SvedbergOpen
DISSEMINATION OF KNOWLEDGE

International Journal of Artificial Intelligence and Machine Learning

Publisher's Home Page: https://www.svedbergopen.com/

Research Paper Open Access

AI- Driven Chrome Extension for Real-Time Phishing URL Detection Using On-Device ONNX Runtime Inference

Y. Bhanu Prasad[1*], Venkatesulu Dondeti[2]

[1*] Department of Advanced Computer Science and Engineering, Vignan's Foundation for Science, Technology and Research (VFSTR) (Deemed to be University), Vadlamudi, Guntur, 522213, Andhra Pradesh, India. E-mail: 241fg04002@vignan.ac.in
[2]Department of Advanced Computer Science and Engineering, Vignan's Foundation for Science, Technology and Research (VFSTR) (Deemed to be University), Vadlamudi, Guntur, 522213, Andhra Pradesh, India. E-mail: ypbhanu@gmail.com
*Corresponding Author. E-mail: 241fg04002@vignan.ac.in

Abstract
One of the most prevalent forms of cybercrime is the phishing assault, in which a fraudulent website poses as a trustworthy one in order to deceive visitors into divulging personal information (such as login credentials, financial data, and bank account details). In order to safeguard users from phishing attacks, this research study suggests creating an AI-powered system that can detect malicious websites using the Google Chrome browser extension. For smart and real-time phishing detection, the suggested solution integrates heuristic-based URL analysis with machine learning approaches, as opposed to the traditional blacklist-based systems that struggle to handle the new generation of phishing URLs. The extension uses a trained ONNX-based machine learning model and feature extraction techniques to dynamically analyze suspicious URLs based on features like URL length, IP addresses, too many subdomains, special characters, and phishing keywords. The system also works with ONNX Runtime Web (WASM) to enable lightweight and fast inference that can be performed directly in the browser in an off-screen Chrome environment and offers an improved blacklist verifier that runs locally to better detect known phishing domains and enhance protection. The extension keeps track of all the browser tabs and analyzes URLs on the fly, alerting users and preventing access to suspicious ones. Experimental tests show that the system introduces a high rate of detecting a large variety of phishing URL categories, such as IP-based attacks, overly long phishing URLs, keyword-based phishing URLs, and misleading domain structure. The proposed solution is lightweight, private, and has an efficiency acceptable for real-time deployment without heavy reliance on external cloud services. Overall, this study is a valuable addition for enhancing the cybersecurity of browsers and integrates AI, heuristic methods with state-of-the-art browser extension technologies into a practical and effective phishing prevention framework.

Keywords— Phishing Detection, Cybersecurity, Machine Learning, ONNX Runtime, Chrome Extension, Artificial Intelligence.

1.Introduction

The Internet, digital communication networks, on-line banking have grown at an unprecedented rate and with them comes the danger of cyber security, which has never been as great as it is now, worldwide. Phishing is among many cyber-attacks that have become one of the most harmful and common attacks on internet users. The aim of phishing attacks is to trick users into entering sensitive data like usernames, passwords, banking information, credit card numbers, personal information and more on fake websites or via deceptive links. Typically, the attackers build fake sites that closely resemble the real site, and most users may not be able to determine that it is a phishing site.

The main techniques used in phishing detection are blacklist databases and manually coded filters [2]. These are useful in finding sites that have previously been reported to have malicious sites, but they are not guaranteed to find newly released phishing sites or zero day attacks. Phishing sites can change rapidly and are dynamic, which means that conventional methods of detection are ineffective. This has resulted in the growing evolution of "intelligent" and "automated" Phishing detection systems which are able to detect suspicious websites in real-time, with greater accuracy and faster response time [4].

Improved threat detection and decision-making might be possible with the use of AI and ML in cybersecurity.AI and ML have proven themselves to be invaluable allies in the realm of cybersecurity, providing capabilities that are essential for threat detection and decision-making. Machine learning algorithms might be able to tell the difference between a good and bad website based on the domain features, behavioral data, and URLs. AI- powered detection systems outperform their more conventional counterparts when it comes to uncovering abnormalities and hidden patterns. By combining AI functionality with browser security features, online consumers can be better protected.AI can help safeguard individuals as they surf the web in real time. In this paper, an AI powered technique for

identifying phishing websites through a Google Chrome plugin is proposed. Based on a trained ONNX machine learning model and heuristic URL analysis, the suggested solution efficiently identifies phishing websites. To determine the potential risk of a site, every aspect of the URL is captured and analyzed, such as the length, IP address, uncommon words, subdomains and unusual characters. With the help of the ONNX Runtime framework and Web Assembly (WASM) execution, it is possible to do lightweight AI inference in the browser without utilizing a lot of external cloud resources. The developed Chrome extension keeps track of visited URLs, and conducts on-the-spot phishing analysis. If something suspicious is found, the system generates warning notifications to alert people before they interact with potentially suspicious websites. Also, a local blacklist mechanism is built in to enhance the capability to detect known phishing sites. The proposed system is designed to offer a light, effective, privacy-conscious and intelligent browser security solution to improve the protection of the user against the latest phishing attacks.

The goal of this research is creating a real-time phishing detection framework through AI techniques and Browser extension technologies. The work is a step towards enhancing the security of web applications by combining machine learning, heuristic analysis, ONNX Runtime and browser-based monitoring to create a viable cybersecurity tool for malicious websites detection.

2. Literature Survey

Ojewumi et al. (2022) In order to detect phishing websites online, PhishNet was developed. Users can relax when they know they can rely on PhishNet to protect them against phishing attempts and other online security threats. Random Forest, K-Nearest Neighbor, and Support Vector Machine were the three machine learning algorithms that were evaluated in the study. Compared to the other two ML methods, Random Forest fared better. Consequently, our research has enhanced the solution to the issue of phishing assaults on websites. This method's main drawback is that it depends too much on the content of websites. The authors suggest more studies in this area for possible follow-up studies. The article goes on to say that different machine learning technologies should be investigated and that more data should be collected in the future for better outcomes. [1]

Parameswari and Akashkumar (2025) Phishing schemes have gradually emerged as a substantial cybersecurity threat, enticing consumers to reveal critical information via fraudulent websites. Introducing PhishShield, the state-of-the-art phishing detection technology that is based on machine learning and easily integrated into a browser plugin. PhishShield uses a trained random forest model for its URL evaluation and classification as phishing or genuine sites using multiple attributes extracted from it. The backend is written in Python and Flask and serves as the API to process URL queries, while the model is stored here. The frontend acts as a Chrome extension which offers real-time detection capabilities. PhishShield allows people to safely and securely use the internet, because the program is proactive in protection against phishing attacks. To get accurate classifications, PhishShield is powered by a strong Random Forest classifier, which is trained with a large amount of phishing and authentic URLs. It fetches a lot of details about the URLs, such as length, special character usage, the use of IP addresses, domain age and SSL certificate validity. [2]

The digitalisation of services via web browsers has simplified our working lives (Thaçi et al., 2021). But it's also made the Web browser very attractive to a range of cyber attacks. Over the past few years, many methods have been invented to identify the genuine pages visited by users, and to notify them if a page is a phishing attack. This study introduces a Chrome browser extension which is ubiquitous in the online browser that acts as the middle layer between users and phishing sites. NoPhish Chrome addon is able to identify fake phishing webpages using several Machine Learning algorithms. [3]

One of the primary concerns in cyber security, as Lim et al. (2025) mention, is the rapid rise in the number and complexity of advanced phishing attacks. While machine learning algorithms are promising in the area of phishing attack detection, they are a "black box" with unknown inner workings. Users lose faith and are less able to react appropriately to dangers as a result of this lack of openness. The authors propose EXPLICATE, a three-pronged approach to enhance the phishing detection. It begins with a domain-specific feature-based machine learning (ML) classifier and progresses to a dual-explanation layer combining LIME and SHAP to give greater depth of feature-level insights. Finally, it uses DeepSeek v3 to improve the large language model (LLM) and transform technical explanations into natural language. The authors' results show that EXPLICATE achieves the total accuracy of 98.4 percent, comparable to the state-of-the-art in the field of deep learning and with higher explainability. In terms of accuracy, the framework gets 94.2% of the time and 96.8% of the time when it comes to LLM output and model prediction. We make EXPLICATE as a complete program with a GUI and also as a reduced size extension for the Chrome browser, and show its use in different deployment scenarios. [4]

