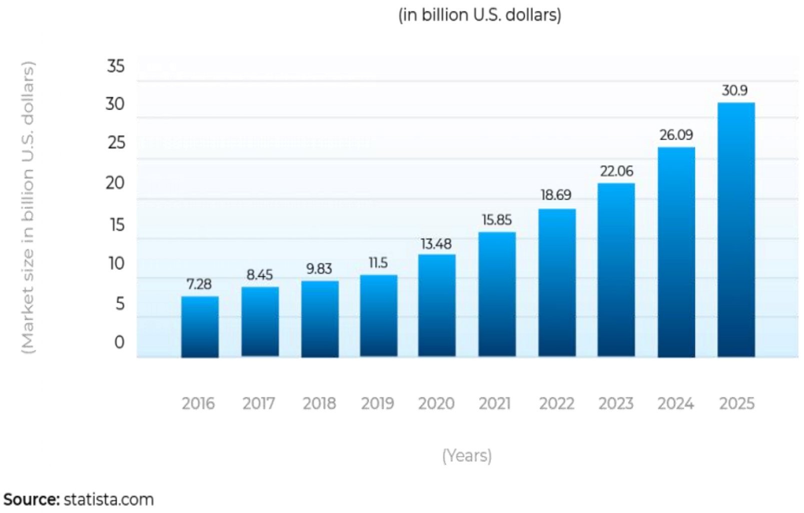


Figure 3 (IOT Overview)

The stats of IOT attacks are measured from https://intersog.com/blog/iot-security-statistics/

Background:

An example of a phishing assault is URL phishing, which uses a fake or malicious URL to deceive people into disclosing personal information or downloading malware. The attacker makes a phony website that mimics a real website, such as a bank or social media platform, and then sends an email or message with the bogus URL. When the receiver opens the link, they're taken to a fake website where they're prompted to enter personal information, including their login credentials or credit card information. Attacks using URL phishing have advanced and become more convincing, posing a greater threat to both people and businesses. When directing people to a phony website even when they enter the proper URL, the attacker may employ strategies including making URLs that are strikingly similar to real URLs or utilizing domain name systems (DNS) spoofing. It is essential to warn individuals and organizations about URL phishing attempts and encourage them to take precautions. Utilizing multiple authentication methods, updating software and security systems, and verifying the validity of URLs are a few of these steps before entering sensitive information. Individuals and organizations can lower their chance of falling victim to a URL phishing assault by being watchful and proactive.

Research Gap:

According to the previous literature reviews, better enhanced preprocessing techniques with DL methods can increase the accuracy of the results. Deep learning performed much better than all other models. So, we use a model of a hybrid Convolutional Neural Network (CNN) with multiple classifiers like LightGBM and XgBoost to enhance the accuracy and time complexity of detecting a URL-based phishing attack.

2. Related Work

Yaseen, Q. (2021) proposed the efficiency of spam emails is inserted into the word. BERT (Bidirectional Encoder Representations from Transformers) is used to detect spam emails into the non-spam emails was used to preprocessing technique on the dataset dev data that contain 5569 emails that detect the 745 spam emails into the total emails are trained and tested the emails. BERT is used to find the 96.43% accuracy in detecting the mail.

Lakshmi, L et.al. (2021) Focus on the Bayesian classification system which is used to distinguish between malicious and legitimate online sites. Adam optimizer used 30 parameters to detect malicious web pages. The preprocessing techniques SVM, Adaboost, and AdaRank were used to compare other traditional machine learning approaches. The purpose AdaRank, SVM, and AdaBoost are used to detect 90% accurate Phishing websites.

Wei, B., et.al. (2019) explored fully utilized K-neighbor's and Support Vector Machine (SVM) methods to detect phishing sites. URLs are efficient to detect harmful sites through cyberattacks the phishing attack. Unlike traditional machine learning approaches that require an unambiguous handcrafted attribute selection procedure, Machine Learning professionals can utilize data without the need for cyber security experts' authorization. URLs use three types of dense layers to detect phishing attacks. These layers gradually increase the achievement of the accurate detection rate of threats. The true rate of deduction phishing attacks achieves an 86.63% accurate result.

Ben Fredj, et.al (2020) focused on RNN, multilayer perceptron, and short long-term memory (MLP) because they alert design to expect phishing cyber threats. The planned models be tested on a lately released dataset named CTF, which yielded promising results. CtF'17 represents the overall volume of traffic generated by the game. According to the IP source of the alert, there are 97 attackers using over 32000 distinct source ports to target over 24000 ports on 29 different servers. Threats have been closely monitored by LSTM and RNN, and results have been achieved for LSTM (93.13% vs. 93.35%) and RNN (91.23% vs. 92.90%).

Samy, A., Yu, H., & Zhang, H. (2020) focused on the deep learning method LSTM to use the detection of vulnerabilities and threats. The purposed methods were performed by traditional machine learning methods. This method detects more efficient and saleable performance than the centralized approach. There are five different types of datasets used to detect the fog base attack on network nodes. DL models outperform ML algorithms because of a high number of weights and variables determined, as well as deep structures, and features hierarchy, and a big number of weights and variables. DL models performed 99.2 % in binary classification and 98.27% in multiclass classification for better results.

Al-Abassi, A., et.al (2020) used a model with the Adam optimizer to filter out new representations from unlabeled data using the SAE attack detection model, resulting in various patterns. Secure Water Treatment (SWAT) is used to perform better in the detection of a cyber-attack by using Random Forest, Deep neural network, and Adaboost technique for detecting threats. SAE find the accuracy of the threat 99.67%.

Manimurugan, et al (2020) et.al discussed a crucial security solution for managing network attacks and detecting malicious activity in computer network traffic by using the Deep belief network (DBN) method. They are used to perform deep learning model DBN-IDS for detecting cyber-attacks. CICIDS data sets are used to train and test data and Normal class accuracy was 99.37 percent, Botnet class accuracy was 97.93 percent, Brute Force class accuracy was 97.71 percent, Dos/DDoS class accuracy was 96.67 percent, Infiltration class accuracy was 96.37 percent, Ports can class accuracy was 97.71 percent, and Web attack accuracy was 98.37 percent. They performed better achievement for intrusion detection.

Hindy, H., et.al (2020) discussed artificial intelligence to build intrusion detection systems by helping with ML and DL methods. In IDS, detecting zero attack detection or calculating a false-negative rate about the threat. IDS are used two types of datasets CICIDS2017 and NSL-KDD for the evaluation of the cyberattacks in IoT for using different preprocessing. Deep Convolutional Generative Adversarial Network (DCGAN) techniques for improved performance of detection. Both datasets were different results to achieve their accuracy for detection like NSL-KDD 89–99%, CICIDS2017, 75–98%. These results demonstrate zero-day attack detection in IoT.

Saha, I., et.al (2020) worked on reducing the dimensionality of the data, Primary Components Analysis (PCA) was used to determine principal components. To draw customers, phishers produce imitation websites that appear just like the real thing and send spam emails. Phishers obtain login information when an internet user views bogus web pages as a result of spam. Preprocessing technique deep conventional neural network used to predict the webpages for detecting phishing attack. Relief-FRFE data sets are used to contain information about threats and measure accurate proficiency 95% in training and 93% in testing accuracy measured.

Sriram, S. et.al (2020, July) focused on detecting the botnet attack on IoT devices with the help of a deep neural network and deep learning methods (DNN). Detection of threats on digital media using preprocessing technique support vector machine (SVM), feature extraction, and conventional neural network (CNN) method. To detect attacks emerging from the botnet detection framework analyses connection records of network traffic flows and applies a DL model to infected IoT devices. Using a variety of models, t-SNE datasets are used to train or detect threats. t-SNE datasets are used to train or detect threats by using various models. t-SNE train 777,600.180 data and testing for results about 264.500. SVM is more accurate than the testing results but takes some time to detect.

Wu, Y. et.al (2020) discussed the insider threat of cyberspace. The problem of the cyberattack was detected in both security and data mining communication. Advanced deep learning techniques are used for end-to-end communication for complex data. Deep Learning modules present multilayer structures to present data like RNN, RNN, DFNN, and CNN were considered secret information. Datasets are more important to train and test the data, but no dataset is publicly available for detecting insider threats. The CERT dataset maintains the system log and highlights insider threat activities. The dataset maintains a database of about 1000 real case studies for insider threats. Cyber surveys predict about 25% of attacks by insiders are committed.

Dutta, et al. (2020) discussed threats and attacks that attempt to bypass the security policies of the system. Deep learning offers a lot of potential for building security applications, and it's already been applied in a lot of them. Deep learning provides some examples of typical applications to demonstrate the applicability of the DL approach for the detection of attacks using the NSL-KDD dataset. NSL-KDD is an updated version of the KDDCup99 dataset. NSL KDD is divided into two types KDD train+ and KDD test+ which imitate real-life network environments with unidentified attacks. NSL-KDD was finding the best accuracy in 98% of the testing results. NSL-KDD is more difficult to find results as compared to the KDDCup99 dataset.

(Adebowale et al., n.d.) focused on deep learning-based design and development of phishing detection solutions by using an innovative method called the IPDS combines two methods, LSTM and CNN are used together as a classifier. One million valid phishing URLs were gathered from the Phish Tank and Common Crawl datasets using a hybrid approach. The outstanding classification accuracy of the suggested IPDS was 93.28%.

(Assegie*, 2021) suggested that's model This also classifies URLs in order to detect phishing attacks using a K Nearest Neighbors (KNN) based model. Using 106 observations, the performance of the suggested model for phishing detection was assessed. The proposed model's overall accuracy is 85.08%. The results of an experiment using accuracy metrics as a performance indicator demonstrate the model's efficiency at detecting phishing attacks.

