THE ASIAN BULLETIN OF BIG DATA MANAGMENT
Vol. 5. Issue 2 (2025)
https://doi.org/10.62019/w701fe48

ASIAN BULLETIN OF BIG DATA MANAGEMENT
http://abbdm.com/
ISSN (Print): 2959-0795
ISSN (online): 2959-0809

A Robust Model for Phishing URL Classification and Intrusion Detection using Machine Learning Techniques

Muhammad Imran Ghafoor, Paras Pervaiz, Shumaila Hussain *, Saima Tareen *, Mehmood Baryalai, Shariqa Fakhar, Shah Noor

Chronicle

Article history
Received: April 2, 2025
Received in the revised format: May3, 2025
Accepted: May 29 2025
Available online: June 27, 2025

Muhammad Imran Ghafoor, is currently affiliated with Department of Information Security, PUCIT Punjab University, Lahore, Pakistan.

Email: engr.imranbhatti09@ieee.org

Paras Pervaiz, is currently affiliated with Balochistan University of Information Technology, Engineering and Management Sciences (BUITEMS), Quetta Pakistan.

Email: paras.pervaiz@buitms.edu.pk

Shumaila Hussain, is currently affiliated with Department of Computer Science Sardar Bahadur Khan Women's University, Quetta Pakistan.

Email: Shumailahussain70@gmail.com

Saima Tareen, is currently affiliated with Computer Science Department, Balochistan University of Information Technology Engineering, and Management Sciences (BUITEMS), Quetta, Pakistan.

Email: saima.tareen26@gmail.com

Mehmood Baryalai, is currently affiliated with Department of IT, Balochistan University of Information Technology, Engineering and Management Sciences (BUITEMS), Quetta Pakistan.

Email:mehmood.baryalai@buitms.edu.pk

Shariqa Fakhar, is currently affiliated with Computer Science Department, Sardar Bahadur Khan Women's University, Quetta, Pakistan.

Email: Shariqa.fakhar@yahoo.com

Shah Noor, is currently affiliated with Computer Science Department, Balochistan University of Information Technology Engineering, and Management Sciences (BUITEMS), Quetta, Pakistan.

Email: shahnoorraza2@gmail.com

Corresponding Author*

Abstract

Phishing is one of the most prevalent and risky online threats. It works when hackers deceive internet users into providing personal information, such as passwords, login credentials, and credit card numbers, in order to obtain data that is frequently used against them. Victims are often sent phishing URLs (Uniform Resource Locators) via email. These URLs send users to fraudulent websites, phishing, spam, drive-by download attacks, and other hazardous websites. It's critical to accurately classify each URL as harmful or legitimate in order to prevent consumers from accessing malicious URLs. Phishing URL categorization helps in avoiding visits to harmful websites beforehand. To recognize intrusion attacks and classify phishing URLs, we provide a deep neural network-based method. Three sources of information were used: Kaggle, PhishTank, and Alexa. Term Frequency Inverse Document Frequency (TF-IDF) properties of a Support Vector Machine (SVM) are used to classify the phishing URLs in the first place. Second, we detect intrusions using a deep neural network. Finally, we evaluate our proposed model against previous approaches. Our research indicates that the SVM algorithm using TF-IDF produces an accuracy rate of 97.14% and a false positive rate of 2.8%. The model's intrusion detection predictions using validation data yielded promising results. We achieved an F1 score of 5.873%. With the exception of NMAP and a few other assaults, we obtained an accuracy rate greater than 95%. The main contributions of this study are: 1) improving phishing URL classification by combining SVM and TF-IDF, 2) utilizing a DNN model for efficient intrusion detection, and 3) conducting a thorough evaluation across multiple datasets to illustrate the reliability and robustness of the proposed method. The findings of the experiment indicate that the suggested model considerably enhances cybersecurity defensive systems, outperforming existing strategies in terms of accuracy, false positive rate, and detection precision.

Keywords: Artificial Intelligence, Deep Learning, Machine Learning, Uniform Resource Locator, Term Frequency-Inverse Document Frequency (TF-IDF), Phishing.

INTRODUCTION

Phishing assaults have significantly increased in recent years, demonstrated by the 1,003,924 phishing attacks reported by the Anti-Phishing Working Group (APWG) in the third quarter of 2025 (APWG, 2025) (Korkmaz et al., 2020) (Bazai, S., et al. 2017). Real-world tasks are being moved online due to the increasing use of mobile devices,

which is the cause of the increase (Dina & Manivannan, 2021).Even while this change makes both personal and professional activities easier, customers are at serious risk for security issues because online transactions have increased the possibilities of cyberattacks, including fraud, forgeries, hacking, Denial of Services (DOS), and social engineering (Bazai & Jang-Jaccard, 2020) (Bazai, S., et al. 2011).

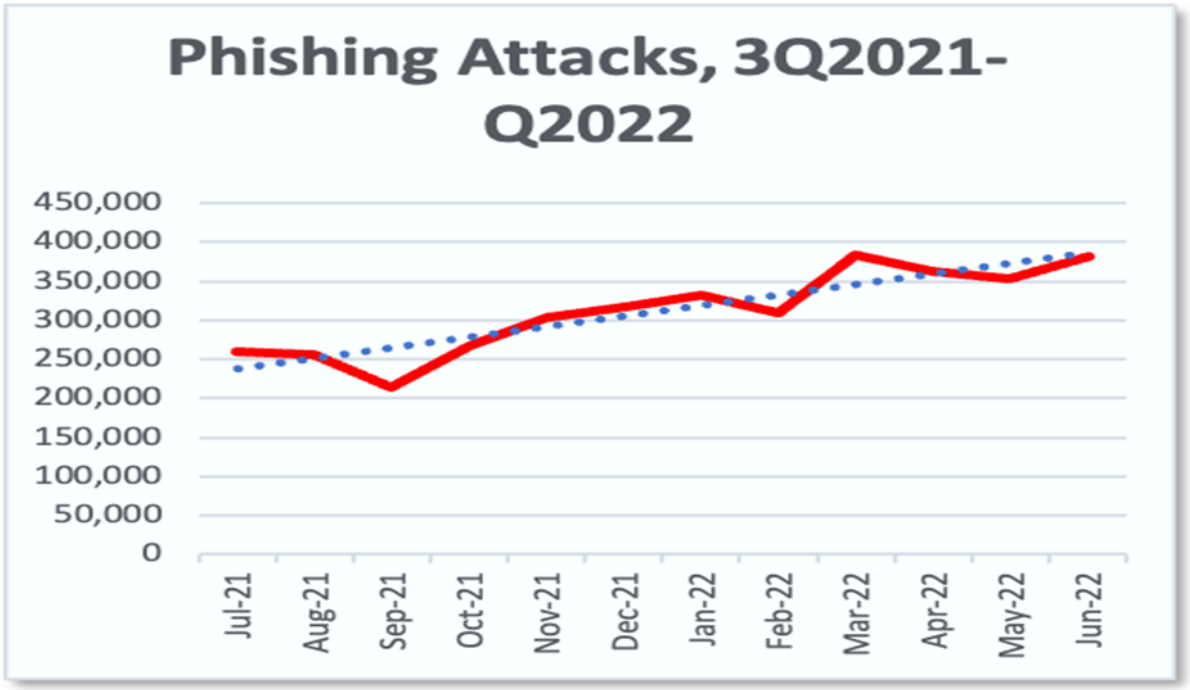

Figure 1.
Phishing attacks report 2021-1Q 2022

A new era has begun, when obtaining prospective information through data analysis and mining has become a top priority for many organizations due to the rapid growth of data in numerous areas (social media, mobile devices, IoT, etc.) (Aftab, F., et al., 2023) (Bazai, S., et al., 2021) (Bazai, S., et al., 2017). Installing firewalls and antivirus software is no longer enough to handle this amount of data. Keeping people's privacy is a constant and challenging issue that affects their daily lives or after first mention: (Khonji et al., 2013). One of the most pervasive and serious cybersecurity risks is still phishing.