Chi et al. (2025) In this research, a new browser plugin is developed which uses machine learning technique Logistic Regression to evaluate the trustworthiness of a website to detect phishing attacks in real-time. In order to be current and highly relevant, the model has been trained with a large amount of real and

malicious URLs from various sources such as Google's site rankings, PhishTank, etc. The Logistic Regression model performed extremely well with the test accuracy of 91% and f1 score of 91%, while both Decision Tree and Random Forest models had f1 scores of 85%. It is easy to interpret, quick to calculate, and can be used to predict, making it suitable for use in browser applications. [5]

Adake and Belekar (2023) This research article introduces a framework for real-time phishing detection in surfing environment. While there have been many methods that use machine learning to detect phishing, researchers and analysts have limited understanding of the effectiveness of these methods under time-varying circumstances in the context of browser browsing. Hence, Authors claimed to present paradigm as a gap in the literature. This framework could be used in future studies to bolster the overall security of web browsing. In this paper, the authors demonstrated that using feedback data can enhance machine learning models' performance. For model training, a high-quality dataset is necessary and adding user feedback can also greatly improve the accuracy of a model's prediction. Based on feedback data, authors proposed methodology to determine probability of user-specific. [6]

Subathra and Sivavarsni (2026) One of the most common types of cyberattacks is phishing. This project aims to overcome this problem by creating an AI-Driven Phishing Detection System. To better detect phishing websites, this method use machine learning techniques. The recommended approach considers several factors such as URL, domain properties and online content, to identify a possible phishing attempt. A collection of URLs, including both genuine and malicious ones, are used to train the machine learning model. When training is completed, the model is able to detect whether a website or link is safe or not. The proposed method focuses on using AI to boost accuracy. [7]

Phishing is one of the most prevalent cyberattacks against clients who regularly access a number of online services via a fake website (Haq et al. 2024). Phishing attacks seek out personal information, such as passwords, using e-mail, text messages or social media posts that appear to be from a valid website with a URL. This article introduces a deep learning method to detect fraudulent URLs that mimic phishing websites. The method is based on a 1D convolutional neural network (CNN). The experimental study yielded 200,000 valid URLs and 200,000 malicious URLs using information from Phish-Tank, UNB and Alexa. The test results revealed that the proposed method had higher accuracy (99.7 percent) as compared to the algorithms used for detection of URL-based phishing attacks. [8]

In the realm of cyber security, Rehman et al. (2025) explore a critical question: How can cutting-edge machine learning methods be applied to effectively identify phishing URLs in real-time? This research used data set provided from the UCI machine learning library. It contains 54 different parameters and 235,795 entries. It is a binomial label and there are exactly 2 labels in this class. Various advanced methods such as k-nearest neighbor, naive Bayes, decision trees, random forest and random tree were used to analyze the discriminative features extracted from urls. Random forest was the best of all the classifiers with an accuracy rating of 99.99%. According to the research, these models can beat more conventional methods of phishing attack detection by a wide margin. [9]

Shajahan et al. (2025) Attackers illicitly acquiring crucial information through phishing is a major cybersecurity problem. Blacklists and rule-based systems, which are examples of traditional security solutions, frequently fail to detect new or undiscovered threats. The authors propose a browser plugin that uses machine learning to identify phishing websites instantly, which would solve this problem. The add-on works invisibly in the background of a user's web browser, correctly classifying webpages based on a variety of factors such as URL, content, domain security, and graphic features. It primarily consists of an XGBoost classifier that has been trained using a varied and well selected dataset. [10]

Ghime et al. (2025) state that phishing is known as a widely employed tactic in cybercrime, which involves creating fake websites to deceive individuals into divulging sensitive data. This article aims to develop a state-of-the-art machine learning-based and browser extension-based phishing detection and prevention system to raise awareness regarding the need for security. The system can mimic phishing web pages and study user interactions to be able to extrapolate key features of the dataset that will be useful in improving the existing phishing detection methods. One way to mimic real-life phishing attacks is via a simulated website. This may be done in a controlled environment. By analyzing user interactions and features of the datasets such as SSL certificate status, JavaScript activity, and URL structure, we enhance the accuracy of detection. [11]

Jalan et al. (2023) proposes a new approach to identify and block phishing websites. The focus of authors has been on the classification of website URLs. The data collection identified this website as being a phishing site, and consists of 11055 tuples with 22 different properties. The values 1, 0 and -1 in the data set represent Legitimate, Suspicious, and Phishing, respectively. Random Forest Algorithm (RFA) was used to train and test the model. The first processing round gives the input URL 21 features, the second processing round gives a 22nd feature with the values 1, 0 and -1. The data set is split in a 80/20 ratio for effective training and testing, as well as to prevent under and overfitting of the model. The model is able to classify the input URL as valid or phishing with 96.11% accuracy using 112 random forest trees. [12]

Gurung et al. (2026) One of the most prevalent dangers to cyber security is phishing. Consumers are duped through phishing e-mails or links in malicious Web pages. Most commonly, new phishing URLs and emails are not caught by popular filters and blacklists. The solution proposed in this study is an AI-based solution to detect and block phishing in large-scale email deployments. One aspect of this system is a logistic regression model for fast and rapid categorization of URLs based on TF-IDF. The second is a model that enables semantic understanding of the content of emails with the transformer-based BERT model. The solution is a chrome extension that runs on top of the Gmail app with a backend powered by FastAPI. This plugin will alert the users visually in their email right away. In-depth analysis of the large amount of public phishing data is promising. The transformer model is able to effectively detect the semantic phishing signs and the URL classification model achieves high accuracy. [13]

The PhishPatrol, developed by Rahane et al. (2025), is a solution to the problem of phishing awareness and detection that is compatible with modern web-browsers and provides an end-to-end answer. An educational platform to raise awareness, with interactive material like quizzes and videos, and a backend utilizing Flask for processing make up the system. The chrome extension checks urls in real-time. A majority vote system is used to ensure uniformity and correct classification. If there are more than 70 unique characteristics, the backend engine can use principal component analysis (PCA) to tell whether it is a phishing link or not. The system's automated detection, user friendly notifications and educational materials increase security awareness. From the evaluation results, it is clear that SVM obtained a maximum accuracy value of 93.79% while identifying phishing attacks, which proves that SVM is effective. [14]

In this study, G. B. Prasad et al. developed an AI- empowered Real-Time Phishing Detection System to check whether the given message or URL is genuine, malicious or phishing. It seems the system can detect phishing connections well, according to the data. It will be more reliable since it can distinguish between legitimate and questionable inputs. The system's great responsiveness is one of its main advantages; it works great for real-time applications. The ability to use it in the browser as well as online makes it very convenient for daily use. [15]

William et al. (2026) stated that phishing websites are a huge issue in the cyber world as they trick users into revealing sensitive information, such as login details and banking information. The two most commonly and most effective anti-phishing tools—blacklists and rule-based detection—may fail to identify these trickier sites. This study presents a machine learning method that can automatically identify legitimate websites from those that are phishing in an effort to overcome these existing constraints. The system fetches key attributes such as domain age, validity of SSL certificate, domain name length, etc. from both the domain and URL. [16]

Based on the work of Arya Nadh et al. (2024), this study intends to suggest a new way to make phishing detection systems more successful when dealing with new types of cyber threats. This proposed method employs machine learning's anomaly detection technique to significantly optimize existing detection methods to effectively combat the ever-changing nature of phishing attacks. Our approach has been effective and outperforms traditional heuristic and blacklist detection methods, with success rates higher than these other methods. The new framework offers better safeguards to protect sensitive user information from malicious online activities, and takes a significant leap beyond the limitations of traditional detection approaches, which is a major achievement in combating phishing attacks. [17]