(Ariyadasa et al., 2022) presented PhishDet, a novelty method of phishing website detection employing URL and HTML characteristics, Graph Convolutional Network and Long term Recurrent Convolutional Network, Currently, PhishDet operates effectively. PhishDet gradually picks up HTML and URL content components to defend against attacks of phishing that are constantly changing with a score of 99.53%.

(Karad et al., 2022) presented a well-trained PAD system that supports the installed facial recognition system and significantly reduces the risk of a system-wide security breach caused by Sensor Characteristics, Blink Detection, and Challenge response techniques. Software-based PAD systems that employ CNNs based on deep learning were the obvious step ahead in comparison to hardware-based PAD systems since they are less expensive to deploy and maintain. Research has been carried out to discover quicker and more efficient ways to adopt PAD, and it has been incredibly successful.

(Christy Eunaicy & Suguna, 2022) described the Using deep learning approaches, the model's threat detection is tested during the phases of Data Cleaning, Prediction, and Data Collection for identifying web attacks. Deep learning classifiers are fed the pre-processed dataset to produce the prediction model for the detection of web attacks. The CSIC 2010 dataset's redundant and missing values were eliminated using the preprocessing methods of deep learning, and machine learning, (ANN, CNN, and RNN). In comparison to other techniques, RNN offered a 6% error rate and 94% accuracy.

(Basit & Zafar, n.d.) reviewed malicious URLs may be easily constructed every day, attackers can develop a method to deceive consumers and modify the URLs to look authentic before launching an attack. To identify phishing attacks, DL and

ML techniques are applied. The most popular classification techniques include SVM, RF, DT, k-NN, PCA, and C4.5, More than 95% accuracy was obtained.

(Wan Ahmad, 2020) proposed ML methods SVM, KNN, RF, NB, and DT used for phishing attack detection. The used techniques to classify the two benchmark datasets, the email dataset, and the SMS message dataset, contained the word content that was used to detect phishing attacks. RF performs exceptionally well in terms of average accuracy. The result obtained an average accuracy of 95.6646%.

(Buber et al., 2018) developed a system based on Random Word Detection Module, and Word Decomposer Module (WDM) to identify URLs used in phishing attacks. Natural Language Processing (NLP), and Sequential Minimal Optimization (SMO) were used as preprocessing techniques on the Anti-Phishing Working Group (APWG) dataset. With a success percentage of 97.2%, The Random Forest Algorithm used in the hybrid approach proved to be more efficient than the other examined algorithms.

(Bu & Kim, 2022) presented a deep learning classifier with incorporated genetic algorithms to find the most effective combination of URL feature sets for recall, and perform 10-fold cross-validation. The process of choosing URL characteristics using an evolutionary algorithm can raise the recall of a deep-learning classifier. and on three benchmark datasets, it has undergone cross-validation. Both recall and accuracy went up by 7.07% and 4.13%, respectively.

(J. Lee et al., 2021) presented a multinodular, comprehensive, and adaptable D-Fence phishing email detection technology was developed. The three separate analysis modules are the URL module, structure module, and text module. D-Fence can defend a larger area against attacks than other methods. A real-world workplace email dataset used for evaluations shows that D-Fence has a good detection capacity. With a recall of a false-positive rate of 0.99 and an of 1 in 10K, D-Fence performs effectively.

(Mughaid et al., 2022) developed a detection model using ML approaches that were Legitimate email, phishing email, using three different APWG phishing attack data sets, in order to classify email text as phishing or non-phishing and to validate the results using test data, the training step aims to record inherent properties of the email text and other variables. After comparing them, it was discovered that the most attributes were used to produce the most precise and effective results. The boosted decision tree's accuracy scores for the applied data sets were 0.88, 1.00, and 0.97 successively.

(Feng et al., 2020) proposed deep learning and representation learning based on the Web2Vec model for phishing webpage detection. Using an NLP representation learning technique, the model comprehensively learns the representation of webpages using the URL, page content, and DOM structure. Four trials were conducted to the model's validation of detection impact, and the findings show that the model's overall classification effect is superior to the approaches currently used to identify phishing websites. The model has a 99.05% accuracy rate and an FPR of less than 0.25%.

(Alma & Das, 2020) proposed intrusion detection model using deep learning for web application identification engine to obtain a receiver operating characteristic curve of 1 accuracy, benign and anomalous web searches are used as training data by ECML-KDD dataset. The proposed model used an auto-encoder that can pick up on word sequences and adjust the weight of each word or character accordingly. Obtained Precision: 0.9979 and Recall: 1.00.

(Zhao et al., 2020) presented a method for ensemble learning based on heterogeneous stacking that has been devised to lessen the effect of class imbalance on spam detection in social networks. Module two of his framework is composed of the basic module and the combining module. Increased the learning impact of the base module by using six different learning processes as basis classifiers. The ensemble method was then put into practice using a deep neural network with cost-sensitive learning improvements. The GNB technique's performance is 0.91, compared to the SVM's G-mean value of 0.31, Kappa value of 0.16, and the VM algorithm's false positive rate of 0.81.

3. Used Approach

In this section, authors will discuss the used approach along the designed framework, algorithms and technical details of the modules of the designed system. In this section, authors will discuss the used approach along the designed framework, algorithms and technical details of the modules of the designed system. In this section, authors will discuss the used approach along the designed framework, algorithms and technical details of the modules of the designed system.

After the literature review, the methodology is discussed in this chapter. Research techniques are the process of finding selections, Considering, and analyzing facts about the subject. The research paper's methodology enables the reader to assess the reliability and validity of the study as a whole.

It is usually performed through email. The aim to either infect the victim's machine with malware or steal personal information like credit card numbers and login credentials. To defend oneself from email attacks, everyone should become aware of the well-known cyberattack known as phishing. Due to the attackers' intricate URL formulation, a large number of phishing URLs to users seem to be valid URLs. In this research, a technique for phishing URL detection was suggested. The system was constructed using several strategies to detect phishing URLs and demonstrate the system's resilience. The five key phases of any NLP activity are model evaluation, feature extraction, model training, data preprocessing, and data collection, The methodology consists following steps:

- Datasets
- Importance of data preprocessing
- Methods
- Proposed framework

Dataset:

Three datasets are used to this research, which is downloaded from Kaggle. https://www.kaggle.com/. A data set is a collection of interconnected, unique pieces of interconnected data that can be accessed separately, together, or under a single control. A certain type of data structure is used to organize a data set. The dataset contains more database table and tabular data in each row and column, each row has certain information about the data. The data is in the form of a CSVs file.

Dataset 1:

The first step in training finding the neural network was URL base phishing attack on the web. Phishing is still one of the best and most successful ways for hackers to cheat us out of our money and steal our financial and personal data. The provided dataset has 11430 URLs with 87 extracted features. The dataset is intended to serve as a benchmark for systems that identify phishing using machine learning. There are three different kinds of qualities that stand out: straight from the information on the corresponding pages, 56 directly from URL structure and syntax, and 7 directly from external service inquiries. The dataset is balanced, with an identical 50/50 split between phishing and authentic URLs. The dataset is downloaded from https://www.kaggle.com/datasets/shashwatwork/web-page-phishing-detection-dataset.

Dataset 2:

Finding web-based URL-based phishing attempt was the initial stage in training the neural network. Hackers continue to use phishing as one of the finest and most effective methods to defraud us of our money and steal our personal and financial information. A cooperative clearinghouse for data and information regarding online phishing, Phish Tank. Additionally, Phish Tank offers a free open API for developers and academics to incorporate anti-phishing information into their apps. The 12490 URLs in the supplied dataset are extracted features from online web phish. The dataset is downloaded from https://www.kaggle.com/.

Dataset 3:

This dataset comprises around 87.5K URLs, of which only a third are marked as spam URLs. A binary classification model may be made using it. Various newsletters provide the dataset for each link. As it parses links from more than 100 newsletters every 30 minutes, the flagging system determines whether a link is spam. If a link occurs three or more times in a single newsletter or has a URL that is probably to be used to subscribe or unsubscribe, it is automatically reported. The dataset is downloaded from https://www.kaggle.com/datasets/shivamb/spam-url-prediction.

Proposed System:

The model's architecture obtained four different datasets using data to take input of data. The dataset concern different URLs for obtaining data in binary form using a LightGBM and Xgboost to classify sentences, highlighting its potential value in the detection of spam and phishing. The utility of deep learning is discussed to show how these approaches uncover hidden patterns and unearth important data. Using hybrid LGBM and Xgboost, during preprocessing, We identified the feature selection's scaling and null values that have the greatest impact on the target variable. Different tactics can be used in this circumstance. Since URLs are essentially just text, using techniques for natural language processing provides us with a variety of options (NLP). Additional elements include the top-level domain, prefix, and if a subdomain is present. These features are all related to URLs in particular.

Proposed framework:

The proposed framework shows the detection of URL base phishing detection with preprocessing techniques, data training, testing, splitting data, and using classifiers, to achieve the prediction of the data.