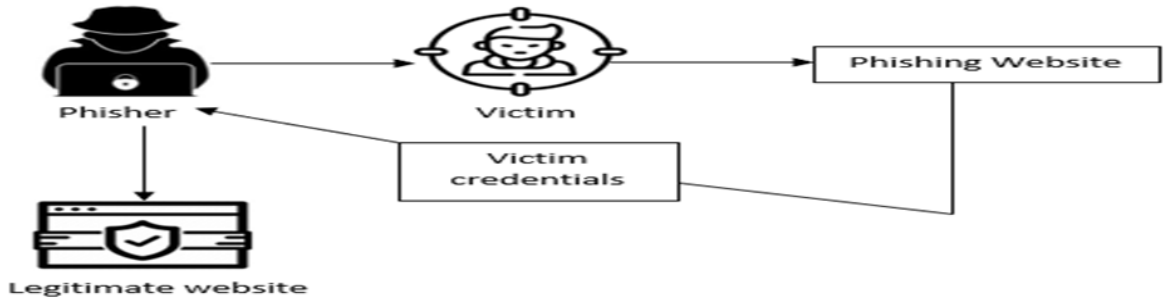

Figure 2.
Phishing Attack Process

Phishing is a kind of cyberattack where hackers trick users into providing personal information, including passwords, login credentials, and credit card numbers, usually for malicious intent (Khonji et al., 2013). Today, phishing assaults are one of the major threats to internet users, businesses, and service providers, especially when working remotely during COVID-19 (Sundaram et al., 2021). Cybercriminals pretend as

legitimate companies in phishing scams to deceive victims into providing personal information, such as bank account details and login credentials (Abdelhamid et al., 2014). Typically, attackers use this technique to trick unwary users by creating fake websites that closely resemble authentic ones. Figure 2 shows the steps involved in phishing attempts. To accurately detect phishing attacks, a variety of techniques are used, including content-based, heuristic-based, rule-based, list-based, and others. The objective of each technique is to accurately detect phishing assaults. A lot of studies have been performed to investigate possible ways to stop these kinds of attacks. To spot phishing attacks, some people start with the website itself (Abdelhamid et al., 2014).

Though various algorithms are made to detect phishing attacks at the email level, the phisher still attempts to persuade the victim to visit the fraudulent website (Dina & Manivannan, 2021). The fact that confirming each Uniform Resource Locator (URL) a user attempts to visit before granting access slows down website browsing and is ineffective against phishing attacks is one of the reasons this is favored. Secondly, when phishing attacks are detected at the email level, they are prevented early, making users safer. Malicious software may attack a user's device, for example, when they visit a malicious website.

Additionally, according to a recent study, phishing websites frequently disappear from the internet after 46 hours, yet phishing email records can be accessed whenever necessary (Barraclough et al., 2013). To increase phishing detection, several strategies have been proposed. These solutions often generate too many false positives due to their intrinsic inaccuracy. The success of some previous systems depends on data mining techniques that use a predefined set of attributes Saha et al. (2020). Other strategies that depend on black- or white-listings are useless because a phishing website only lasts a short time. In contrast, the majority of recent studies use content-based approaches and lexical URLs.

Phishing attack detection in real time has never been easy, but it's now more important than ever. In order to avoid being a victim of phishing, it is necessary to swiftly and accurately detect these assaults. However, because phishers' tactics are always evolving, it can be challenging to detect phishing attempts. The majority of security measures are easily overcome by attackers. Phishing tools are constantly being updated by hackers to produce websites that can get over nearly any kind of defense. Consequently, effective and efficient anti-phishing detection technologies are needed. Phishing attacks in cyberspace can be accurately detected by classification algorithms that use machine learning.

We propose an approach for detecting phishing URLs using machine learning. The problem of phishing attacks is addressed using the SVM-based machine learning technique. This SVM-based method helps to solve classification problems in an efficient manner. The effectiveness of the proposed approach is evaluated on a significant set of data obtained from Kaggle, Alexa, and PhishTank. A Deep Neural Network (DNN) is used to identify intrusions. The suggested model has improved accuracy in identifying various attack types.

LITERATURE REVIEW

This section is composed of two parts. The initial section discusses phishing detection techniques and related studies in the field. In the second section, the methods for detecting intrusion attacks are presented.

A. Phishing Detection Methods

As previously mentioned, Phishing is a type of social engineering attack. Phishers develop fake websites that look like the real ones in order to fool users into entering their login information. Therefore, this type might interact with the victims through a variety of media, including emails, short text messages (SMS), and smartphone platforms. The most common technique, though, is URL phishing. There have been numerous methods and strategies proposed to detect phishing.

CANTINA is a revolutionary platform that uses SDN-based deep machine learning to prevent phishing attacks. This study aims to increase classification accuracy through the use of the Deep Machine Learning with Cantina Approach, or DMLCA (Mourtaji et al., 2021) (Oest et al., 2019). SVM (Support Vector Machine) is a machine learning approach used to address the problem of phishing attacks. The TF-IDF (Term Frequency-Inverse Document Frequency) information retrieval method helps evaluate the webpage's contents by comparing, classifying, and retrieving documents. Results and simulation provide the maximum accuracy when compared to existing methods. Nevertheless, it fails when images are substituted for text (Adewole et al., 2019). Over time, it also has an efficiency problem because it relies on a third-party service's search engine, which might slow down identification.

The study (Oest et al., 2019) uses the DML Approach to classify phishing websites using a Feed Forward Neural Network. The authors propose classifying websites into three groups—phishing, suspicious, and trustworthy using a DML-based methodology. The data was collected from the 10,000-site Kaggle database, having ten features. After data collection and preprocessing, a feed-forward neural network is used to predict whether a web page is phishing. A Confusion Matrix was utilized to assess the proposed approach's effectiveness, with outcomes of 93.00% test and 95.00% training accuracy.

The authors of (Barraclough et al., 2013) suggested a simple deep learning approach for detecting phishing URLs, which allowed them to design a real-time, cost-effective phishing detection system. The suggested method has a 95.80% accuracy rate. A 2000-record dataset containing 1000 legitimate URLs and 1000 phishing URLs was utilized to evaluate the system's SVM algorithm. They demonstrated that a low-power integrated single-board computer can be used to accomplish the suggested strategy in real time. Nevertheless, the phishing website's information is not enough to assess the system. To test the system towards newly developed phishing efforts, a sizable phishing database is needed (Lohiya et al., 2021).

Additionally, a better machine learning (ML) prediction model is proposed to increase the effectiveness of anti-phishing measures (Bell & Komisarczuk, 2020)

(Tareen et al., 2022). An effective feature vector is generated by the predictive model's feature selection module. These features are extracted from the URL, webpage attributes, and webpage activity using the incremental component-based approach. The model is then given the feature vector that was produced in order to make predictions. Three criteria are included in the feature selection module (Bell & Komisarczuk, 2020). A 15-dimensional feature vector is used in the suggested method to train the SVM and NB models. To evaluate the model's accuracy, NB and SVM-based classification experiments were performed on datasets containing 2541 phishing and 25.000 actual sites (Barraclough et al., 2021). A novel strategy for detecting phishing attacks was presented by the research's authors (R. S. & Ravi, 2020), who combined ML algorithms with a range of features with heuristic-based,

online content-based, and list-based approaches. Evaluation techniques (metrics) based on the ANFIS (Adaptive Neuro-fuzzy inference system), NB, PART (Projective Adaptive Resonance Theory), J48, and JRip with features were used to test the results of the proposed strategy. Overall, more than 99.33% of classifiers were accurate. The highest-scoring program, PART, completed the task in 0.006 seconds with an accuracy of 99.33 percent. Studies have shown that the proposed technique can properly and quickly detect phishing websites. Due to this paper's limitation, the error rates were 0.66%, indicating that over-fitting is caused by certain noisy characteristics. The authors of (Masoud, M., et al., 2017) (Zouina, M., et al., 2017) (Kim, J., et al., 2017) studies included techniques for using machine learning models to identify phishing URLs. However, real-time use of these strategies is not possible. Phishing data is insufficient for these approaches' models. The suggested model must be tested on a sizable dataset. Additionally, these methods produce higher error rates. We tested the model on the large data set in this research. This has improved the accuracy of the learning process.

B. Intrusion Attacks Detection Techniques

The term intrusion refers to any type of unauthorized action that harms an information system. In order to detect intrusions, various methods have been proposed, including methods based on machine learning, deep machine learning, anomaly detection, and signatures. Deep Machine Learning techniques were utilized in studies (Sarker, I. et al., 2021) K. M., et al., 2021) (Topbas, A., et al., 2021) to identify intrusions. The internal understanding of the deep learning algorithms that trigger neurons is still lacking, though. In order to manage the results of both anomaly and abuse detection, Ozgur et al. 2021 presented a hybrid system that integrated both, along with a decision support system. In the anomaly detection strategy, they used the Self-Organizing Map (SOM) structure to mimic usual behavior, and in the misuse method, they classified several kinds using the decision tree methodology (Noor, S., et al., 2021). Every odd behavior is perceived as an attack.