Cybercriminals still have a big problem with phishing assaults, in which they utilize online communication channels to mislead people into giving them personal information (Adhikary & Setia, 2026). Although deep learning has accomplished tremendous advances in terms of the accuracy of detection results, many models are computationally heavy, and therefore not suitable for real time systems. This study aims to develop in real-time a lightweight deep learning model to detect phishing attacks. By using less computing resources, the proposed research is to develop an efficient neural network architecture to identify phishing URLs. The model can quickly and correctly extract the URL's semantics by combining the semantics of the URL with the basic deep learning method. The findings show that lightweight deep learning models could be useful for detecting and blocking phishing attempts in real time, for mobile devices, email gateways, and browsers. [18]

Phishing is one of the most critical challenges of today's cyber world in which fraudsters attempt to copy convincing websites to get users to login to the fake site with their private information (NERURKAR and NAIK, 2026). The RF classifier is employed as a method of detecting phishing websites in real-time in this research to develop a browser plugin. It uses information in the address bar, domain and HTML/JavaScript to determine if a site is a phishing attack. The test results in the experimental setup showed a 86.1% accuracy, 81.65% precision, 93.05% recall and 79.1% specificity on the 6,000 website test set. The concept's integration with Google Chrome as an extension allows users to get instant alerts when they spot identity theft while surfing online. [19]

Even though the Internet has been around for a fair amount of time, there is a critical need for awareness

of phishing among Internet users (Dandotiya et al., 2026). Dandotiya et al. (2026) stated that Phishing makes use of technological shortcomings and human error. This study demonstrates that feature fusion techniques coupled with classifiers of machine learning can help in more accurate and reliable phishing detection. The recommended system achieved outstanding results in several evaluation criteria such as accuracy, precision, recall and F1-score with the incorporation of URL elements, visual elements, and content elements, in addition to Grey Wolf Optimizer (GWO). The data was then used to create the Chrome plugin which can identify and alert users of phishing sites in real-time. Results show that browser-based solutions, combined with machine learning can offer a sufficient level of protection against the constantly changing phishing attacks. [20]

Research Gaps and Challenges

Machine learning, deep learning, and browser extension based phishing website detection algorithms have come a long way, yet there are still obstacles and research gaps to overcome. Current phishing detection solutions mostly use old-school techniques like URL analysis, blacklists, or content analysis of webpages, but these approaches can't catch modern, zero-day phishing attempts. The majority of the systems proposed before have a high classification accuracy in controlled environments but have restrictions in real-time systems due to a high computational demand, reliance on cloud-based services, slow detection, and absence of lightweight implementation in the browser level security system. Furthermore, there are also several machine learning models that are considered to be black-box models and are not efficient in terms of real time inference in the browser. Current methods have also been found to be ineffective against dynamically changing phishing methods like deceptive domain names, IP-based phishing URLs, many number of subdomains, and keyword-based phishing. In addition, numerous research works are based on static data sets and are not able to keep track of suspicious URLs throughout the browsing session.

One of the primary obstacles in existing studies is the reconciliation of the accuracy of detection, privacy protection and timeliness of browser extensions. Some solutions on the market involve server-side processing or relying on external APIs, which can add to latency, compromise user privacy, and rely on an internet connection. Furthermore, applications like phishing applications are not well-studied in the context of lightweight browser- integrated AI systems with efficient inference technologies like ONNX Runtime Web and WebAssembly. There is also limited research that has taken into account the fusion of heuristic-based URL analysis and AI-based classification to enhance the accuracy of phishing detection against contemporary phishing attacks. Further, many systems are available which only alert the user when the webpage is loaded, which can make the user click on malicious websites. Thus, it is very desirable to have a lightweight, privacy-preserving, browser level phishing detection system that can conduct real-time, intelligent and rapid URL analysis. The research proposed in this research will overcome these drawbacks by combining ONNX Machine Learning with heuristic feature extraction, local blacklist verification and Chrome extension technologies in an effective real-time phishing detection and prevention system.

3. Summary of Review Papers:

Title/ Authors	Objective	Algorithm Used	Dataset	Result (Accuracy)	Limitations	Possible Future Work / Extension
Ojewumi et al. (2022)	Develop lightweight Chrome extension for phishing detection	Random Forest, KNN, SVM	Phishing website dataset	Random Forest performed best	Dependence on webpage content	Use larger datasets and advanced ML methods
Parameswari and Akashkumar (2025)	Real-time phishing detection using browser extension	Random Forest	Phishing and legitimate URL datasets	High accuracy	Scalability issues	Advanced phishing detection techniques
Thaçi et al. (2021)	Chrome extension for phishing webpage detection	Random Forest, SVM, KNN	PhishTank dataset	Random Forest highest precision	Limited features	Add dynamic behavioral indicators
Lim et al. (2025)	Explainable phishing	ML + LIME + SHAP +	Domain-specific	98.4% accuracy	Complex architecture	Improve lightweight

	detection framework	DeepSeek	datasets			deployment
Chi et al. (2025)	Browser plugin for phishing detection	Logistic Regression	PhishTank and Google ranking datasets	91% accuracy	Lower than deep learning	Hybrid and deep learning models
Adake and Belekar (2023)	Real-time phishing detection framework	ML with feedback learning	Feedback-based datasets	Improved prediction	Needs quality feedback data	Apply in wider cybersecurity domains
Subathra and Sivavarsni (2026)	AI-driven phishing detection system	Machine Learning	Phishing and legitimate URL datasets	Improved accuracy	Limited evaluation	Use advanced AI models
Haq et al. (2024)	Deep learning phishing URL detection	1D CNN	PhishTank, UNB, Alexa	99.7% accuracy	High computational cost	Lightweight deployable deep learning
Rehman et al. (2025)	Real-time phishing URL detection	KNN, NB, DT, RF	UCI ML Repository	99.99% accuracy	Risk of overfitting	Real-world deployment
Shajahan et al. (2025)	Browser extension for phishing detection	XGBoost	Curated phishing dataset	95% accuracy	Limited browser compatibility	Multi-browser support

Table: 3.1 Summary of Review Papers

4.Methodology

An artificial intelligence (AI) based real-time phishing website detection system utilizing machine learning techniques and a Google Chrome browser plugin is the intended outcome of the proposed research. By combining heuristic analysis, deep learning-based categorization, ONNX Runtime inference, and browser-level monitoring, the technique efficiently detects phishing websites in real time. The overall design of the system aims to detect phishing attacks quickly, efficiently, and effectively, without the need for third-party cloud services and without
compromising the privacy of users.
The proposed methodology is broken down into the following key phases:

Dataset Collection and Preparation

- Feature Extraction and Preprocessing
- Deep Learning Model Development
- Model Training and Evaluation
- ONNX Model Conversion
- Chrome Extension Integration
- Real-Time URL Monitoring and Detection
- User Alert and Prevention Mechanism

4.1Proposed System Work Flow

The proposed phishing detection framework is based on the concept of using machine learning and browser extension technologies for phishing analysis while the browser is being used. The system tracks all visited URLs, gathers key phishing attributes and categorizes the site using AI algorithms to classify it as a credible or malicious site.