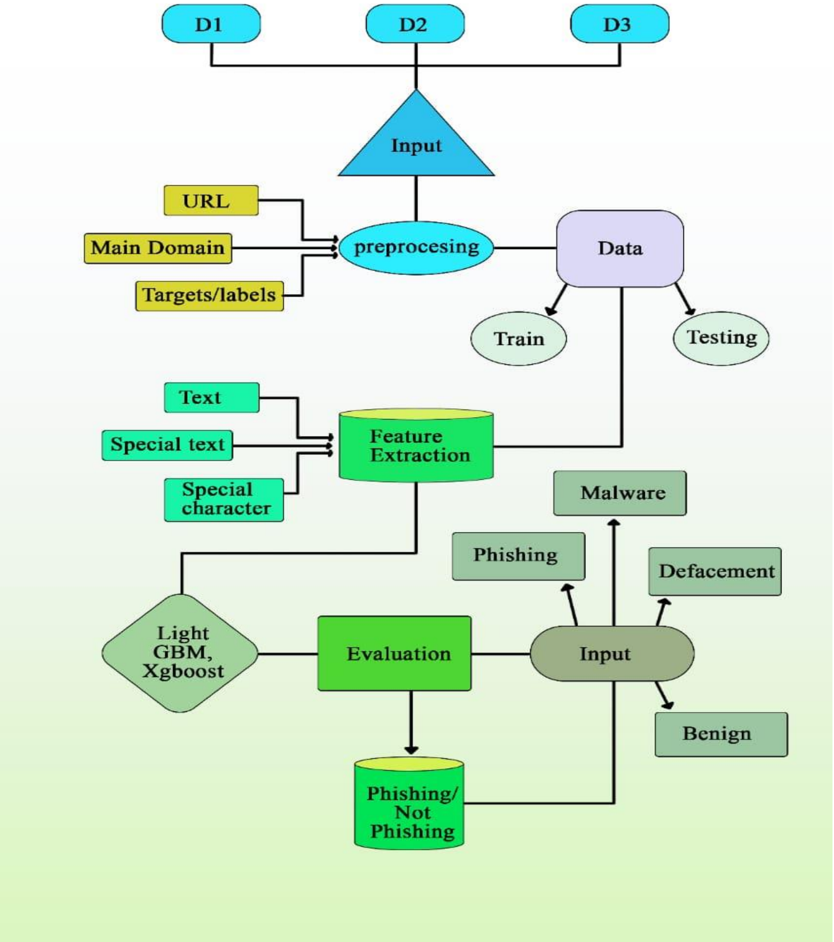


Figure 4 (Proposed Framework)

Data Preprocessing:

Data preprocessing is going to process data into different categories like URL domain, special text, and special characters to identify spam URLs and different websites.

Data Splitting:

The datasets of data splitting were utilized for testing and training. The distribution of training and testing sets, the choice of hyperparameters chosen during training, and other factors all have an impact on the success of deep learning algorithms. Based on the value that helped the CNN and LightGBM models function better, each parameter was chosen (CNNLGBM). Each layer's number of neurons, batch size, learning rate, dropout rate, number of epochs, type of activation function, and optimizer type are among these parameters.

Feature Extraction:

The tasks of feature extraction and classification have been finished by deep learning. Contrary to the text-based datasets, which largely consisted of email bodies that were cleaned and translated using NLP techniques, the properties of the numerical datasets depended on the author. For the aim of identifying phishing attempts, a phishing email or website can include numerous characteristics that can be retrieved.

Pre-processing:

To prepare the data for analysis, this phase entails cleansing and transformation. This might entail cleansing the data of duplicates, managing missing values, and normalizing it.

Model Training:

Using a dataset of well-known phishing assaults and genuine communications, this stage entails using the extracted attributes to train the phishing detection system, such as a machine learning model.

Model Validation:

By contrasting the trained model's predictions with the actual results, this stage entails assessing the trained model's performance. F1 score, recall, precision, and accuracy are a few measures that may be used to do this.

Performance Analysis:

Analyzing the outcomes of the model validation at this point will help you find any areas that might want improvement. The model may need to be adjusted, the prediction characteristics may need to be changed, or alternative phishing detection techniques may need to be investigated.

Deployment:

It is necessary to implement the phishing detection system at this point so that it may be used to identify and stop genuine phishing assaults in a real-world setting.

Experiments and Results:

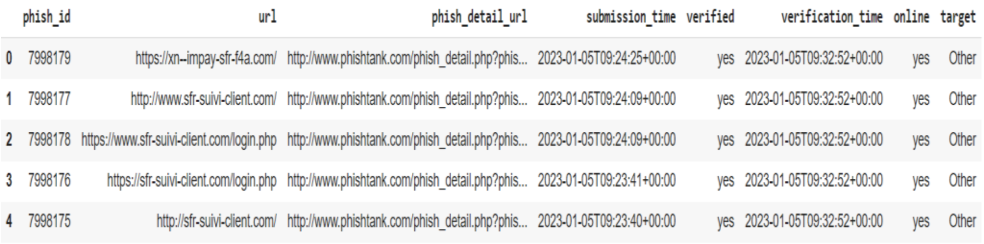
The domain name dataset used for model training came from Kaggle's publicly accessible data. Domain names were divided into two groups: those belonging to reliable websites and those belonging to phishing websites. The volume of visitors might, in part, indicate the reliability of the domain name. Compared to the domain names of phishing websites, the typical domain name has a long lifetime and more users.

Assessment Indicators:

Commonly used metrics for evaluating machine-learning-based techniques include accuracy, F value (F1), precision, and recall the . Recall gives the fraction of right predictions among all positive data, whereas accuracy shows the ratio of accurate predictions. F1 represented the harmonic mean of recall and accuracy.

Model Parameter Effects on Experimental Findings:

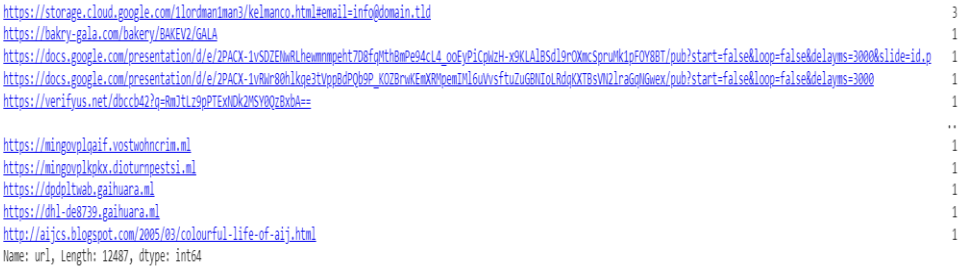
The performance of the phishing detection method is depending on several parameters like feature selection, model architecture, training data, hyperparameters, and evaluation metrics. Phishing detection models can be affected by a variety of factors, and it is important to carefully consider each of these factors when designing experiments to evaluate phishing detection models. In the phishing detection method using hybrid LightGBM, XgBoost combination to detect the phishing attack on URLs. The performance of the LightGBM and XgBoost is very appreciated to give good accuracy on a single tree. On the other hand, the multiple trees are trained the performance of the combined classifiers is good. To use google colab to finding phishing attacks on URLs. Import some important python libraries and load the data. After loading the data, the data will be shown as



	phish_id	url	phish_detail_url	submission_time	verified	verification_time	online	target
0	7998179	https://xn--impay-sfr-f4a.com/	http://www.phishtank.com/phish_detail.php?phis...	2023-01-05T09:24:25+00:00	yes	2023-01-05T09:32:52+00:00	yes	Other
1	7998177	http://www.sfr-suivi-client.com/	http://www.phishtank.com/phish_detail.php?phis...	2023-01-05T09:24:09+00:00	yes	2023-01-05T09:32:52+00:00	yes	Other
2	7998178	https://www.sfr-suivi-client.com/login.php	http://www.phishtank.com/phish_detail.php?phis...	2023-01-05T09:24:09+00:00	yes	2023-01-05T09:32:52+00:00	yes	Other
3	7998176	https://sfr-suivi-client.com/login.php	http://www.phishtank.com/phish_detail.php?phis...	2023-01-05T09:23:41+00:00	yes	2023-01-05T09:32:52+00:00	yes	Other
4	7998175	http://sfr-suivi-client.com/	http://www.phishtank.com/phish_detail.php?phis...	2023-01-05T09:23:40+00:00	yes	2023-01-05T09:32:52+00:00	yes	Other

Figure 5 (Visualize Data)

After a visual look at the URLs. The URL shown as their attributes

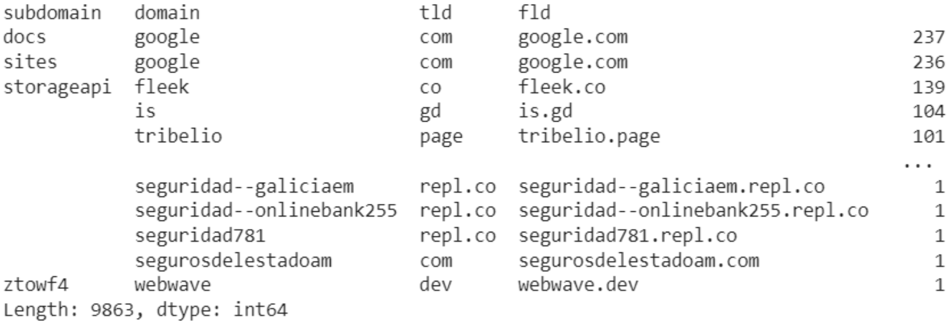


```
https://storage.cloud.google.com/1lordman1man3/kelmanco.html#email=info@domain.tld    3
https://bakry-gala.com/bakery/BAKEV2/GALA    1
https://docs.google.com/presentation/d/e/2PACX-1vSDZENwRLhewmnmpeht7D8fqMthBmPe94cL4_ooEyPiCpWzH-x9KLAlBSdl9rOXmcSpruMk1pFOY8BT/pub?start=false&loop=false&delayms=3000&slide=id.p    1
https://docs.google.com/presentation/d/e/2PACX-1vRWr80hlkqe3tVppBdPQb9P_KOZBrwKEmXRMpemIMl6uVvsftuZuGBNIoLRdqKXTBsVN2lraGqNGwex/pub?start=false&loop=false&delayms=3000    1
https://verifyus.net/dbccb42?q=RmJtLz9pPTExNDk2MSY0QzBxbA==    1
                                                                                    ..
https://mingovplqaif.vostwohncrim.ml    1
https://mingovplkpkx.dioturnpestsi.ml    1
https://dpdpltwab.gaihuara.ml    1
https://dhl-de8739.gaihuara.ml    1
http://aijcs.blogspot.com/2005/03/colourful-life-of-aij.html    1
Name: url, Length: 12487, dtype: int64
```

Figure 6 (Detecting URLs)

After applying the attributes methods, applying some feature selection methods on it, and selecting domain features we get some domains like http://sfr-suivi-client.com/ , com, and 'com'.