One important aspect of cybersecurity technology is intrusion detection, which tracks and analyzes network data from many sources in order to spot malicious behavior. In recent years, deep learning-based deep neural network (DNN) techniques have been preferred methods for detecting malicious attacks (Subba, B., et al 2021). In order to reduce the false detection rate, earlier research used a range of machine learning methods to find attack patterns. Chung developed and assessed the intrusion detection model utilizing one or more of the numerous machine learning techniques, including Bayesian Classification, decision trees, and support vector machines (SVM) (Hussain et al., 2025) (Hussain, N., et al., 2024) (Akram et al., 2025) (Hussain, B., et al., 2024) ((Bhatti et al., 2023) (Nabeel et al., 2024) (Fakhar et al., 2022).

K-means clustering was the only method used in another study to identify fraudulent communications. The K-Means technique, which is frequently used with non-hierarchical clustering, was implemented by Shin to identify patterns in the data. Consequently, he found a parameter that might also identify a Witty worm attack and a DDoS attack (Zaland, Z., 2021). In order to reduce the risk of illegal attacks such as DDoS, (Tang, L., et al. 2021) offers a program that we personally guarantee is safe, and its access protocols. Table 1 summarizes previous research and identifies the most significant contributions in this area. Most studies have addressed the problem of phishing URL detection using traditional ML and DL-based methods, as is seen from most of the research. This study focuses attention on the current problems in the area, such as the dynamic nature of phishing techniques, the need for large and diverse

datasets, and the challenge of achieving high detection accuracy while maintaining sustainability and real-time accuracy.

Table I.
Literature Review Of Existing Methods

Ref	Models	Dataset	Description	Accuracy %	Limitation
(Tang & Mahmoud, 2021)	C4.5, OneRule, Conjunction Rule, eDRI, RIDOR, Bayes Net, SMO, AdaBoost	PhishTank, Millersmiles	The study compared several ML algorithms in terms of classification accuracy.	96.0%, 89.7%,89.5%, 94.2%, 93.5%, 93.7%, 94.3%, 92.1%	Without concentrating on the content of the prediction models produced, the study compared a number of algorithms in terms of classification accuracy.
(Aljofey et al., 2020)	character-level convolutional neural network (CNN)	PhishTank, Alexa, OpenPhish, spamhaus.org, techhelplist.com, isc.sans.edu	The algorithm efficiently classifies phishing URLs without depending on the content of the page, third-party services, or previous phishing knowledge.	95.2%	Data from phishing attempts is not enough to test the system. To evaluate the technique, a sizable phishing database is required.
(Rani et al., 2023)	Random Forest, Decision Tree	PhishTank	The study utilized Random Forest and Decision Tree models to identify phishing URLs from authentic URLs based on URL characteristics.	87.0%, 82.4%	The dataset and feature extraction process are not provided, and just a few evaluation metrics have been provided.
(Abad et al., 2023)	Decision Tree (DT), Random Forest (RF), Support Vector Machine (SVM), k-Nearest Neighbors (KNN)	650,000 URLs were utilized in the dataset (552,500 for training and 97,500 for testing).	The study used MR feature selection and four machine learning models with various instance selection techniques to categorize phishing URLs.	90.6%, 93.4%, 92.3%, 87.6%	Phishing URL classification is challenging; model sensitivity to instance-selection techniques varies, and SVM computation is expensive.
(Cherradi & El Mahajer, 2025)	Logistic Regression (LR), Support Vector Machine (SVM), Naive Bayes (NB), Decision Tree (DT)	Kaggle dataset	Four machine learning models were used in the study to classify malicious URLs. They were tested both before and after hyperparameter tuning, and they were made available through a web application built with FastAPI.	0.915%, 0.908%, 0.924%, 0.918%	Limited generalization to unknown or multilingual material; depends on static traits without adaptive learning; susceptible to obfuscation, redirection, and zero-day URLs.
(Wang, 2025)	Logistic regression (LR), decision trees	Kaggle dataset	The study evaluated three machine learning classifiers (LR, DT, and RF) for detecting malicious	0.877%, 0.913%, 0.942%	Limited algorithm diversity, incomplete dataset description, and lack of phishing-specific analysis.

(DT), and Random Forest (RF)	URLs and analyzed the value of features and model explainability using SHAP values.

Although previously proposed phishing detection methods work well, they have some issues. They frequently rely on minimal data, have a high rate of false alarms, and find it difficult to adjust to novel phishing techniques. By leveraging a large dataset and applying machine learning techniques with improved feature selection techniques to boost accuracy and flexibility, our suggested method aims to address these problems.

PROPOSED METHODOLOGY

This chapter presents the research methodology. We suggested a machine learning-based technique to detect phishing websites. Phishing attack-related issues are addressed using the SVM technique. A supervised machine learning technique called SVM helps in the effective resolution of classification issues. The usefulness of the suggested method is assessed using a sizable dataset sourced from PhishTank, Alexa, and Kaggle. The DNN technique is used to detect intrusions. In order to detect zero-day threats and enhance network security, DNN has recently been integrated with intrusion detection systems (IDS).

The SVM classifier takes URLs as input. Following the URL analysis, features are extracted initially, followed by TF-IDF features. Machine learning algorithms are trained using these feature sets. The features are then used as training data in machine-learning models after they have been retrieved. A variety of features are gathered, including search engine features, lexical, WHOIS, and keywords. To use DNN to detect the intrusion, the NSL-KDD dataset and the output of the phishing URLs are taken as input. The Scikit-learn or Keras dataset was used to train the model. The process of phishing detection is illustrated in Figure 4. The general approach for detecting intrusions and phishing attempts is shown in Figure 3.

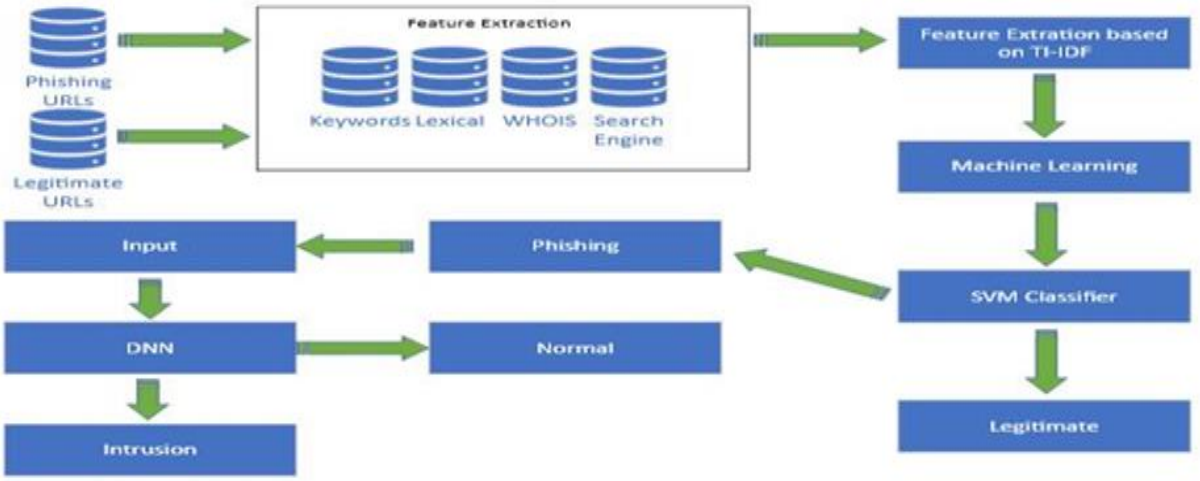

Figure 3.
Overall Phishing and Intrusion Detection Mechanism

We describe the execution and design of phishing URLs obtained from different sources during the phishing phase. The SVM classifier will be provided the URLs as input. Initially, we analyzed the URLs to check for duplicate or null data. Null values will be removed, and duplicate data will be eliminated. Features are first extracted following URL analysis, followed by TF-IDF features. In order to train the machine learning algorithms, several feature sets are utilized.