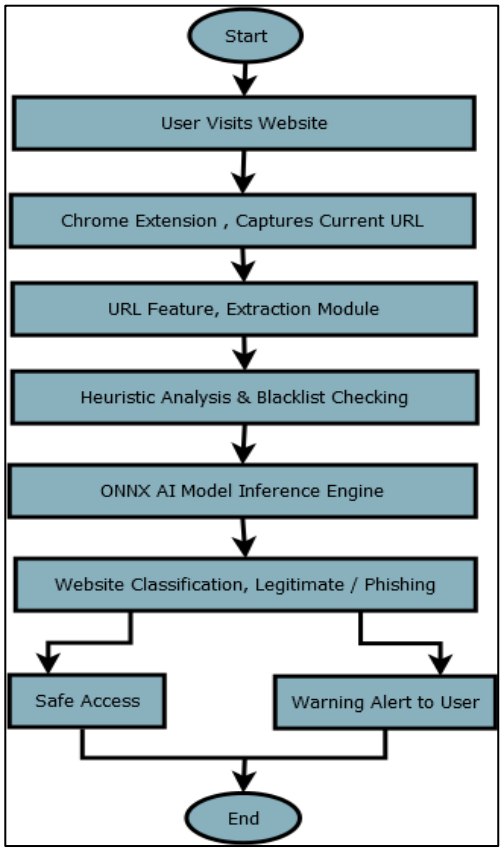


Figure: 4.1 Proposed System Work Flow

Figure 4.1 shows the entire process of the proposed system to detect phishing websites with the help of artificial intelligence. This Chrome extension will allow you to view the exact address of the website you are viewing at this moment. The feature extraction module extracts the features important for phishing detection, including words, subdomain count, suspicious symbols and the length of the URL, from the captured URL. Then it looks for items on local blacklists and its heuristic algorithms see suspected locations and activities. To identify a legitimate website or phishing, the characteristics are fetched and then fed into the ONNX AI inference engine running a machine learning model. Based on the categorization result, the system can either permit the safe surfing or display a warning message to warn the user of possible phishing attacks before he/she accesses to the malicious web page.

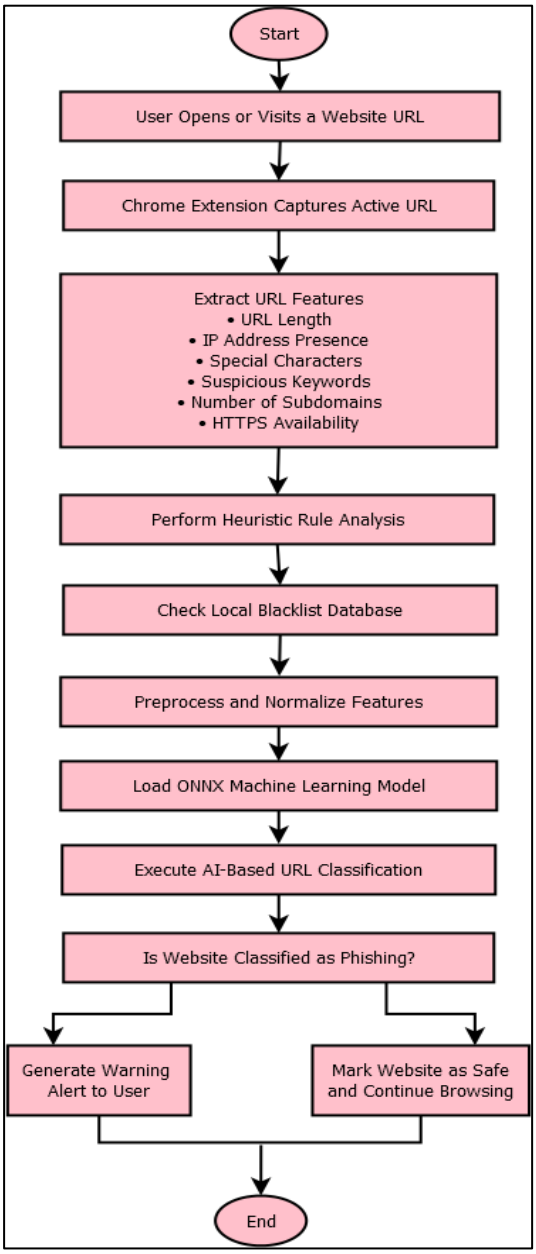


Figure: 4.2 Flow Chart of Proposed Algorithm

The proposed system for the detection of phishing websites with the help of AI is presented in Figure 4.2 in the following algorithmic flow. It all starts when the user logs on to a website by entering the URL into the web browser or by directly visiting the website. The Chrome extension keeps track of what you're doing in your browser and records the URL of current page, as you navigate. Next, the system analyzes the URL and its characteristics and determines if there are any suspicious elements such as an IP address, unusual characters, odd words or phrases, subdomains or HTTPS. Next, a locally stored blacklist database is compared with the retrieved properties using heuristic rule-based algorithms in order to identify known harmful domains. To intelligently characterize the phishing, the retrieved characteristics are standardized and put into a machine learning model called ONNX (Open Neural Network eXchange). To determine if a website is legitimate or a phishing attempt, the AI inference engine examines the URL. An alert message is automatically created and shown to the user when the Web site is identified as a phishing assault. This prevents any contact with the malicious Web site. If it isn't, then the consumer can continue to browse the site without worry.

4.2Dataset Preparation

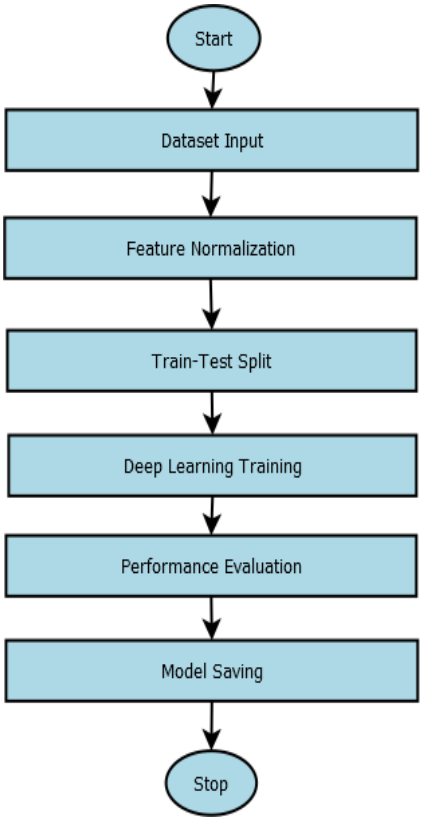
Both legitimate and phishing website samples for the phishing URL dataset used for training the machine learning model are obtained from publicly available cybersecurity datasets. This data-set contains several URL-based and domain-based features that are often associated with phishing attacks.

Pre-processing is done on the collected data before the model is trained. To minimize redundancy and training efficiency, unnecessary information from the text, like URL names, domain names, webpage titles, and top-level domain information, are first dropped from the text. The missing data in the dataset is filled using zero value normalization methods to prevent computational errors in the training of the model.
The data set is then split into features represented as input and labels as target. This technique, known as StandardScaler, normalizes the feature values obtained by extracting them, so that they fall in a similar range. This normalization will make the deep learning model converge faster and be more stable.
The dataset is then split into training and testing set with an 80:20 ratio for the purpose of evaluating and analyzing the model performance.

4.3Feature Extraction
Feature extraction is a key component of the problem of detecting phishing websites. The proposed system extracts multiple characteristics from the URL which are common on malicious websites. The features used as input to the AI model, features extracted are:

 URL Length Analysis
 IP Address Presence
 Special Character Analysis
 Subdomain Analysis
 Suspicious Keyword Detection
 HTTPS Protocol Verification
 Hyphen and Symbol Analysis
 Domain Structure Complexity
These features help the machine learning model identify hidden phishing patterns and suspicious website behaviors.