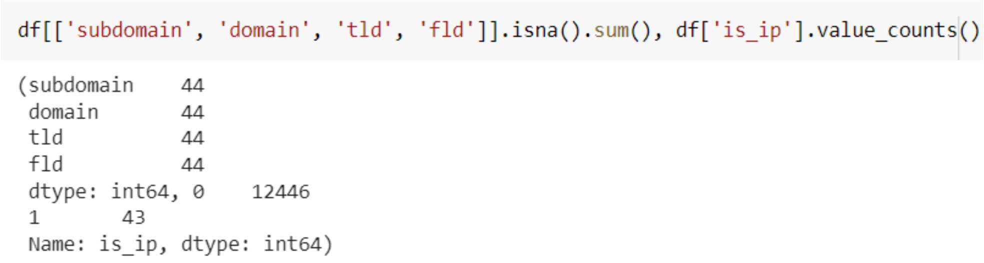
It appears that some sorts of URLs cause the processing method described above to fail. by examining the ones, it failed on. I'm not sure what kind of encoding is used, but the URLs appear to be encoded in some way.



```
subdomain   domain                   tld     fld
docs        google                   com     google.com                   237
sites       google                   com     google.com                   236
storageapi  fleek                    co      fleek.co                     139
            is                       gd      is.gd                        104
            tribelio                 page    tribelio.page                101
                                                                          ...
            seguridad--galiciaem     repl.co seguridad--galiciaem.repl.co     1
            seguridad--onlinebank255 repl.co seguridad--onlinebank255.repl.co 1
            seguridad781             repl.co seguridad781.repl.co             1
            segurosdelestadoam       com     segurosdelestadoam.com           1
ztowf4      webwave                  dev     webwave.dev                      1
Length: 9863, dtype: int64
```

Figure 7 (Informational URL)

After that, we are checking the slipped through cracks are given

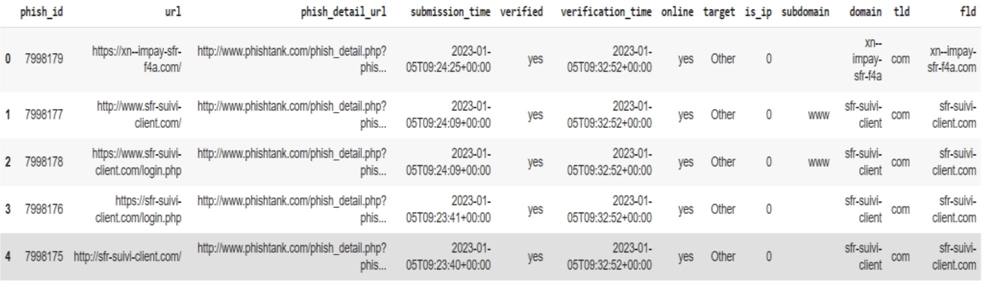


```
df[['subdomain', 'domain', 'tld', 'fld']].isna().sum(), df['is_ip'].value_counts()
```

```
(subdomain    44
 domain       44
 tld          44
 fld          44
 dtype: int64, 0    12446
 1      43
 Name: is_ip, dtype: int64)
```

Figure 8 (Checking URL)

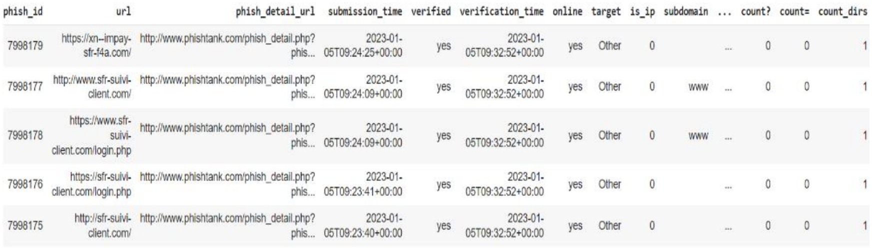
The results of the following code snippet show that when the newly developed features are empty and the row is an IP, 90 managed to avoid detection in every row. Figure 10 details the characteristics of the provided URLs, including their kind, verification period, and submission period. Also mentioned are the features of the domain, subdomain, top-level domain (tld), and free-level domain (fld).



	phish_id	url	phish_detail_url	submission_time	verified	verification_time	online	target	is_ip	subdomain	domain	tld	fld
0	7998179	https://xn--impay-sfr-f4a.com/	http://www.phishtank.com/phish_detail.php?phis...	2023-01-05T09:24:25+00:00	yes	2023-01-05T09:32:52+00:00	yes	Other	0		xn--impay-sfr-f4a	com	xn--impay-sfr-f4a.com
1	7998177	http://www.sfr-suivi-client.com/	http://www.phishtank.com/phish_detail.php?phis...	2023-01-05T09:24:09+00:00	yes	2023-01-05T09:32:52+00:00	yes	Other	0	www	sfr-suivi-client	com	sfr-suivi-client.com
2	7998178	https://www.sfr-suivi-client.com/login.php	http://www.phishtank.com/phish_detail.php?phis...	2023-01-05T09:24:09+00:00	yes	2023-01-05T09:32:52+00:00	yes	Other	0	www	sfr-suivi-client	com	sfr-suivi-client.com
3	7998176	https://sfr-suivi-client.com/login.php	http://www.phishtank.com/phish_detail.php?phis...	2023-01-05T09:23:41+00:00	yes	2023-01-05T09:32:52+00:00	yes	Other	0		sfr-suivi-client	com	sfr-suivi-client.com
4	7998175	http://sfr-suivi-client.com/	http://www.phishtank.com/phish_detail.php?phis...	2023-01-05T09:23:40+00:00	yes	2023-01-05T09:32:52+00:00	yes	Other	0		sfr-suivi-client	com	sfr-suivi-client.com

Figure 9 (Features Detecting URL)

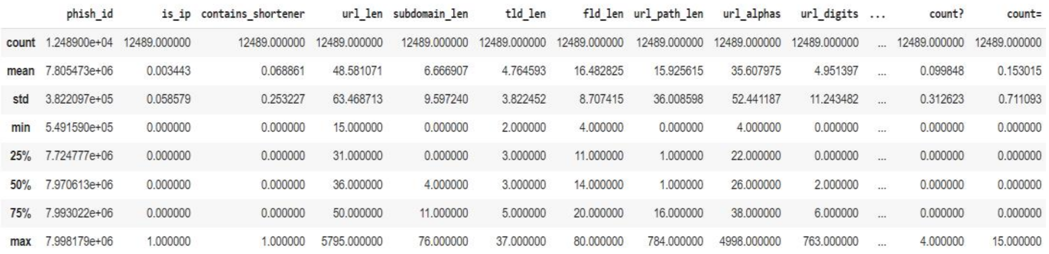
After extracting the aforementioned traits, let's extract some more. We go into the extraction of additional features in more detail in Figure 11. One of them involves quantifying URLs, which takes into account elements like URL count, count direction, and numerical tallies of URLs. This procedure involves determining several qualities from URLs, such as those that are confirmed, their time of submission, and the status they carry. The URLs' current online or offline status is indicated by this status.



phish_id	url	phish_detail_url	submission_time	verified	verification_time	online	target	is_ip	subdomain	...	count?	count=	count_dirs
7998179	https://xn--impay-sfr-f4a.com/	http://www.phishtank.com/phish_detail.php?phis...	2023-01-05T09:24:25+00:00	yes	2023-01-05T09:32:52+00:00	yes	Other	0		...	0	0	1
7998177	http://www.sfr-suivi-client.com/	http://www.phishtank.com/phish_detail.php?phis...	2023-01-05T09:24:09+00:00	yes	2023-01-05T09:32:52+00:00	yes	Other	0	www	...	0	0	1
7998178	https://www.sfr-suivi-client.com/login.php	http://www.phishtank.com/phish_detail.php?phis...	2023-01-05T09:24:09+00:00	yes	2023-01-05T09:32:52+00:00	yes	Other	0	www	...	0	0	1
7998176	https://sfr-suivi-client.com/login.php	http://www.phishtank.com/phish_detail.php?phis...	2023-01-05T09:23:41+00:00	yes	2023-01-05T09:32:52+00:00	yes	Other	0		...	0	0	1
7998175	http://sfr-suivi-client.com/	http://www.phishtank.com/phish_detail.php?phis...	2023-01-05T09:23:40+00:00	yes	2023-01-05T09:32:52+00:00	yes	Other	0		...	0	0	1

Figure 10 (Extracting Informational URL)

The binned and ratio features are the part I contributed. The idea behind both of these is that they could give a useful sign al to the model of malicious URLs. In figure 12 shows the values that are depending on the dataset, selected ratios, and machine learning method employed, the precise design and efficacy of such features can change. Binned ratio attributes are merely one of the methods that can help create a powerful spam URL detection system.