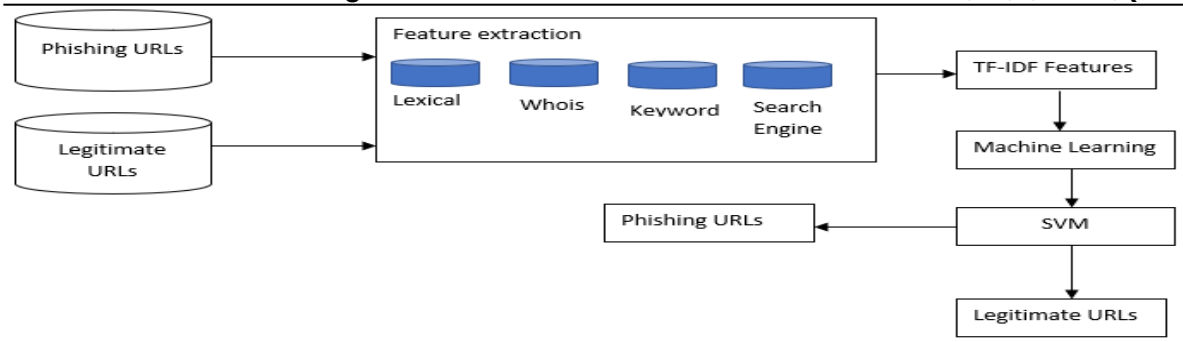

Figure 4.
Phishing detection mechanism

After the features are retrieved, they are used as training data for machine learning. A machine learning model, which includes SVM, is used to train the dataset. The accuracy of these models using the testing dataset is then used to assess their effectiveness. The testing dataset was then pre-processed, and features like Count and TF-IDF features were retrieved. Once more, the models are trained using the retrieved features as training data. Finally, we compare the accuracy rate of the two machine learning techniques. The machine learning algorithms will categorize the URLs as benign or phishing after they have been analyzed.

A. Dataset

Models were trained on Kaggle and PhishTank datasets. PhishTank is a database of phishing URLs that the business keeps up to date. The Kaggle data set contains a selection of URs that have been categorized as legitimate or phishing. It includes ten features and URLs from 5,49346 different websites. 5,49346 records are in the database. The label column, also known as the prediction column, is divided into two sections. A good URL is one that doesn't contain any malicious content or phishing scams. These websites are phishing schemes that have been labeled as bad, and their URLs are fake. There are no blanks in the data collection.

B. Pre-processing

Data Preprocessing and data distribution are necessary to ensure that the model will perform well with new data. An uneven dataset may produce biased predictions, decreasing accuracy in previously experienced occurrences. Stratified sampling was used to balance class representations in order to solve this problem. To improve both the quality and the usefulness of the data for the model, preprocessing techniques such as encoding, feature selection, and normalization were applied.

Finding and removing duplicates from the data collection process is essential to ensuring the accuracy and integrity of the data. This study identified and removed 42151 duplicate entries from the dataset.

The TF-IDF vectorizer is specifically designed to encode URLs that have been eliminated because they are invalid. By using the word-frequency weights from the vectorizer, the model is able to understand the data more accurately. The URLs can be fed into the SVM model after they have been encoded. Using the TF-IDF approach, we may convert our data into a collection of features that we can utilize to develop a word vectorizer. The maximum feature count is set at 1000, and a data frame with the unigram information is also being produced.

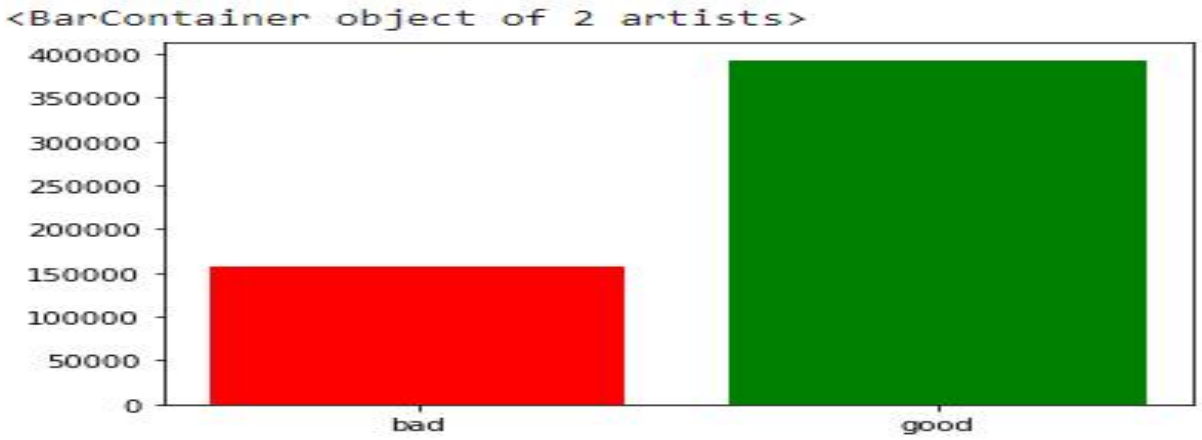

Figure 5:
The overall entries and their respective ratios

TF-IDF Vectorizer and Count Vectorizer are used to extract features from the URLs, and finally, a 1 is produced from the data frame elements with values larger than 0.

C. Feature Extraction

After the data has been cleaned and preprocessed, feature extraction begins. Figure 6 shows the features that are collected, including search engine parameters, keywords, Lexical, and Whois. Keywords, search engine properties, WHOIS, and lexical properties will all be extracted from the collected data. Text-based characteristics are known as lexical features. Word length, word count, word frequency, and vocabulary preference are examples of lexical features.

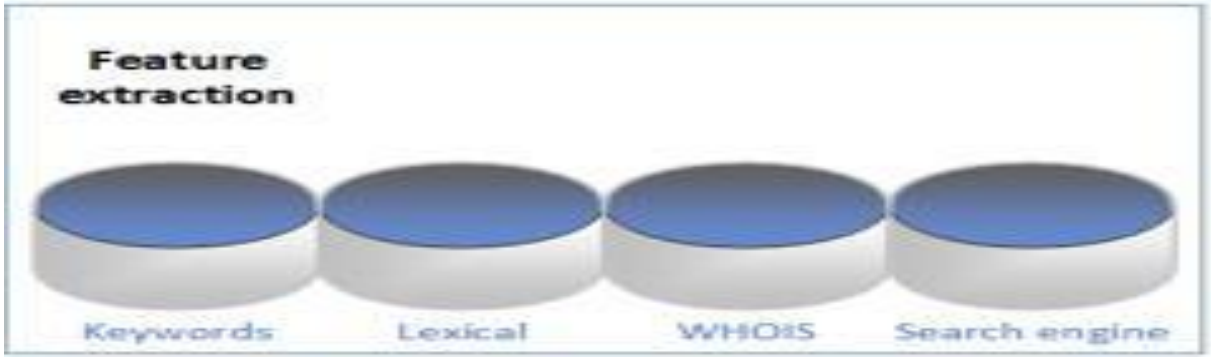

Figure 6.
Feature Extraction

The website domain information (WHOIS) is the source of many useful website features. Some of the characteristics of WHOIS include details about the domain's age, registrar, registrant, and name server. Index-based functionalities are referred to as "search engine features" in search engines.

D. Machine Learning

Natural language processing, energy production, image processing and computer vision, computational finance, automotive, and aerospace are some of the key techniques used in the machine learning approach. A machine learning algorithm is used to create a mathematical model of the sample data, also known as training data. With the help of features identified in phishing URLs, our suggested method distinguishes between them. The analysis of various machine learning methods is shown in Figure 7.

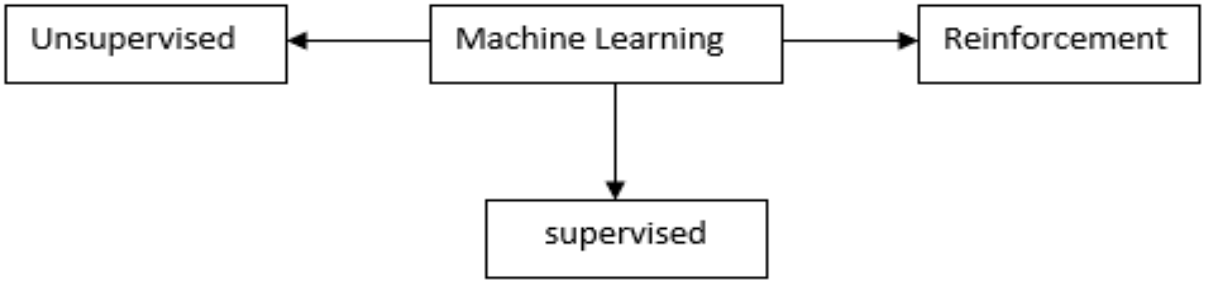

Figure 7.
Machine Learning Methods

1)SUPPORT VECTOR MACHINE (SVM)

The support vector machine is a supervised learning model that analyzes data for regression and classification. This non-probabilistic binary linear classifier can be used to allocate new data from a single category. After a sizable amount of data has been classified, the SVM classifier is evaluated.