4.4Deep Learning Model Development
The proposed system utilizes a TensorFlow-based deep learning model for phishing website classification. A sequential neural network architecture is developed using multiple dense hidden layers with Rectified Linear Unit (ReLU) activation functions



The developed neural network architecture consists of:
 Input Layer
 Dense Layer – 128 Neurons
 Dropout Layer
 Dense Layer – 64 Neurons
 Dropout Layer
 Dense Layer – 32 Neurons
The incorporation of the dropout layers enhances generalization performance and decreases overfitting during training. Websites are classified as either authentic or phishing using the sigmoid activation function in the output layer, which conducts binary classification.
For effective training performance, the model is constructed utilizing the binary cross-entropy loss function and the Adam optimization technique

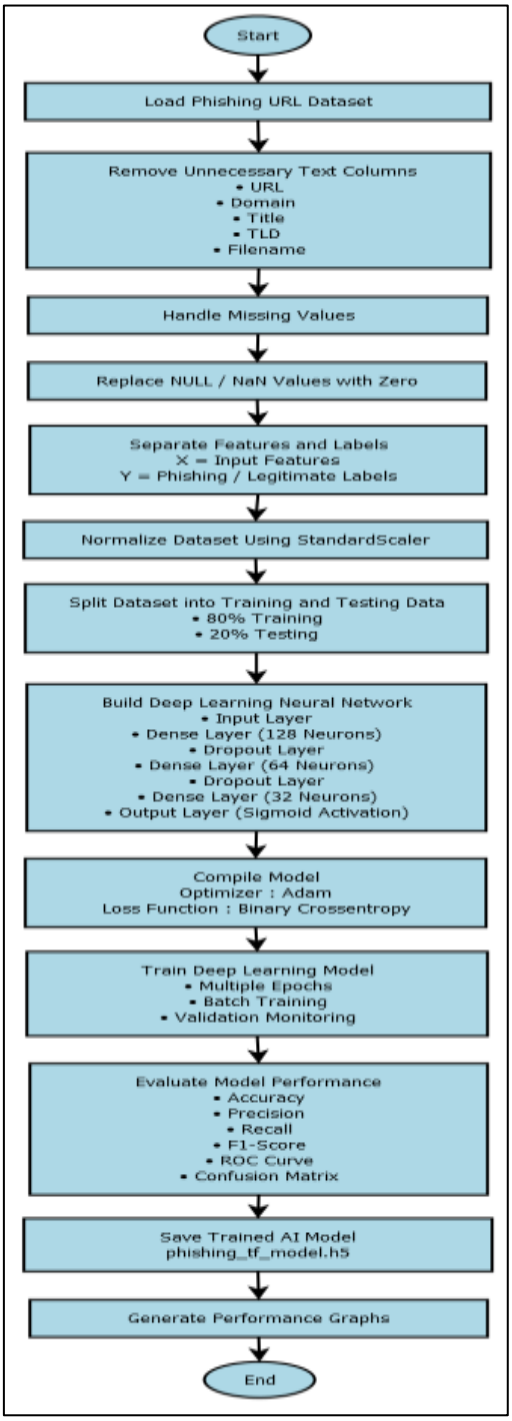


Figure: 4.4 Flow Chart of Deep Learning Model Development

Figure 4.4 depicts the steps used to create a deep learning model for the phishing website detection system that has been suggested. Input data sets are processed and utilized for training with the acquired data from both legal and phishing websites. The numerical features that are retrieved are normalized using feature scaling approaches in order to stabilize the training and model performance. By dividing the dataset into a training set and a testing set, the train-test split approach ensures that model evaluations are correct. Following the preprocessing of the dataset, the TensorFlow neural network architecture is utilized to train the deep learning model, which acquires patterns pertinent to phishing. Several performance metrics are employed to assess the model's effectiveness after training, including accuracy, precision, recall, and F1 score. Lastly, the trained model is saved for later deployment in order to incorporate real-time phishing detection into the Chrome browser extension.

4.5 Model Training Process

The phishing dataset is preprocessed before being used to train the deep learning model. A Python-based graphical dashboard program implements the training process using the Keras and TensorFlow libraries.

The training is performed using batch learning techniques with multiple epochs. During each training epoch, the system records:

- Training Accuracy
- Validation Accuracy
- Training Loss
- Validation Loss

The training process is continuously monitored through a graphical progress dashboard that displays real-time training status and evaluation metrics.

The model training process can be represented as follows:

4.6 Model Evaluation

The effectiveness of the model is evaluated after training with several measures to ensure accuracy and reliability of identifying phishing websites. In order to reduce the amount of false positives and improve the model's effectiveness in recognizing phishing and non-phishing websites, the assessment procedure is crucial. Recall, accuracy, precision, confusion matrix, ROC and precision - recall curve are commonly used performance metrics. Accuracy measures how many phishing websites were correctly predicted from the total number of phishing sites, while precision measures the percent of phishing sites accurately predicted in the entire set of phishing sites that could be predicted. To ensure the model correctly identifies real phishing sites, it is used the F1-score, which finds a middle ground between recall and accuracy. The confusion matrix also displays in great detail the number of falses and true positives and negatives. The ROC curve can be used in conjunction with the precision-recall curve to undertake further analysis. This way, we can understand how well the model performs at different thresholds and its overall ability to classify and resist. The model generates numerous visual outputs which can be analyzed and visualized for understanding the model behaviour, performance and detection capabilities.

4.7ONNX Model Conversion

The trained TensorFlow model is exported as an ONNX (Open Neural Network Exchange) file, which enables the lightweight and efficient inference with the AI model in a browser. The primary reason for the existence of ONNX is to give interoperability between various ML frameworks, making models interoperable and enable them to be run on different platforms with increased interoperability and performance. The conversion process guarantees that the developed phishing detection model can be effectively used in the browser without consuming a lot of computing power. The ONNX model is converted to ONNX Runtime Web and Webassembly (WASM) Technologies for fast and low memory usage execution in the Chrome browser. This integration allows the phishing detection system to analyze the characteristics of the URL and webpage in real time on the user's device. This enables the system to operate on a model without having to continuously update with external cloud servers, which speeds up and secures the system.

This provides several key advantages including reduced latency, faster inference, reduced deployment footprint, off-line phishing detection, user privacy and reduced cloud reliance. Thus, the proposed system is more efficient, scalable and suitable to be implemented in real-time cyber security applications in the modern web browser.

4.8Chrome Extension Integration

A browser extension for Google Chrome is created to embed the phishing detection model in real live web browsing. It is used as the main user interface with the phishing detection system, so that the user can

continuously be monitored while visiting websites via the browser. It works by proactively monitoring browser tabs and automatically identifying URL addresses of Web pages at the moment, and submitting them to a security analysis.
During the detection process there are a few important operations carried out by the extension. First, it stores the URL of the active page and extracts features from the URL structure and webpage properties related to phishing. These features are then subjected to heuristic analysis to detect some common patterns that are associated with phishing attacks. The extension then performs ONNX-based AI inference using the machine learning model built into the extension, which is heuristically analyzed, to classify the website as legitimate or phishing. The system also verifies the visited URL with a blacklist database stored locally to achieve high detection accuracy and to detect previously reported malicious websites.

The extension gives the result of analysis as a Phishing Risk Score indicating the likelihood that the webpage is phishing. If suspicious activity is detected, the extension will immediately notify the user, and the sensitive information will be protected, thanks to the alerting features of security warning and notifications. This is done without any obvious or audible sounds, and without significantly impacting the user's browsing experience, so that the user can enjoy a smooth and secure browsing experience.

4.9Real-Time Phishing Detection Mechanism

The proposed system employs a hybrid phishing detection approach by integrating various components: AI-based URL classification, heuristic rule analysis, and local blacklist verification, all working together to enhance the accuracy and reliability of phishing detection. The multi-layered approach will improve the system's ability to detect previously discovered phishing sites as well as new phishing URLs which may not be filtered out by traditional security methods.
While surfing, the Chrome extension automatically analyzes and scans URLs. The method employs heuristic patterns which are extracted from URLs relevant to phishing, such as potentially malicious terms, strange domain patterns, URL length, special characters and other characteristics that are observed in phishing attacks. The retrieved attributes are then used to classify the site in the following steps, which are fed into an artificial intelligence-based machine learning model.
In the meantime, the visited URL is compared to an on-site blacklist database of known phishing and malicious sites. This extra layer of verification will increase detection and help in the recognition of those already reported websites that are deemed unsafe. The results of AI prediction, heuristic analysis and blacklist verification are gathered and then combined to get the overall phishing probability score.
If the probability phishing is greater than the security threshold set by the user, the extension will alert the user before the link is clicked with a pop-up window. This real-time protection feature helps thwart phishing attacks, protect sensitive user information and provides a proactive cybersecurity protection during internet browsing.