	phish_id	is_ip	contains_shortener	url_len	subdomain_len	tld_len	fld_len	url_path_len	url_alphas	url_digits	...	count?	count=
count	1.248900e+04	12489.000000	12489.000000	12489.000000	12489.000000	12489.000000	12489.000000	12489.000000	12489.000000	12489.000000	...	12489.000000	12489.000000
mean	7.805473e+06	0.003443	0.068861	48.581071	6.666907	4.764593	16.482825	15.925615	35.607975	4.951397	...	0.099848	0.153015
std	3.822097e+05	0.058579	0.253227	63.468713	9.597240	3.822452	8.707415	36.008598	52.441187	11.243482	...	0.312623	0.711093
min	5.491590e+05	0.000000	0.000000	15.000000	0.000000	2.000000	4.000000	0.000000	4.000000	0.000000	...	0.000000	0.000000
25%	7.724777e+06	0.000000	0.000000	31.000000	0.000000	3.000000	11.000000	1.000000	22.000000	0.000000	...	0.000000	0.000000
50%	7.970613e+06	0.000000	0.000000	36.000000	4.000000	3.000000	14.000000	1.000000	26.000000	2.000000	...	0.000000	0.000000
75%	7.993022e+06	0.000000	0.000000	50.000000	11.000000	5.000000	20.000000	16.000000	38.000000	6.000000	...	0.000000	0.000000
max	7.998179e+06	1.000000	1.000000	5795.000000	76.000000	37.000000	80.000000	784.000000	4998.000000	763.000000	...	4.000000	15.000000

Figure 11 (Binned Ratio of URL)

Looking at this output above reveals a number of intriguing things. Although the maximum is 2175, the average is 60. This clearly indicates that there are some outliers, and it could be worthwhile to try to realign those. Additionally, based on a check at subdomain len and URL path length, the excessive length is caused by one or more outliers in the URL path section of the URLs. Several punctuation marks, such %, appear to make up a significant amount of the URLs. Figure 13 depicts four distinctive URL properties, each of which captures a different element of the URL structure. These characteristics cover a range of measures, such as the overall length of the URL, the size of the subdomain, the length of the top-level domain, and the total length of the domain. Together, they offer thorough coverage of the URL from a range of angles.

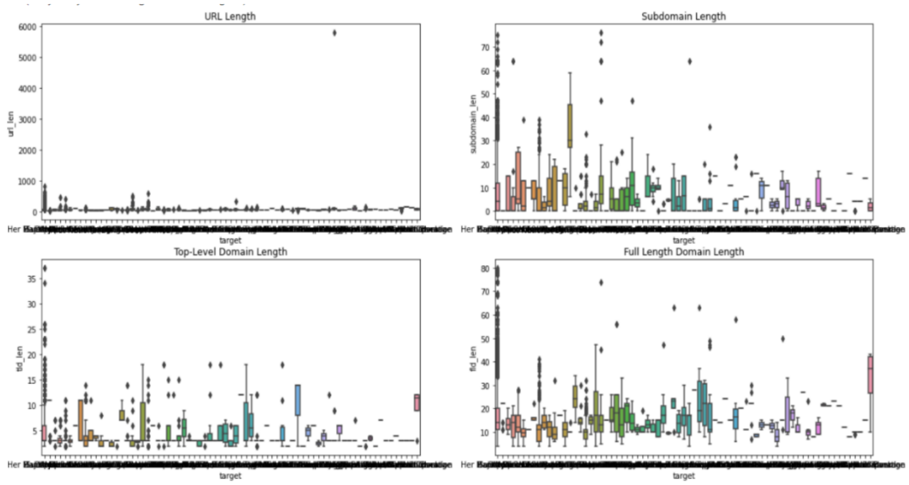


Figure 12 (Different Path Length of URL)

Now Figure 13 shows a variety of variables related to alpha URLs, emphasizing the analysis of elements such URL length, character percentage, and the number of digits incorporated into the URLs. This analysis aims to quantify the proportion of characters used in URLs and the relative frequency of digits in the URL structure.

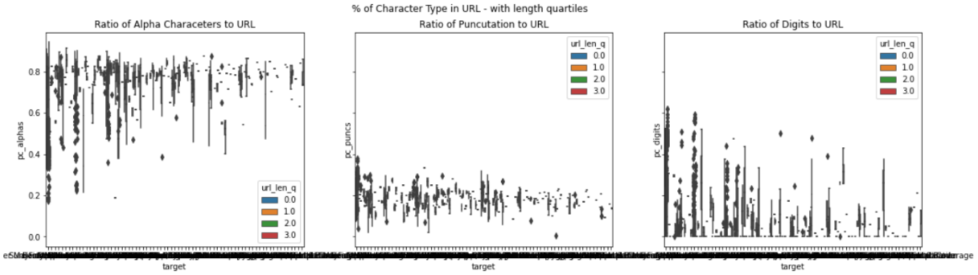


Figure 13 (Different URL Length)

When training with all decision trees, the results of phishing detection using LightGBM and XgBoost are assessed. The overall URL length, character percentages, and character ratios are all used in the training and testing of both LightGBM and XgBoost. The metrics for a URL's quality of service as determined by LightGBM and XgBoost are shown graphically in table 1.

	Precision	recall	f1-score	support
1	0.96	0.96	0.96	2498
Accuracy	0.96	0.96	0.96	2498
macro avg	0.96	0.96	0.96	2498
weighted avg	0.96	0.96	0.96	2498

Table 1 (LightGBM Accuracy Result)

	Precision	recall	f1-score	support
1	0.95	0.95	0.95	2498
Accuracy	0.95	0.95	0.95	2498
macro avg	0.95	0.95	0.95	2498
weighted avg	0.95	0.95	0.95	2498

Table 2 (XgBoost Accuracy Result)

Both are shown 96% results in all fields. Now showing the plotting of the LightGBM and XgBoost classifiers.

Figure 14 uses a mat plot table to represent the axes of the URL graph and shows the aggregate strength of URLs in relation to the number of URLs being examined. The graphical results are represented visually in the graph.

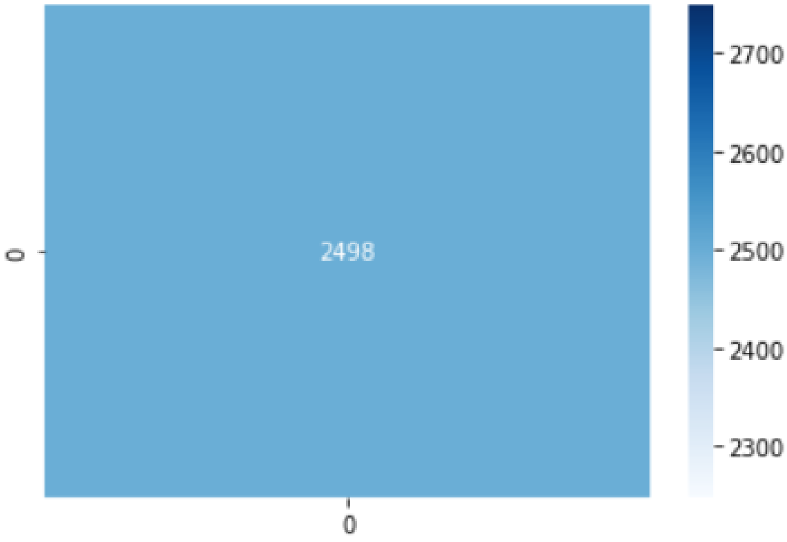
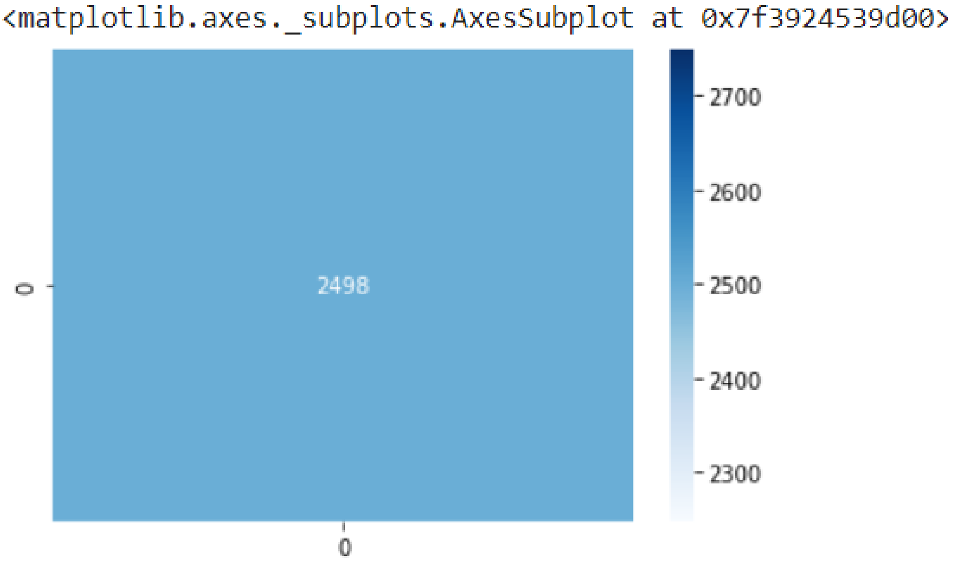


Figure 14 (LightGBM plot)



```
<matplotlib.axes._subplots.AxesSubplot at 0x7f3924539d00>
```

Figure 15 (XgBoost Plot)

LightGBM:

Tree-based learning methods are used in LightGBM, an open-source, distributed gradient boosting system. It is a popular option for machine learning professionals dealing with big datasets since it is made to be effective and scalable.