2)TERM FREQUENCY INVERSE DOCUMENT FREQUENCY (TF-IDF)

A numerical metric called TF-IDF is used to assign a value to each word in a document according to its appearance in the given collection of texts. There are two fundamental parts to it. The number of times a word appears in a document is its frequency (TF).

There is an inverse relationship between the IDF and the number of publications that utilize the word. The sum of the TF and IDF results is the TF-IDF score. The word's importance to the document is shown by its TF-IDF score. Term frequency (TF) and inverse document frequency (IDF) data are combined to create the TF-IDF score (IDF). The following is the mathematical calculation for TF-IDF:

$$TF - IDF\ (T_s, D_e, D_e) \times\ IDF\ (T_s, D_{c)}) \quad (1)$$

$$IDF(T_s, D_c) = log\ \left(\frac{|D_c|}{1+|\{D_e \in D_c : T_s \in D_e\}|}\right) \quad (2)$$

TF-IDF represents the frequency with which specific terms occur in documents, where T stands for "terms in Word document," De for "each document," and Dc for "the collection of documents." The above formula can be used to calculate the inverse document frequency. |De, Dc, Ts, and De| are variables that specify the total number of times that must appear in all papers. When compared to other embedding methods, TF-IDF offers additional benefits. TF-IDF prioritizes less common but more informative phrases to help differentiate between phishing and legitimate URLs, while BoW examines all words equally. On the other hand, TF-IDF is computationally cheap and suitable for real-time phishing detection, unlike other word embedding that require big training datasets and a lot of processing power.

TF-IDF is ideal for detecting distinctive patterns in phishing URLs, which is why we selected it for the phishing URL detection challenge. It can efficiently assess the importance of words in each corpus. Our approach, which utilizes TF-IDF, optimizes the performance of Support Vector Machines (SVM) by concentrating on important URL features that are suggestive of phishing attacks. As a result, detection performance is enhanced with increased efficiency and accuracy. We can convert our data into a collection of features using the TF-IDF method, which we can then use to create a word vectorizer. Additionally, a data frame containing the unigram

information is generated, and the maximum feature count is set at 1000. Lastly, data frame elements with values greater than 0 are being used to generate a 1. The Count vectorizer and the TF-IDF vectorizer are utilized to extract features from the URLs.

	00	0001	01	02	03	04	05	06	07	08	...	yelp	york	you	young	your	youtube	za	zimbio	zip	zoominfo
0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	...	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
1	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	...	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
2	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	...	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
3	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	...	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
4	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	...	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
507190	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	...	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
507191	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	...	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
507192	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	...	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
507193	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	...	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
507194	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	...	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0

507195 rows × 1000 columns

Figure 8.
TF-IDF features Extraction

E. Deep Neural Network (DNN)

The second phase of the study simulates the deep learning approach and assesses how well it handles intrusion attack detection. In artificial intelligence research, neural networks are one field that aims to mimic human brain activity by mimicking the human nervous system, namely, its capacity to recognize and fix errors. Neuronal networks are made up of many neurons. Figure 9 shows a neuron block diagram.

The ability of neural networks to evaluate attack characteristics and differentiate components that are different from those under study is one of their key features (Ahmed et al. 20250) (Songpeng et al., 2025) (Bhatti et al., 2025). In applications like object detection and natural language processing, which include picture classification, deep neural networks (DNNs) perform exceptionally well. In order to identify and categorize network traffic intrusions, DNNs have been applied in the intrusion detection field. The suggested architecture for intrusion detection consists of two components: feature extraction and classification. In order to extract meaningful features from the input data, the DNN training process begins with pre-processing. The collected features are used in training deep neural networks to classify the input data into discrete intrusion categories.

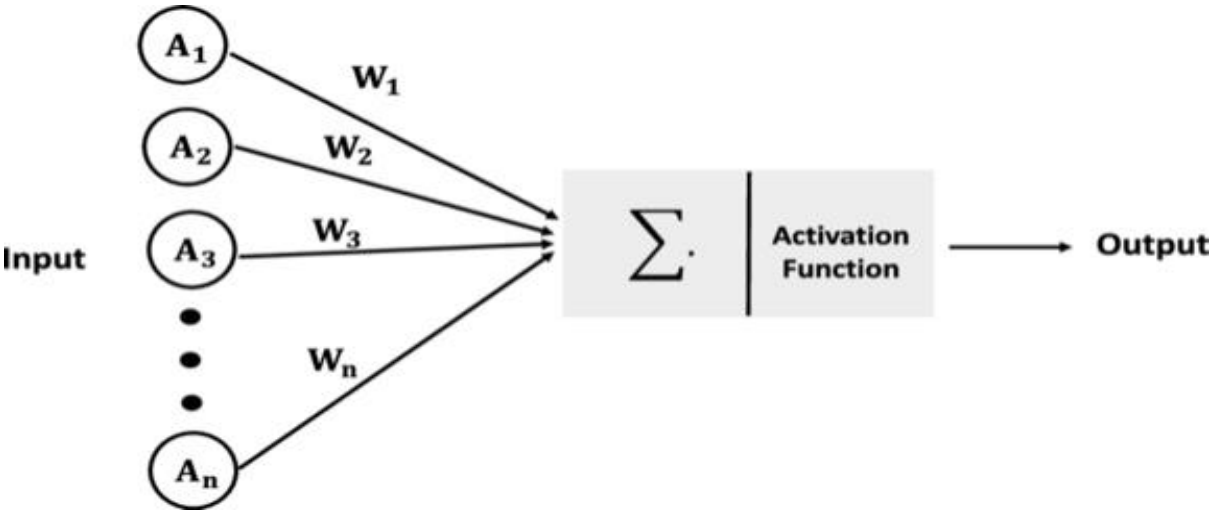

Figure 3.
Schematic block of neuron

Three layers comprise a DNN: input, hidden, and output. As illustrated in Figure 10.

1) INPUT LAYER

The features of the data that will be used to train and design a model are represented by this layer. These features are intrusions of various kinds in our situation. Subsequently, the DNN would try to abstract or generalize these attributes.

2)HIDDEN LAYER

Depending on the data entering the model, this layer can be thought of as a sequence of on-off switch nodes that are triggered in particular combinations. A prediction or output is then generated by the values for these switches. Any number of nodes can be included in the hidden layer, and the neural network model itself may have several hidden layers. The model gets increasingly complicated as a buried layer's node count rises. We used nearly five hidden layers in our case, which serve as a basis for our predictions.

3)OUTPUT LAYER

The model's final predictions are represented by the output layer. The output layer will have a single node if the model predicts a numerical value, such as a product's price. On the other hand, the output layer will have a large number of nodes if the model is predicting whether something belongs to one of several categories. Every node will stand for a different category. Upon grouping them, we obtain several nodes that belong to the respective category.

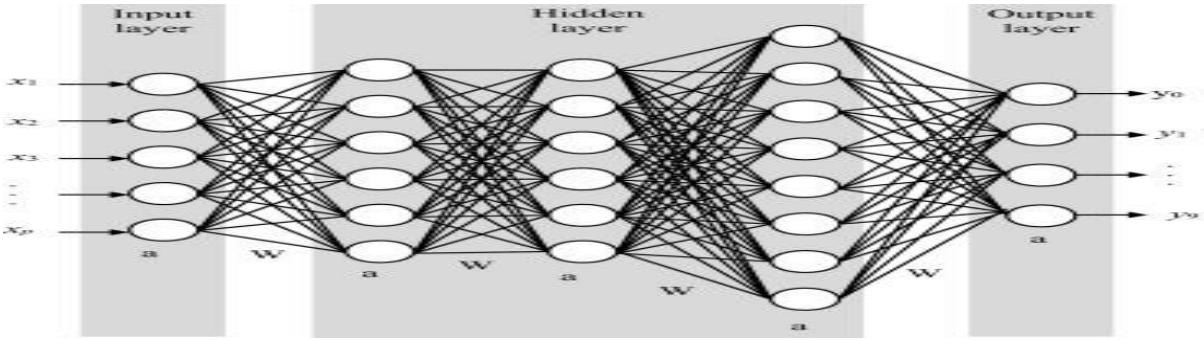

Figure 10.
DNN Components

4)DNN MODEL IMPLEMENTATION

A DNN architecture can be applied to intrusion detection and classification in a number of ways. The basic guideline states that the number of nodes in the first layer should be twice that of the features. The model has five levels, with the first layer having 244 nodes and the next layers having 122, 61, 30, and 11 nodes. The output layer utilizes the softmax activation function for multi-class classification, and the rectified linear unit (ReLU) activation function is used in the hidden layers to avoid vanishing gradients. After preprocessing the dataset into feature-target pairs for simpler classification, the target values were encoded one-hot. Using the stochastic gradient descent (SGD) approach, the model was trained across 50 epochs with a batch size of 2500. In order to minimize errors, the model iteratively adjusted weights and biases via backpropagation. To train and assess the model, the pre-processed dataset is used. The model's F1 score, recall, accuracy, and precision are compared to the most advanced approaches.