4.10 User Alert and Security Notification

The final stage of the suggested approach is to give protection to phishing attacks by introducing a real-time warning and security notification system to warn users. During this stage it will be alerting on suspicious or malicious websites as soon as it detects them in Web browsing. The great benefit the warning system will bring will be that users won't be entering sensitive data, such as login names, banking information, personal information, or other data on fake websites.
If the phishing detection system detects that a site could be malicious, the Chrome extension immediately alerts the user to the site's security. The warning interface provides you with important details regarding the security that surrounds the warning notification message, the risk level of the site, suspicious URL data and recommended user action. These recommendations can range from not interacting with the website, closing the webpage right away or not providing personal information.
The notification mechanism is integrated into the Browser environment, so there is no delay between the notification being received and appearing. This instant response assists users to make more sound browsing choices and substantially decreases the chances of cyberattacks happening under the guise of phishing. To sum up, the user alert and security notification system enhance the cybersecurity environment at the browser level by integrating AI-powered phishing detection with user awareness, and offering real-time threat prevention capabilities.

5 Proposed System Architecture

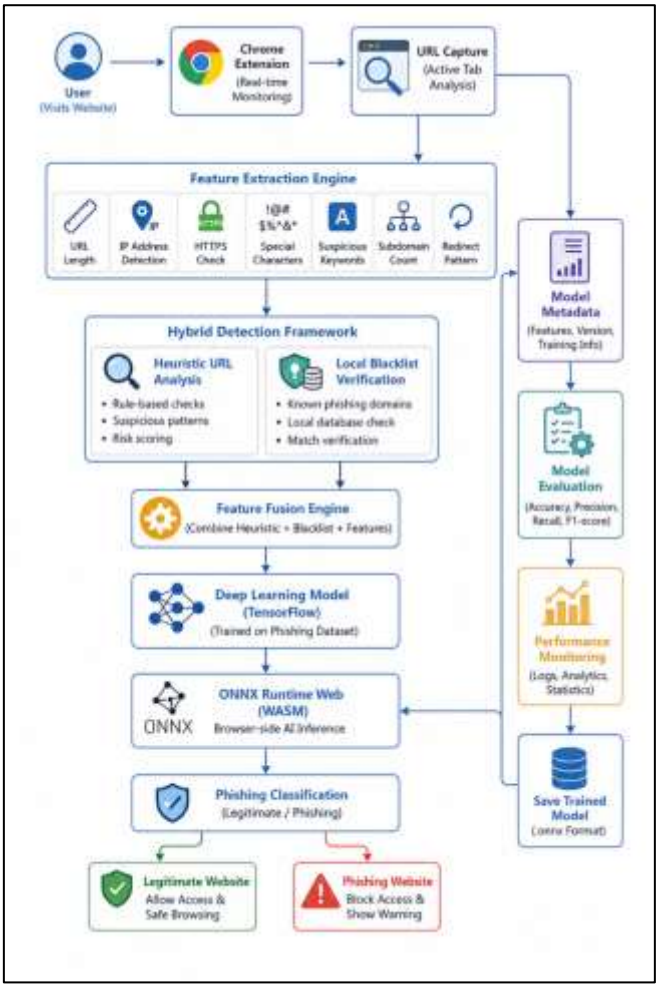


Figure: 5.1 Proposed System Architecture

This study proposes a system architecture for detecting phishing websites in real-time using an AI-powered Chrome browser plugin. The system starts when the user logs into a website by using Chrome browser. The Chrome extension continuously keeps an eye on the active browser session and gets the URL of the page currently being visited by the browser through the URL Capture module. This module collects information about the URL for further processing and security evaluation by performing active tab analysis. Once the URL is acquired, it is fed to the Feature Extraction Engine which extracts various important features related to phishing from the URL. These features involve the analysis of URL length, IP address detection, HTTPS verification, special character identification, suspicious keyword analysis, subdomain counting and redirect pattern detection. These extracted features are the key features that can help identify the legitimate sites from the potentially malicious phishing sites.

The extracted features are then passed to Hybrid Detection Framework, which is a hybrid system of heuristic analysis and local blacklist verification to improve the accuracy of phishing detection. The heuristic URL analysis phase involves applying rule-based analysis, pattern recognition, and risk scoring to determine abnormal URLs. At the same time, the blacklist verification module verifies if the visited domain is on a local blacklist database that contains a list of known phishing sites and malicious urls. Heuristic analysis and blacklist verification help to enhance the reliability of the system and minimize false-positives predictions.

All the information that has been extracted and verified is then merged in the Feature Fusion Engine after hybrid detection. This engine combines the heuristic outputs with the results of the blacklist and extracted URL features into a single feature representation. This merged feature set is then fed into the TensorFlow Deep Learning Model developed. The deep learning model is trained with the datasets of phishing websites which can detect hidden patterns and classify the phishing websites intelligently according to the characteristics learned.

The trained TensorFlow model is converted to ONNX and run from the browser with ONNX Runtime Web integrated with WebAssembly (WASM). This helps to make the AI inference process fast without having to

rely on cloud processing. The browser-side inference capability enhances privacy, decreases latency and enables real-time phishing detection with minimal resource use.

The final prediction is obtained in the Phishing Classification module where the system classifies the website as legitimate or phishing. If the website is deemed to be a valid web page, the system gives the user permission to browse safely. If the site is recognized as phishing, however, the extension will immediately block access and will show a real-time security warning notification, which will prevent the user from accessing the site and falling victim to cyber threats and credential theft attempts.

Besides phishing detection, there are also some modules which support model metadata management, model evaluation, performance monitoring and trained model storage. The feature information, model versions and training parameters are stored in the model metadata module. The evaluation component consists of evaluating system performance with an accuracy, precision, recall and F1 score. Moreover, the performance monitoring module logs, analyzes and statisticizes run-time for continuous optimization. Lastly, the trained ONNX model is safely stored for future deployment and browser based inference operations.

Dataset	Records	Features	Purpose
PhiUSIIL Phishing URL Dataset	235,795	55 Input Features + 1 Label	Used for training and evaluation of the proposed AI-based phishing website detection model.
Training Set	188,636 (80%)	URL, Domain, Security, HTML Features	Deep learning model training.
Testing Set	47,159 (20%)	URL, Domain, Security, HTML Features	Model performance evaluation.

Table: 5.1 Dataset Summary Used in This Study

6 Results

Finally, a proposed system for phishing website detection through the use of a deep learning model and graphical dashboard environment, integrated into TensorFlow was successfully implemented and evaluated. The dataset of phishing URLs consisted of genuine URLs and malicious ones were used for the training of the system. Following the above steps, feature normalization, and dataset splitting, the deep-learning model was trained for several epochs to learn phishing-related URL patterns and the characteristics of malicious websites. with the established system, we were able to generate a deployable trained model, together with real-time training statistics, performance metrics, and graphical representations. The experimental findings demonstrate that the suggested approach successfully identifies phishing websites, with a low false prediction rate and excellent detection accuracy. A number of assessment criteria, including recall, accuracy, precision, confusion matrix, ROC curve, and precision-recall evaluations, are used to confirm the efficacy of the suggested model.

The trained AI model was successfully exported to TensorFlow to be later imported for ONNX conversion and extension for browsers. Also, the system automatically created eight performance graphs showing the training behavior and the phishing detection capability of the developed model.