Its effective utilization of memory and computing resources gives LightGBM a significant edge over other gradient boosting systems. This is accomplished using a variety of methods, including a split finding algorithm based on histograms, which speeds up tree construction, and leaf-wise tree development, which produces models that are more manageable and easier to understand.

Additionally, LightGBM offers parallel and GPU-accelerated training, allowing for the quick training of models on huge datasets. The capacity to manage missing values and extreme values in the data is only one of its many sophisticated capabilities. It also includes automated feature selection.

For a number of tasks, including classification, regression, and ranking, LightGBM is widely utilized in a variety of including healthcare, e-commerce, industries and banking. Additionally, it has shown to be effective in Kaggle machine-learning competitions and has been utilized to take home several awards.

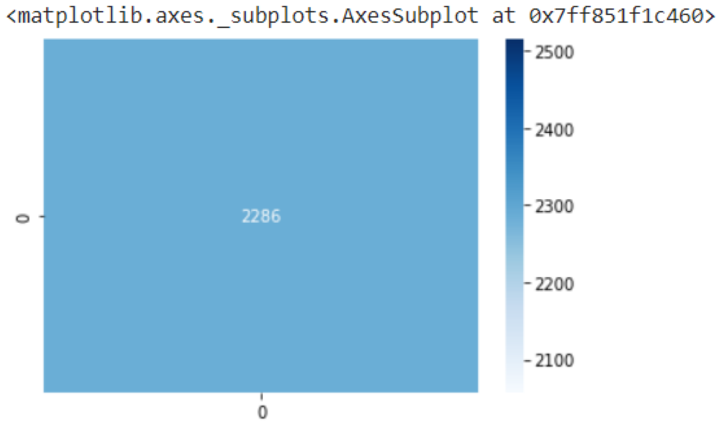
Working with huge datasets is made possible by LightGBM, a quick, effective, and scalable gradient boosting system. Machine learning practitioners frequently choose it because of its effectiveness, cutting-edge features, and great performance.

The dataset was trained with LightGBM Classifier with binary label trees. The accuracy of the data is 96%.

	Precision	**recall**	**f1-score**	**support**
1	0.96	0.96	0.96	2286
Accuracy	0.96	0.96	0.96	2286
macro avg	0.96	0.96	0.96	2286
weighted avg	0.96	0.96	0.96	2286
Total	0.96			

Table 3 (Result of LightGBM)

Now shown is the plotting result of LightGBM.



```
<matplotlib.axes._subplots.AxesSubplot at 0x7ff851f1c460>
```

Figure 16 (Plot of LightGBM)

XGBOOST:

The open-source gradient boosting framework XGBoost is used in machine learning. It is a preferred option for developing sophisticated predictive models since it is quick, scalable, and adaptable. With its strong handling of sparse data and missing values, the tree-based learning algorithm XGBoost is well recognized. Additionally, it provides parallel processing, which enables the quick training of big models. One of XGBoost's main advantages is its capacity to automatically manage feature selection, which human feature engineering eliminates the need. This makes it a useful tool for practitioners who are dealing with a lot of features or are unfamiliar with feature engineering.

In machine learning contests like Kaggle, XGBoost has a proven track record and has been utilized to take home several awards. For a number of tasks, including classification, regression, and ranking, it is also frequently utilized in a variety of including healthcare, ecommerce, industries and banking. Building sophisticated predictive models is a good fit for the robust, quick, and adaptable gradient-boosting framework known as XGBoost. It is a popular option among machine learning practitioners because of its capacity to handle sparse data, and missing values, and automatically execute feature selection.

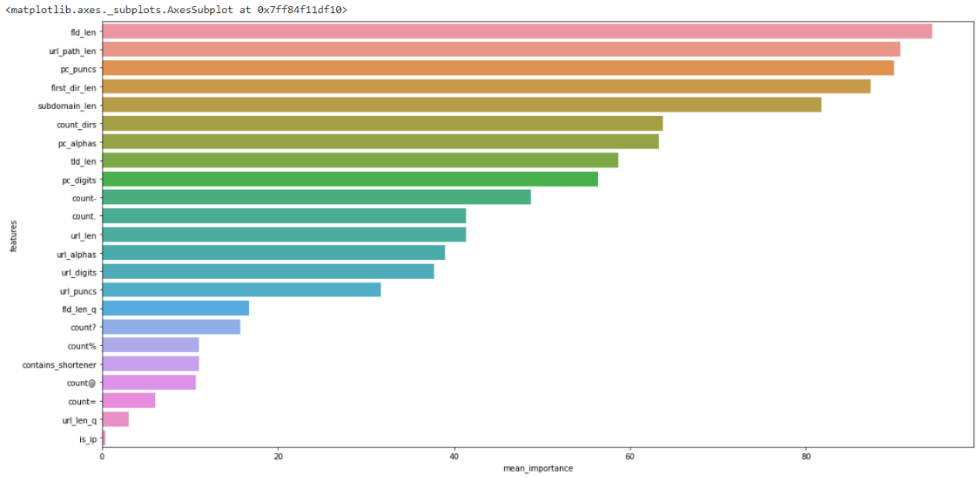
The dataset was trained with CNN, LightGBM, and XgBoost Classifier with binary label trees. The accuracy of the data is 94%.

	Precision	recall	f1-score	support
1	0.947	0.947	0.947	2286
Accuracy	0.947	0.947	0.947	2286
macro avg	0.947	0.947	0.947	2286
weighted avg	0.947	0.947	0.947	2286
Total	0.947			

Table 4 (Result of XgBoost)

Analysis of Features' Importance:

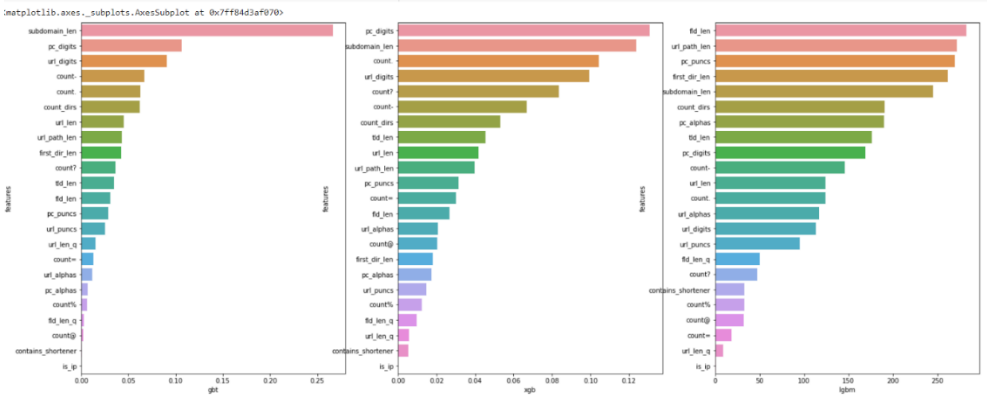
The feature significance values of the LGBM model have a significant skewed effect on the mean measure of centrality, which has been used to quantify centrality.



```
<matplotlib.axes._subplots.AxesSubplot at 0x7ff84f11df10>
```

Figure 17 (Importance of Features)

Let's have a look at how the feature importance’s for different models appear.



```
<matplotlib.axes._subplots.AxesSubplot at 0x7ff84d3af070>
```

Figure 18 (Comparison of URL Different models)

Comparison of all features of the URL length.

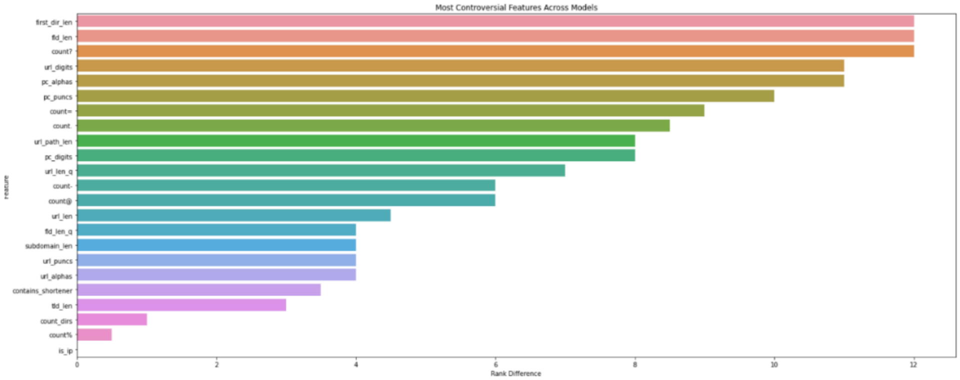


Figure 19 (URL length)

(Zhou et al., 2023) The accuracy of the model was 87.51% when it was trained simply on the character-level characteristics of the domain name and 88.36% when it was trained only on the information-level characteristics of the domain name. When using two domain name features, their respective phishing website detection accuracy rates were 6.37 and 5.52% greater than when using just one feature. The domain feature model performed better than the single-feature model on the other three assessment metrics as well, outperforming it in terms of precision and recall by 5.29% and 3.92%, 5.96% and 9.46%, and F value by 5.63% and 6.84 respectively. The properties of the domain name's characters were added to better depict in addition to the information on the domain name, there are discrepancies between the domain names of trustworthy websites and phishing websites. The model as a whole performed better as a result in the fig is shown as.