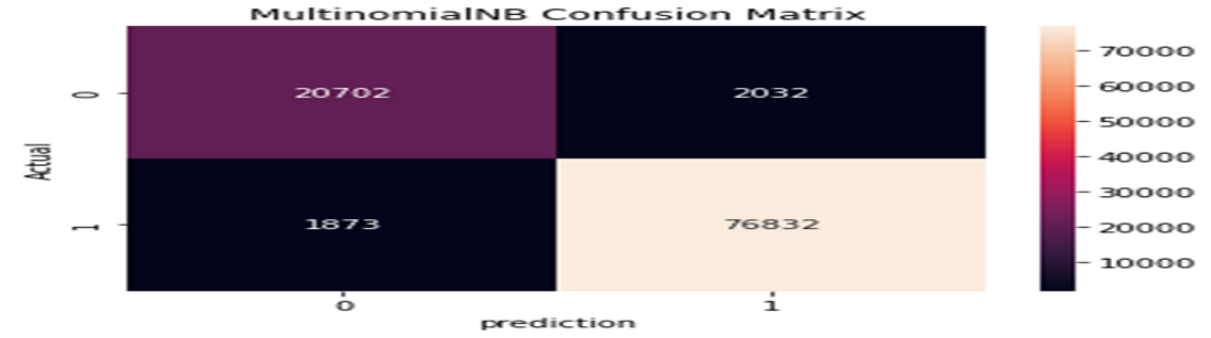

Figure11.
SVM Performance

RESULTS AND DISCUSSION

The experimental setup and outcomes of the suggested method for phishing URLs and intrusion detection using SVM and DNN models are covered in this part. Model performance is thoroughly assessed and compared through the use of key measures such as F1 score, accuracy, and precision.

A. Experimental Environment

This study's model was developed on Google Collaboratory with GPU support utilizing TensorFlow, ML frameworks, and Python.

B. SVM Results

In order to prevent phishing attempts, SVM-based machine learning techniques were used. Model performance is assessed using both training and testing accuracy. Testing accuracy evaluates the model's ability to generalize to new data, whereas training accuracy determines how well the model learns from the provided data. The F1-score is a statistic that combines precision and recall to assess how well a model performs on a given dataset. In general, SVM combined with the TF-IDF algorithm outperforms SVM individually in terms of accuracy, precision, and F1 score for phishing detection.

Table 2.
Svm With Tf-Idf Result Comparison With Svm Alone

Evaluation Metric	SVM	SVM with TF-IDF
Accuracy	96.3	97.3
Precision	96.2	97.2
F1 Score	96.3	97.3

Finding the relevant terms in a URL and assigning them weights is the aim of the TF-IDF algorithm. This can enhance SVM's capacity to distinguish between legitimate and phishing websites. Furthermore, SVM with the TF-IDF technique can capture subtler aspects of the URLs than conventional SVM. Figures 11 and 12 illustrate the comparative and evaluation performance of Multinomial NB and SVM, respectivel.

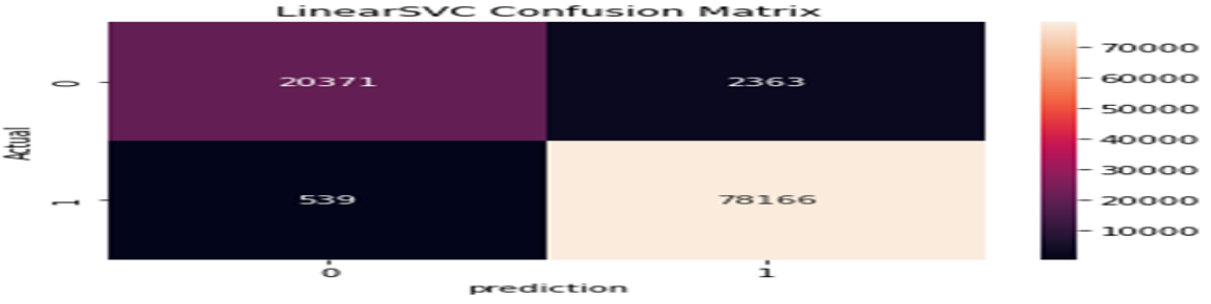

Figure 12.
Multinomial NB Performance

Table II demonstrates that SVM with TF-IDF performs better than SVM when comparing the results of the two models. Table III illustrates the outcomes of comparing the suggested SVM with other approaches.

Table 3.
Proposed Svm Comparison With Different Algorithms

No	Algorithm	Accuracy	Precision	F1 Score
1	SVM with TFIDF (proposed)	97.3%	97.3%	97.3%
(Ali, Shahbaz, et al. 2019)	SVM with Entropy	91.30%	90.53%	90.91%
(Elkouay, M., et al., 2022)	SVM with N-Gram	87.10%	87.17%	87.13%
(Hashem, 2013)	SVM with PCA	89.20%	89.23%	89.21%
Al-Sabbagh, H., et al., 2024)	SVM with K-Means	90.90%	90.80%	90.85%

C. DNN Results

The second part of the study concentrated on using a Deep Neural Network (DNN) for intrusion detection. In the model, every neuronal layer has a unique activation function. Performance metrics such as accuracy, precision, recall, and F1 score were utilized to evaluate the model compared to existing approaches. In terms of intrusion detection and classification, the experiment's findings demonstrate that the DNN model outperforms baseline systems. By testing the proposed model on a benchmark dataset and comparing it with other models, the most efficient intrusion detection technique was determined. Throughout the training phase, backpropagation demonstrated better results than conventional deep neural network training techniques. The suggested method categorizes attack methods into four major categories: R2L (Remote to Local Attack), U2R (User to Root Attack), Probe (Probing Attack), and DOS (Denial of Service Attack). The intrusion descriptions and classifications are presented in Tables IV and V.

Table 4.
Types of attacks and kinds of intrusions discovered in training and testing dataset

Category	Training Set	Testing Set
DoS	Back, land, Neptune, pod, smurf, teardrop	Back, land, Neptune, pod, smurf, teardrop, mailbomb, process table, udpstorm, apache2, worm
R2L	Fpt-write, guess-password, imap, multihop, phf, spy, warezclient, warezmaster	Fpt-write, guess-password, imap, multihop, phf, spy, warezclient, warezmaster, xlock, xsnoop, snmpguess, snmpgetattack, httptunnel, sendmail, named
U2R	Buffer-overflow, loadmodule, perl, roolkit	Buffer-overflow, loadmodule, perl, roolkit, sqiattack, xterm, ps
Probe	Ispweep, nmap, portsweep, satan	Ispweep, nmap, portsweep, satan, mscan, saint

When tested on validation data, the proposed model achieved 96% accuracy and 97.873% F1 score, as shown in Figure 13. Although the model performed well overall, it struggled to detect "nmap" and a few other attacks due to their subtlety and similarity to harmless traffic patterns.

Table 5.
Description Of Intrusions

Feature Name	Description
Duration	Length (number of seconds) of the connection
Protocol _type	Type of the protocol, e.g., tcp, udp
Src_bytes	Number of databytes from source to destination
dst-bytes	Number of databytes from destination to source
Srv_count	Number of connections to the same service as the current connection in the past two seconds
Dst_host_same_src_port_rate	Number of connections that were to the same source port

D. Proposed Work Benefits

The advantages of the proposed approach are as follows:

1) NO THIRD-PARTY SERVICES ARE REQUIRED

The proposed method is not dependent on any third-party services since it does not harvest features based on third parties.

2) FAST COMPUTATION

There is no need to visit the website because only URL-based attributes are extracted. Consequently, the time required to extract and process the features is reduced.

3) INDEPENDENT OF DRIVE BY DOWNLOAD

There is no chance of viruses being downloaded from the web pages because the actual website is not visited in order to extract features.

CONCLUSION

This paper evaluated the application of Deep Neural Networks for intrusion detection in Software-Defined Networking environments and represents the effectiveness of combining Support Vector Machines with Term Frequency-Inverse Document Frequency for phishing detection.