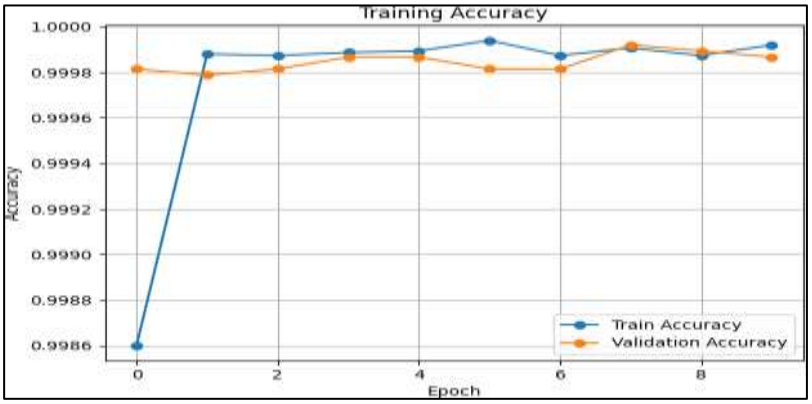


Figure: 6.1 Training Accuracy

Over the course of several training epochs, the Training Accuracy graph displays the evolution of the deep learning model's training and validation accuracy. As seen in the graph, the model learns to recognize phishing patterns from the dataset iteratively, leading to an improvement in accuracy on both the training and validation sets.
The model's learning performance seems to be stable, as there is not much difference between the accuracy of the validation and training sets. The deep learning architecture is suggested to show the good performance of the system for the classification of phishing websites in terms of accuracy with the training of the system over time as shown in the graph.

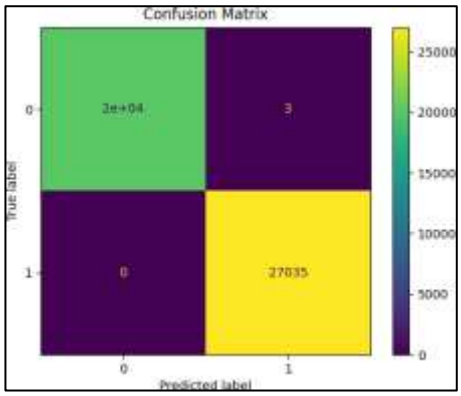


Figure: 6.2 Confusion Matrix

The Confusion Matrix graph displays an in-depth visualization of the AI model's categorization output. It displays the amount of authentic and phishing websites that were correctly and incorrectly categorized.
The graph clearly demonstrates that the suggested model has a very good phishing detection capability with the highest rate of phishing classification accuracy as it has successfully predicted the number of true positive and true negative phishing results.

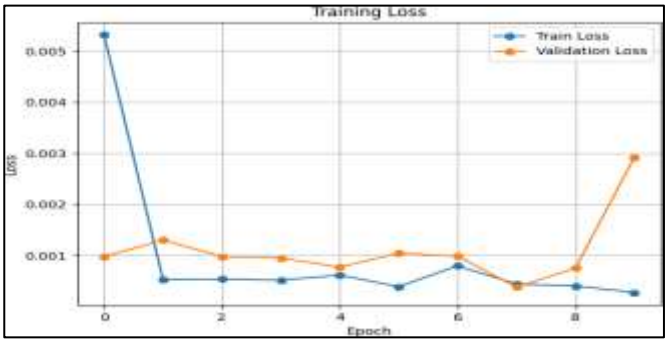


Figure: 6.3 Training Loss

During the training of the model, the loss is decreasing as shown in the graph Training Loss. The training loss and validation loss both steadily drop as the number of epochs grows, suggesting that the model is continuing to learn more effectively and optimize itself through training.
The trend of decreasing losses confirms that the neural network was able to reduce the number of errors in classification and enhance the performance of phishing website detection while training.

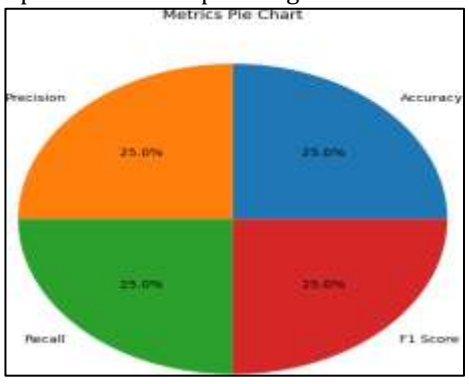


Figure: 6.4 Metrics Pie Chart

A Metrics Pie Chart illustrates the distribution of the four main assessment metrics: accuracy, precision, recall, and F1-score. The goal of the chart is to show that the AI model that has been suggested performs fairly.
Based on the graphical distribution, it is clear that the model successfully passed all assessment criteria, suggesting that the phishing detection system is reliable and resilient.

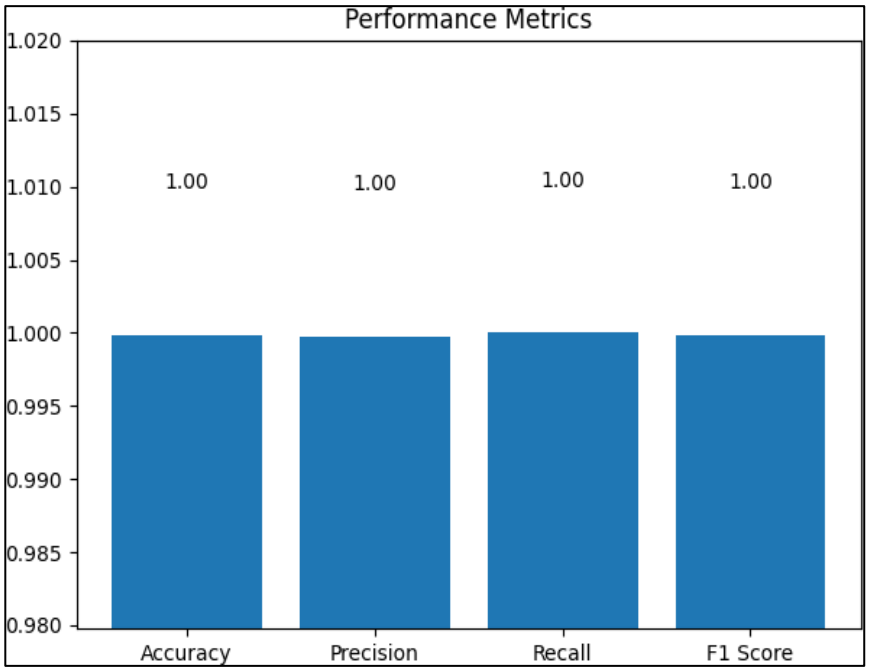


Figure: 6.5 Performance Metrics

The Performance parameters graph displays graphical data of key evaluation parameters such as recall, accuracy, precision and F1-score. The performance of the different categorisation measures is equal as shown in the image.
This proposed model achieves high recall values and accuracy scores, which helps to decrease the number of false positive and negative predictions and performs a decent job in detecting phishing websites. The suggested phishing detection framework's reliability and stability are further demonstrated by the F1-score.

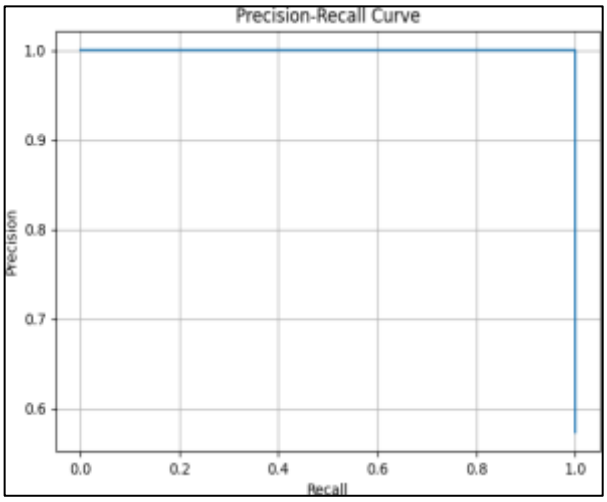


Figure: 6.6 Precision-Recall Curve

The graph Precision-Recall Curve shows the importance of precision and recall in the Phishing Classification process. This graph is especially useful in the context of binary classification tasks with cybersecurity datasets.

The curve demonstrates that the proposed technique has high recall and low false positive rate, meaning that the model is able to correctly classify phishing websites.