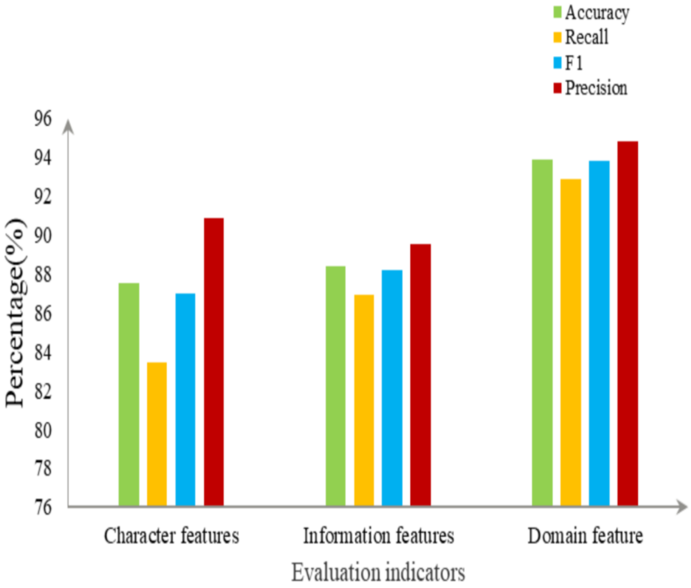


Figure 20 (Existing Model Graph)

compared NLP and DL algorithms he most effective to identify combination for phishing and spam email detection; with a hybrid model combination of combined classifier with LightGBM and XgBoost perform ensemble learning technique the results shows the better performance of detection Phishing URLs and spam emails. Fig shown as.

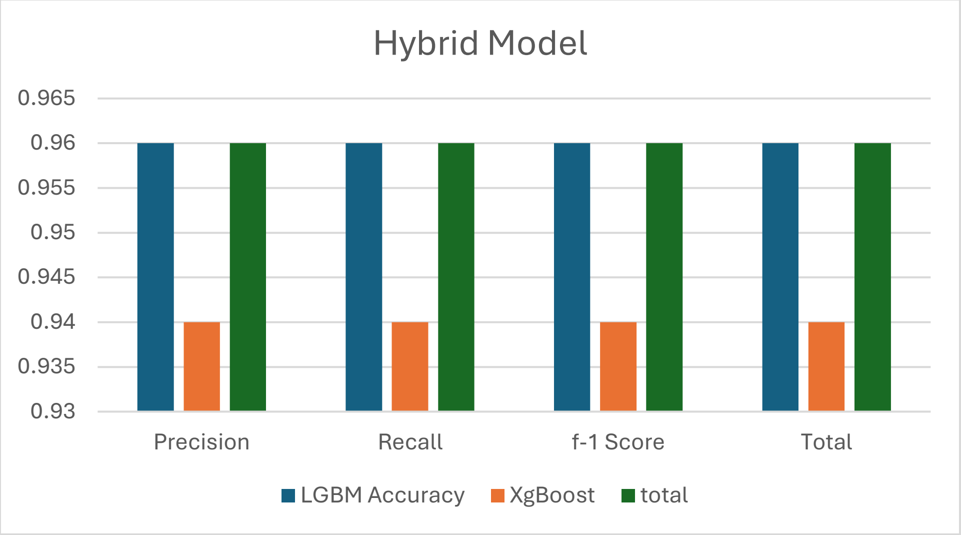


Figure 21 (Hybrid model Graph)

When ensemble Learning technique applied on LightGBM, XgBoost, and Gradient Boost the accuracy of LightGBM is higher than other classifiers is 91%, and it's a better result shows as a previous work. The fig shown as:

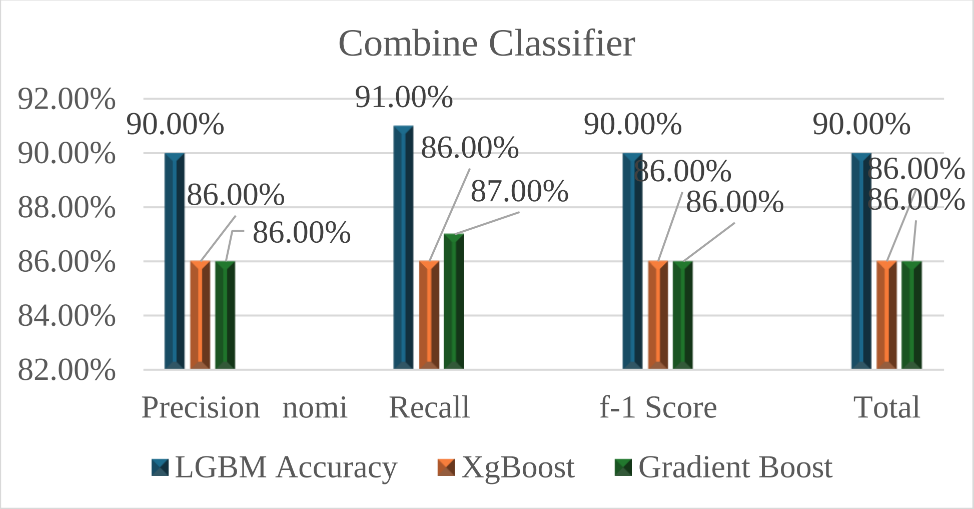


Figure 22 (combine comparison classifier)

4. Discussion

This study suggests a hybrid model for phishing websites that incorporates characteristics from the URL and domain name and is based on the LightGBM+XgBoost classifier. This paper focused on the detection of phishing attacks on URLs. Some attacker contains some extra feature to make URL threat, the men can't know the right URL and click on it. The attacker makes this advantage to steal your personal information. The hybrid model detects the phishing URL by using feature extraction methods like domain features attributes features and some special characters to find easily detect the phishing attack.

Research Q# 1: How the feature engineering techniques will be implemented on phishing attacks (URL-based data)?

Ans: Using the NLP-based feature engineering technique like top-level domain subdomain characters of URL etc. for detecting Phish URLs to enhance the quality-of-service parameters

Research Q# 2: Which type of ML-based classifiers are suitable for phishing attack (URLbased data) detection?

Ans: The proposed model becomes more efficient than the existing model because the LightGBM and Xgboost are suitable classifiers to enhance the quality-of-service parameters of bogus URLs for using AI Techniques.

Research Q# 3: How the proposed model will be validated and verified?

Ans: By using LightGBM and Xgboost to analyze the bogus URLs will be validated and verified with the existing model.

In this paper the research answer of the question has to more the accuracy of the data to using the classifier and reduce the text base some feature to improve the outcomes of the attacks. Using ensemble learning technique to increase the ratio of phishing attack detection on URL or web. From the existing model the hybrid mode become faster and reduce time complexity for detecting phishing attacks on URL or web.

5. Conclusion

In the Conclusion proposed paper for testing, we used data from the Phish Tank, phish, and spam URL datasets. Features of the characters, qualities, and classes utilized in the domain names, as well as characteristics of the information on the names of domain are divide into categories for the major names of domain in phishing websites.

Finally, more than 16 domain name characteristics were chosen for model training after being filtered. During training, the LightGBM model's parameters were optimized using the gridsearch technique. With the suggested model, we contrasted the effectiveness of additional models. Improvements in recall, F score, precision, and accuracy the value demonstrate that the model that employed domain name for training features beat the models that used only a feature in single.

Additionally, the hybrid model proposes outperformed the gradient boost, XGBoost, and LightGBM models as well.

6. References

1. Yaseen, Q. (2021). Spam email detection using deep learning techniques. Procedia Computer Science, 184, 853-858..
2. Lakshmi, L., Reddy, M. P., Santhaiah, C., & Reddy, U. J. (2021). Smart phishing detection in web pages using supervised deep learning classification and optimization technique adam. Wireless Personal Communications, 118(4), 3549-3564.
3. Wei, B., Hamad, R. A., Yang, L., He, X., Wang, H., Gao, B., & Woo, W. L. (2019). A deep-learning-driven light-weight phishing detection sensor. Sensors, 19(19), 4258.
4. Ben Fredj, O., Mihoub, A., Krichen, M., Cheikhrouhou, O., & Derhab, A. (2020, November). CyberSecurity attack prediction: a deep learning approach. In 13th International Conference on Security of Information and Networks (pp. 1-6).
5. Samy, A., Yu, H., & Zhang, H. (2020). Fog-based attack detection framework for internet of things using deep learning. IEEE Access, 8, 74571-74585.
6. Al-Abassi, A., Karimipour, H., Dehghantanha, A., & Parizi, R. M. (2020). An ensemble deep learning-based cyber-attack detection in industrial control system. IEEE Access, 8, 83965-83973.
7. Manimurugan, S., Al-Mutairi, S., Aborokbah, M. M., Chilamkurti, N., Ganesan, S., & Patan, R. (2020). Effective attack detection in internet of medical things smart environment using a deep belief neural network. IEEE Access, 8, 77396-77404.
8. Hindy, H., Atkinson, R., Tachtatzis, C., Colin, J. N., Bayne, E., & Bellekens, X. (2020). Utilising deep learning techniques for effective zero-day attack detection. Electronics, 9(10), 1684.

9. Saha, I., Sarma, D., Chakma, R. J., Alam, M. N., Sultana, A., & Hossain, S. (2020, August). Phishing attacks detection using deep learning approach. In 2020 Third International Conference on Smart Systems and Inventive Technology (ICSSIT) (pp.1180-1185). IEEE.

10. Sriram, S., Vinayakumar, R., Alazab, M., & Soman, K. P. (2020, July). Network flow based IoT botnet attack detection using deep learning. In IEEE INFOCOM 2020-IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS) (pp. 89-194). IEEE.

11. Wu, Y., Wei, D., & Feng, J. (2020). Network attacks detection methods based on deep learning techniques: a survey. Security and Communication Networks, 2020.