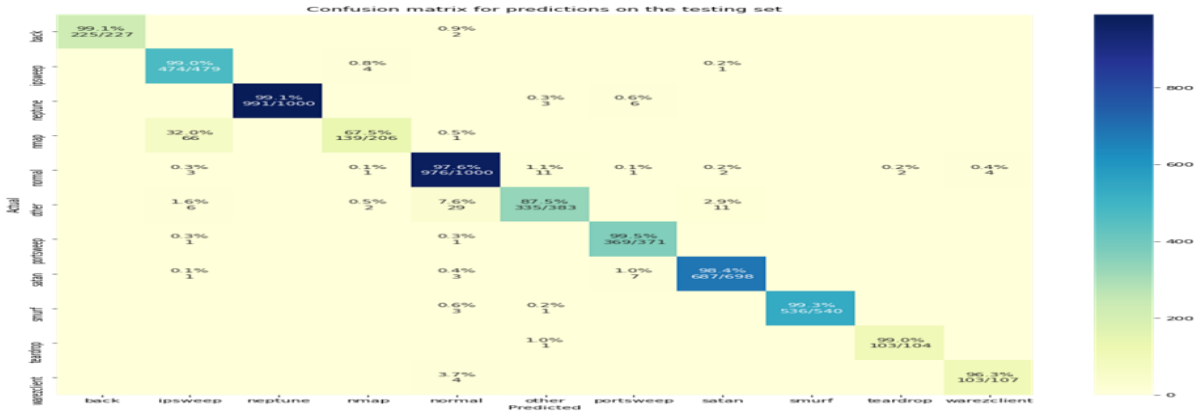

Figure 13.
Prediction Set Confusion Matrix

We suggested a method that uses machine learning (ML) to categorize phishing websites. Using SVM along with TF-IDF, the machine learning method addresses

phishing URL detection. We find that SVM and TF-IDF together improve accuracy and efficiency when used to detect phishing URLs compared to SVM alone. Deep neural networks (DNNs) outperform other deep and machine learning models in detecting intrusions. DNNs are very good at identifying outlier occurrences and complex patterns. The diversity of data sources that DNNs can process, such as system logs, network traffic, and user behavior, is too much for conventional models to handle. A unique advantage of DNNs is their capacity to detect patterns and abnormalities that traditional approaches often ignore. This study thoroughly evaluates their application in intrusion detection. Our findings indicate that DNNs are an excellent option due to their superior performance and faster processing.

Further research on DNNs for intrusion detection may reveal more, such as adding more complex structures and exploring the potential for transfer learning. To find out how DNNs might be used in conjunction with other machine-learning techniques to increase IDS efficacy, more research is also essential.

Several future studies are recommended in order to overcome the limitations that have been identified and enhance the system's functionality. First, to stop people from visiting those phishing URLs, incorporate a blacklist mechanism into the suggested work. To stop users from falling victim to phishing and intrusion assaults, a prevention-based strategy utilizing a blacklist mechanism will be implemented. The development of adaptive models to enhance the system's detection of zero-day assaults is another area of future research. These models don't require complete retraining because they dynamically add new phishing and intrusion patterns to their knowledge base. In order to reduce processing overhead, hierarchical network topologies and efficient routing protocols, such as shortest path routing, should be combined. Additionally, the throughput and latency of SDN should be increased. Finally, compare the performance of TF-IDF with various ML and DL models, including CNN, LSTM, and Random Forest. Exploring more efficient packet processing methods that lower latency and improve real-time detection capabilities through the use of hardware acceleration and sophisticated DL algorithms is essential.

DECLARATIONS

Acknowledgement: We appreciate the generous support from all the contributor to the research and their different affiliations.
Funding: No funding body in the public, private, or nonprofit sectors provided a particular grant for this research.
Availability of data and material: In the approach, the data sources for the variables are stated.
Authors' contributions: Each author participated equally in the creation of this work.
Conflicts of Interest: The authors declare no conflict of interest.
Consent to Participate: Yes
Consent for publication and Ethical approval: Because this study does not include human or animal data, ethical approval is not required for publication. All authors have given their consent.

REFERENCES

Abad, S., Gholamy, H., & Aslani, M. (2023, September). Classification of malicious URLs using machine learning. Sensors, 23(18), 7760. doi: 10.3390/s23187760

Abdelhamid, N., Ayesh, A., & Thabtah, F. (2014, October). Phishing detection based associative classification data mining. Expert Systems with Applications, 41(13), 5948–5959. doi: 10.1016/j.eswa.2014.03.019

Adewole, K. S., Akintola, A. G., Salihu, S. A., Faruk, N., & Jimoh, R. G. (2019). Hybrid rule-based model for phishing URLs detection. Lecture Notes of the Institute for Computer Sciences, Social Informatics and Telecommunications Engineering (LNICST, 285), 119–135. doi: 10.1007/978-3-030-23943-5_9

Aftab, F., et al. (2023). A comprehensive survey on sentiment analysis techniques. International Journal of Technology, 14(6), 1288–1298. doi: 10.14716/ijtech.v14i6.6632

Ahmad, M., Bazai, S. U., Hussain, S., Ashirova, A. I., Erkaboy Ugli, Y. J., & Bhatti, U. A. (2025). Predicting household electricity consumption using machine learning and big data analytics. In Proceedings of the 2025 IEEE 2nd International Conference on Deep Learning and Computer Vision (DLCV) (pp. 1–7). IEEE. https://doi.org/10.1109/DLCV65218.2025.11088841.

Akram, M., et al. (2025). EEMLCR: Energy-efficient machine learning-based clustering and routing for wireless sensor networks. IEEE Access, 13, 70849–70871. https://doi.org/10.1109/ACCESS.2025.3562368

Ali, S., Shahbaz, M., & Jamil, K. (2019, December). Entropy-based feature selection classification approach for detecting phishing websites. 2019 13th International Conference on Open Source Systems and Technologies (ICOSST 2019), 48–53. doi: 10.1109/ICOSST48232.2019.9044042

Aljofey, A., Jiang, Q., Qu, Q., Huang, M., & Niyigena, J. P. (2020, September). An effective phishing detection model based on character level convolutional neural network from URL. Electronics, 9(9), 1514. doi: 10.3390/electronics9091514

Al-Sabbagh, A., Hamze, K., Khan, S., & Elkhodr, M. (2024, September). An enhanced K-means clustering algorithm for phishing attack detections. Electronics, 13(18), 3677. doi: 10.3390/electronics13183677

APWG. (2025, July 17). Phishing activity trends report. APWG. Available: https://apwg.org/trendsreports/

Barraclough, P. A., Fehringer, G., & Woodward, J. (2021, May). Intelligent cyber-phishing detection for online. Computers & Security, 104. doi: 10.1016/j.cose.2020.102123

Barraclough, P. A., Hossain, M. A., Tahir, M. A., Sexton, G., & Aslam, N. (2013, September). Intelligent phishing detection and protection scheme for online transactions. Expert Systems with Applications, 40(11), 4697–4706. doi: 10.1016/j.eswa.2013.02.009

Bazai, S. U., & Jang-Jaccard, J. (2020, October). In-memory data anonymization using scalable and high performance RDD design. Electronics, 9(10), 1732. doi: 10.3390/electronics9101732

Bazai, S. U., Jang-Jaccard, J., & Alavizadeh, H. (2021, February). A Novel Hybrid Approach for Multi-Dimensional Data Anonymization for Apache Spark. ACM Trans. Priv. Secur. 25, 1, Article 5, 25 pages. https://doi.org/10.1145/3484945

Bazai, S. U., Jang-Jaccard, J., & Alavizadeh, H. (2021, March). Scalable, high-performance, and generalized subtree data anonymization approach for Apache Spark. Electronics, 10(5), 589. doi: 10.3390/electronics10050589

Bazai, S. U., Jang-Jaccard, J., & Wang, R. (2017, May). Anonymizing k-NN classification on MapReduce. International Conference on Mobile Networks and Management, 364–377. Springer. doi: 10.1007/978-3-319-90775-8_29

Bazai, S.U., Jang-Jaccard, J., Zhang, X. (2017, June). A Privacy Preserving Platform for MapReduce. Applications and Techniques in Information Security. ATIS. doi: 10.1007/978-981-10-5421-1_8

Bell, S., & Komisarczuk, P. (2020, February). An analysis of phishing blacklists: Google Safe Browsing, OpenPhish, and PhishTank. ACM International Conference Proceedings Series. doi: 10.1145/3373017.3373020

Bhatti, U. A., Bazai, S. U., Hussain, S., Fakhar, S., Chin Soon Ku, Marjan, S., Yee, P. L., & Liu, J. (2023). Deep learning-based trees disease recognition and classification using hyperspectral data. Computers, Materials & Continua, 77(1), 681. doi: 10.32604/cmc.2023.037958

Bhatti, U. A., et al. (2025). ABi-LT: Intelligent medical decision model clinical decision support system. In Proceedings of the 2025 2nd International Conference on Electronic

Engineering and Information Systems (EEISS) (pp. 1–7). IEEE. https://doi.org/10.1109/EEISS65394.2025.11085933

Cherradi, M., & El Mahajer, H. (2025, May). Malicious URL detection using machine learning techniques. International Journal of Data Informatics and Intelligent Computing, 4(2), 41–52. doi: 10.59461/ijdiic.v4i2.187

Dina, A. S., & Manivannan, D. (2021, December). Intrusion detection based on machine learning techniques in computer networks. Internet of Things, 16, 100462. doi: 10.1016/j.iot.2021.100462

Elkouay, A., Moussa, N., & Madani, A. (2022). Classification of URLs using N-gram machine learning approach. Lecture Notes in Networks and Systems, 489, 85–99. doi: 10.1007/978-3-031-07969-6_7

Fakhar, S., Baber, J., Bazai, S. U., Marjan, S., Jasinski, M., Jasinska, E., Chaudhry, M. U., Leonowicz, Z., & Hussain, S. (2022). Smart classroom monitoring using novel real-time facial expression recognition system. Applied Sciences, 12(23), 12134. https://doi.org/10.3390/app122312134

Hashem, S. H. (2013). Efficiency of SVM and PCA to enhance intrusion detection system. Journal of Asian Scientific Research, 3(4), 381–395. Available: http://aessweb.com/journal-detail.php?id=5003

Hussain, S., Bazai, S. U., Ghafoor, M. I., Noor, M., & Marjan, S. (2024). Predicting air quality using temporal features: An analysis using Apache Spark and machine learning. In Proceedings of the 2024 5th International Conference on Innovative Computing (ICIC) (pp. 1–6). IEEE. https://doi.org/10.1109/ICIC63915.2024.11116550

Hussain, S., Bazai, S. U., Qadir, S., Marjan, S., Ghafoor, M. I., & Pervaiz, P. (2025). Sentiment analysis of Balochi text using deep learning. VAWKUM Transactions on Computer Sciences, 13(1), 190–200. https://doi.org/10.21015/vtcs.v13i1.2081

Hussain, S., Nadeem, M., Baber, J., et al. (2024). Vulnerability detection in Java source code using a quantum convolutional neural network with self-attentive pooling, deep sequence, and graph-based hybrid feature extraction. Scientific Reports, 14, 7406. https://doi.org/10.1038/s41598-024-56871-z

K. M., S. A., Y. S., J. D., & M. S. (2021). Detection of intrusion attacks using neural networks. Repository of Kharkiv National University of Economics. Available: http://repository.hneu.edu.ua/handle/123456789/26824

Khonji, M., Iraqi, Y., & Jones, A. (2013, March). Enhancing phishing e-mail classifiers: A lexical URL analysis approach. International Journal of Information Security Research, 3(1), 236–245. doi: 10.20533/ijisr.2042.4639.2013.0029

Kim, J., Shin, N., Jo, S. Y., & Kim, S. H. (2017, March). Method of intrusion detection using deep neural network. 2017 IEEE International Conference on Big Data and Smart Computing (BigComp 2017), 313–316. doi: 10.1109/BIGCOMP.2017.7881684

Korkmaz, M., Sahingoz, O. K., & Diri, B. (2020, July). Detection of phishing websites by using machine learning-based URL analysis. 2020 11th International Conference on Computing, Communication and Networking Technologies (ICCCNT 2020). IEEE. doi: 10.1109/ICCCNT49239.2020.9225561

Lohiya, R., Thakkar, A., & Thakkar, A. (2021). Intrusion detection using deep neural network with antirectifier layer. Lecture Notes in Networks and Systems, 187, 89–105. doi: 10.1007/978-981-33-6173-7_7

Masoud, M., Jaradat, Y., & Ahmad, A. Q. (2017, February). On tackling social engineering web phishing attacks utilizing software defined networks (SDN) approach. 2016 2nd International Conference on Open Source Software Computing (OSSCOM 2016). doi: 10.1109/OSSCOM.2016.7863679

Mourtaji, Y., Bouhorma, M., Alghazzawi, D., Aldabbagh, G., & Alghamdi, A. (2021, January). Hybrid rule-based solution for phishing URL detection using convolutional neural network. Wireless Communications and Mobile Computing, 2021(1), 8241104. doi: 10.1155/2021/8241104

Nabeel, S. M., Bazai, S. U., Alasbali, N., et al. (2024). Optimizing lung cancer classification through hyperparameter tuning. Digital Health, 10. https://doi.org/10.1177/20552076241249661

Noor, S., Bazai, S. U., Ghafoor, M. I., Marjan, S., Akram, S., & Ali, F. (2023, March). Generative adversarial networks for anomaly detection: a systematic literature review. In 2023 4th International Conference on Computing, Mathematics and Engineering Technologies (iCoMET) (pp. 1-6). IEEE. doi: 10.1109/iCoMET57998.2023.10099175.

Oest, A., Safaei, Y., Doupe, A., Ahn, G. J., Wardman, B., & Tyers, K. (2019, May). PhishFarm: A scalable framework for measuring the effectiveness of evasion techniques against browser phishing blacklists. Proceedings of the IEEE Symposium on Security and Privacy, 1344–1361. doi: 10.1109/SP.2019.00049

R. S., E., & Ravi, R. (2020, March). A performance analysis of software defined network based prevention on phishing attack in cyberspace using a deep machine learning with CANTINA approach (DMLCA). Computer Communications, 153, 375–381. doi: 10.1016/j.comcom.2019.11.047

Rani, L. M., Foozy, C. F. M., & Mustafa, S. N. B. (2023, May). Feature selection to enhance phishing website detection based on URL using machine learning techniques. Journal of Soft Computing and Data Mining, 4(1), 30–41. doi: 10.30880/jscdm.2023.04.01.003

Saha, I., Sarma, D., Chakma, R. J., Alam, M. N., Sultana, A., & Hossain, S. (2020, August). Phishing attacks detection using deep learning approach. Proceedings of the 3rd International Conference on Smart Systems and Inventive Technology (ICSSIT 2020), 1180–1185. doi: 10.1109/ICSSIT48917.2020.9214132

Sarker, I. H. (2021, May). Machine learning: Algorithms, real-world applications and research directions. SN Computer Science, 2(3), 160. doi: 10.1007/s42979-021-00592-x

Singh, G., & Khare, N. (2022). A survey of intrusion detection from the perspective of intrusion datasets and machine learning techniques. International Journal of Computer Applications, 44(7), 659–669. doi: 10.1080/1206212x.2021.1885150

Songpeng, G., et al. (2025). Refining KNN classification of hyperspectral images through PCA and hyperparameter optimization techniques. In Proceedings of the 2025 2nd International Conference on Electronic Engineering and Information Systems (EEISS) (pp. 1–6). IEEE. https://doi.org/10.1109/EEISS65394.2025.11085598

Subba, B., Biswas, S., & Karmakar, S. (2016, March). Intrusion detection systems using linear discriminant analysis and logistic regression. 12th IEEE International Conference on Electronic, Energy, Environment, Communication, Computer, and Control (E3-C3), INDICON 2015. doi: 10.1109/INDICON.2015.7443533

Sundaram, K. M., Sasikumar, R., Meghana, A. S., Anuja, A., & Praneetha, C. (2021, June). Detecting phishing websites using an efficient feature-based machine learning framework. Revista Gestão Inovação e Tecnologia, 11(2), 2106–2112. doi: 10.47059/revistageintec.v11i2.1832

Tang, L., & Mahmoud, Q. H. (2021, August). A survey of machine learning-based solutions for phishing website detection. Machine Learning and Knowledge Extraction, 3(3), 672–694. doi: 10.3390/make3030034

Wang, B. (2025, July). Malicious URL detection with explainable machine learning techniques. Proceedings of the 2025 2nd International Conference on Informatics, Education, Computing, and Technology Applications (IECA 2025), 293–299. doi: 10.1145/3732801.3732854

Zaland, Z., Bazai, S. U., Marjan, S., & Ashraf, M. (2021). Three-tier password security algorithm for online databases. 2nd International Informatics and Software Engineering Conference (IISEC 2021). doi: 10.1109/IISEC54230.2021.9672434

Zouina, M., & Outtaj, B. (2017, December). A novel lightweight URL phishing detection system using SVM and similarity index. Human-centric Computing and Information Sciences, 7(1), 1–13. doi: 10.1186/s13673-017-0098-1