12.Dutta, V., Choraś, M., Pawlicki, M., & Kozik, R. (2020). A deep learning ensemble for network anomaly and cyber-attack detection. Sensors, 20(16), 4583.

13. Sahu, A. K., Sharma, S., Tanveer, M., & Raja, R. (2021). Internet of Things attack detection using hybrid Deep Learning Model. Computer Communications, 176, 146- 154.

14. Tekerek, A. (2021). A novel architecture for web-based attack detection using convolutional neural network. Computers & Security, 100, 102096.

15. Sengan, S., Subramaniyaswamy, V., Indragandhi, V., Velayutham, P., & Ravi, L. (2021). Detection of false data cyber-attacks for the assessment of security in smart grid using deep learning. Computers & Electrical Engineering, 93, 107211..

16. Chen, D., Yan, Q., Wu, C., & Zhao, J. (2021). Sql injection attack detection and prevention techniques using deep learning. In Journal of Physics: Conference Series (Vol. 1757, No. 1, p. 012055). IOP Publishing.

17. Al-Mhiqani, M. N., Ahmed, R., Zainal, Z., & Isnin, S. (2021). An integrated imbalanced learning and deep neural network model for insider threat detection. Int. J. Adv. Comput. Sci. Appl, 12(1), 1-6.

18. Sarker, I. H. (2021). Deep cybersecurity: a comprehensive overview from neural network and deep learning perspective. SN Computer Science, 2(3), 1-16..

19. Pantelidis, E., Bendiab, G., Shiaeles, S., & Kolokotronis, N. (2021). Insider Detection using Deep Autoencoder and Variational Autoencoder Neural Networks. arXiv preprint arXiv:2109.02568.

20. Fotiadou, K., Velivassaki, T. H., Voulkidis, A., Skias, D., Tsekeridou, S., & Zahariadis, T. (2021). Network traffic anomaly detection via deep learning. Information, 12(5), 215.

21. Sánchez-Paniagua, M., Fidalgo, E., Alegre, E., & Alaiz-Rodríguez, R. (2022). Phishing websites detection using a novel multipurpose dataset and web technologies features. Expert Systems with Applications, 207, 118010.

22. Ho, G., Sharma, A., Javed, M., Paxson, V., & Wagner, D. (2017). Detecting credential spearphishing attacks in enterprise settings. Proc. of 26th USENIX Security.

23. Soon, G. K., On, C. K., Rusli, N. M., Fun, T. S., Alfred, R., & Guan, T. T. (2020, March). Comparison of simple feedforward neural network, recurrent neural network and ensemble neural networks in phishing detection. In Journal of Physics:Conference Series (Vol. 1502, No. 1, p. 012033). IOP Publishing.

24. Evans, K., Abuadbba, A., Wu, T., Moore, K., Ahmed, M., Pogrebna, G., ... &Johnstone, M. (2022, December). RAIDER: Reinforcement-aided spear phishing detector. In Network and System Security: 16th International Conference, NSS 2022, Denarau Island, Fiji, December 9–12, 2022, Proceedings (pp. 23-50). Cham: Springer Nature Switzerland.

25. Xiujuan, W., Chenxi, Z., Kangfeng, Z., Haoyang, T., & Yuanrui, T. (2019, February). Detecting spear-phishing emails based on authentication. In 2019 IEEE 4th International Conference on Computer and Communication Systems (ICCCS) (pp. 450-456). IEEE.

26. Butnaru, A., Mylonas, A., & Pitropakis, N. (2021). Towards lightweight url-based phishing detection. Future internet, 13(6), 154.

27. Mittal, A., Engels, D. D., Kommanapalli, H., Sivaraman, R., & Chowdhury, T. (2022). Phishing Detection Using Natural Language Processing and Machine Learning. SMU Data Science Review, 6(2), 14.

28. Beaman, C., & Isah, H. (2022). Anomaly Detection in Emails using Machine Learning and Header Information. arXiv preprint arXiv:2203.10408.

29. Ghafir, I., Hammoudeh, M., Prenosil, V., Han, L., Hegarty, R., Rabie, K., & AparicioNavarro, F. J. (2018). Detection of advanced persistent threat using machine-learning correlation analysis. Future Generation Computer Systems, 89, 349-359.

30. Lee, H., Jang, H., Han, S., & Gim, G. (2019). Security Monitoring Technological Approach for Spear Phishing Detection. Software Engineering, Artificial Intelligence, Networking and Parallel/Distributed Computing, 149-163.

31. Ghosh, A., & Senthilrajan, A. (2019, December). An approach for detecting spear phishing using deep packet inspection and deep flow inspection. In Proceedings of the 5th International Conference on Cyber Security & Privacy in Communication Networks (ICCS).

32. Rijnbergen, K. J. (2020). Improving the effectiveness of phishing detection Using lexical semantics; A machine-learning based approach (Bachelor's thesis, University of Twente).

33. Rasymas, T., & Dovydaitis, L. (2020). Detection of phishing URLs by using deep learning approach and multiple features combinations. Baltic journal of modern computing, 8(3), 471-483.

34. Feng, J., Zou, L., & Nan, T. (2019). A phishing webpage detection method based on stacked autoencoder and correlation coefficients. Journal of computing and information technology, 27(2), 41-54.

35. Safonov, Y. PHISHING DETECTION USING DEEP LEARNING ATTENTION TECHNIQUES.

36. Adebowale, M. A., Lwin, K. T., & Hossain, M. A. (2020). Intelligent phishing detection scheme using deep learning algorithms. Journal of Enterprise Information Management, (ahead-of-print).

37. Assegie, T. A. (2021). K-nearest neighbor based URL identification model for phishing attack detection. Indian Journal of Artificial Intelligence and Neural Networking, 1, 18-21.

38. Ariyadasa, S., Fernando, S., & Fernando, S. (2022). Combining Long-Term Recurrent Convolutional and Graph Convolutional Networks to Detect Phishing Sites Using URL and HTML. IEEE Access, 10, 82355-82375.

39. Tiwari, M., Thomble, D., Thite, A., Kapurkar, D., Surve, P., & Patil, C. H. Literature Review on Presentation Attack Detection using Deep Learning.

40. Eunaicy, J. C., & Suguna, S. (2022). Web attack detection using deep learning models. Materials Today: Proceedings, 62, 4806-4813.

41. Basit[1], A., Zafar, M., & Jalil, Z. (2020). A Review of Website Phishing Attack Detection Methods.

42. Ahmad, S. W., Ismail, M. A., Sutoyo, E., Kasim, S., & Mohamad, M. S. (2020). Comparative performance of machine learning methods for classification on phishing attack detection. International Journal of Advanced Trends in Computer Science and Engineering.

43. Buber, E., Diri, B., & Sahingoz, O. K. (2018). NLP based phishing attack detection from URLs. In Intelligent Systems Design and Applications: 17th International Conference on Intelligent Systems Design and Applications (ISDA 2017) held in Delhi, India, December 14-16, 2017 (pp. 608-618). springer international Publishing.

44. Bu, S. J., & Kim, H. J. (2022). Optimized URL Feature Selection Based on GeneticAlgorithm-Embedded Deep Learning for Phishing Website Detection. Electronics, 11(7), 1090.

45. Lee, J., Tang, F., Ye, P., Abbasi, F., Hay, P., & Divakaran, D. M. (2021, September).D-Fence: A flexible, efficient, and comprehensive phishing email detection system. In 2021 IEEE European Symposium on Security and Privacy (EuroS&P) (pp. 578- 597). IEEE.

46. Mughaid, A., AlZu'bi, S., Hnaif, A., Taamneh, S., Alnajjar, A., & Elsoud, E. A. (2022). An intelligent cyber security phishing detection system using deep learning techniques. Cluster Computing, 25(6), 3819-3828.

47. Mughaid, A., AlZu'bi, S., Hnaif, A., Taamneh, S., Alnajjar, A., & Elsoud, E. A. (2022). An intelligent cyber security phishing detection system using deep learning techniques. Cluster Computing, 25(6), 3819-3828.

48. Feng, J., Zou, L., Ye, O., & Han, J. (2020). Web2vec: Phishing webpage detection method based on multidimensional features driven by deep learning. IEEE Access, 8, 221214-221224.

49. Alma, T., & Das, M. L. (2020). Web Application Attack Detection using Deep Learning. arXiv preprint arXiv:2011.03181.

50. Zhao, C., Xin, Y., Li, X., Yang, Y., & Chen, Y. (2020). A heterogeneous ensemble learning framework for spam detection in social networks with imbalanced data. Applied Sciences, 10(3), 936.

51. Zhou, J., Cui, H., Li, X., Yang, W., & Wu, X. (2023). A Novel Phishing Website Detection Model Based on LightGBM and Domain Name Features. Symmetry, 15(1),180.

52. SRINIVAS, G. V., & MALINA, S. Identification Of Spammer Detection And Fake User On Social Networks Using Naive Bayes And Random Forest Algorithms.

53. YILDIRIM, M. (2022). Using and Comparing Machine Learning Techniques for Automatic Detection of Spam Website URLs. NATURENGS, 3(1), 33-41.

54. Awajan, A., Alazab, M., Khurma, R. A., Alsaadeh, R., Wedyan, M., & Abraham, A. (2022). Fake News Detection and Prevention Using Artificial Intelligence Techniques: A Review of a Decade of Research. International Journal of Computer Information Systems and Industrial Management Applications.

55. Tida, V. S., & Hsu, S. (2022). Universal spam detection using transfer learning of BERT model. arXiv preprint arXiv:2202.03480.