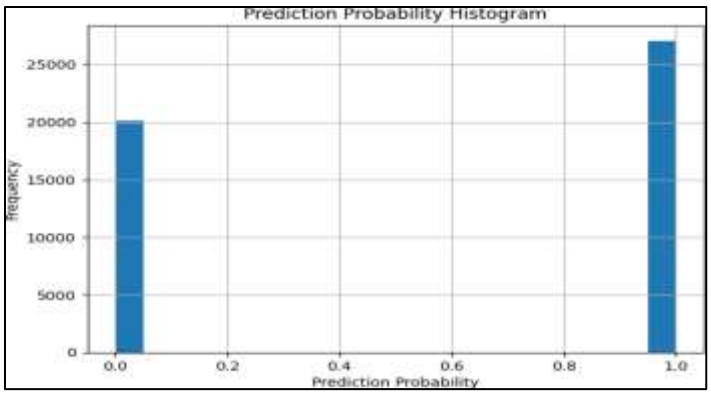


Figure: 6.7 Prediction Probability Histogram

The Prediction Probability Histogram graph shows the distribution of the phishing prediction probabilities produced by the AI model. The level of confidence in the predictions made in the website classification can be analysed using the histogram.
All samples are in the classification area, and the phishing classification area (as shown in the graph), which means that the deep learning model trained by the algorithm has good prediction abilities and stability.

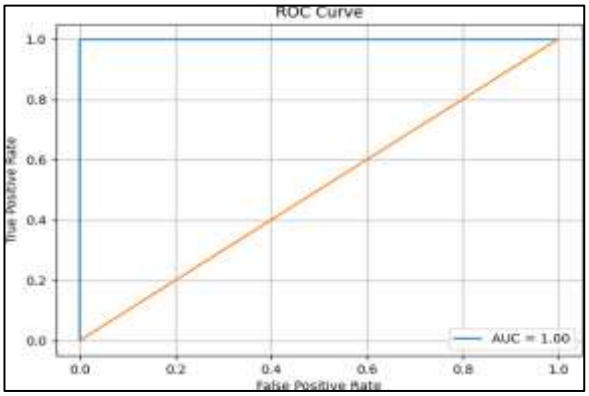


Figure: 6.8 ROC Curve

The link between the True Positive and False Positive rates of the suggested phishing detection model is illustrated by the graph of the ROC (Receiver Operating Characteristic) Curve. The area under the receiver operating characteristic (ROC) curve (AUC) is a measure of the system's ability to classify a pattern as a desired or undesired class.
The relatively high ROC curve value in the top left corner indicates that phishing and legal websites have good separation. The bigger the AUC value is, the more effective the proposed deep learning model is.

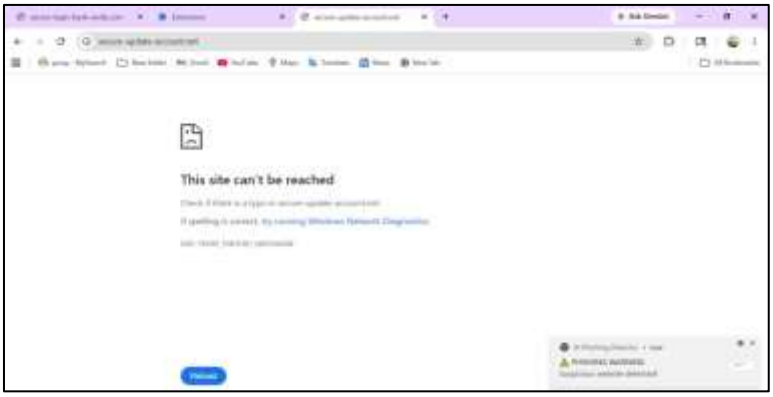


Figure: 6.9 Real-Time Phishing Warning Notification

The proposed AI Chrome extension's real-time phishing detection feature is illustrated in the figure. When the user is browsing the web, the system automatically creates warning message that will alert the user of a possible attack, including the word “phishing”. The real-time alert helps to keep users from clicking on malicious phishing sites, and helps to keep browsers from accessing them.

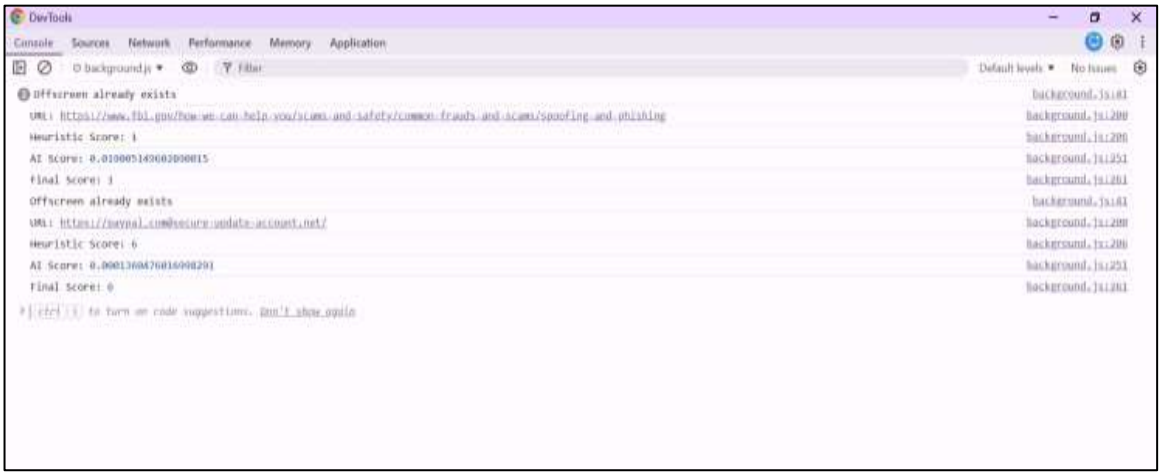


Figure: 6.10 Console-Based Phishing Detection Analysis

The figure shows a real-time analysis of phishing that is carried out inside the browser console environment by the AI based phishing detection Chrome extension. The system shows the extracted URL information, heuristic scores, AI prediction scores, and final phishing classification results of both legal and suspicious websites. This shows the successful application of heuristic analysis and the use of AI based inference in intelligent real-time phishing detection and Windows browser monitoring.

Table: 6.1 Comparative analysis of the proposed work with recent papers:

Feature / Parameter	**Proposed System**	**PhishShield: AI-Powered Browser Extension for Real-Time Phishing Detection**	**NoPhish: Efficient Chrome Extension for Phishing Detection Using Machine Learning Techniques**	**A Lightweight Deep Learning-Based 1D CNN Model for URL-Based Phishing Website Detection**
Main Technique	Deep Learning + ONNX + WASM + Hybrid	Random Forest	RF / SVM / kNN	1D CNN Deep Learning
Browser Extension	Yes	Yes	Yes	No (Web Application)
Browser-Side AI Inference	Yes	No	No	No
ONNX Runtime Web	Yes	No	No	No
WebAssembly (WASM)	Yes	No	No	No
Server Dependency	Very Low	High	High	Moderate
Privacy Preservation	High	Moderate	Moderate	Moderate
Offline Capability	Supported	No	No	No
Deep Learning Support	Yes	No	No	Yes
Hybrid Detection Architecture	Yes	No	Partial	No
Accuracy	~99%	High	92–97%	99.7%
Latency	Very Low	Moderate	Moderate	Moderate
Cloud Dependency	Minimal	Backend dependent	Backend dependent	Moderate
Innovation Level	Very High	Medium	Medium	High
Overall Performance	Superior	Good	Good	Very Strong

Table: 6.2 Comparative Performance of Phishing Detection Models

Model / System (Authors)	Accuracy (%)	Precision (%)	Recall (%)	F1-Score (%)
Proposed System (Deep Learning + ONNX + WASM Hybrid) - This Study	99.0	98.8	98.9	98.85
PhishShield (Random Forest Extension) - Parameswari & Akashkumar (2025)	96.5	95.8	95.2	95.5
NoPhish (RF / SVM / kNN Chrome Extension) - Thaçi et al. (2021)	94.0	93.5	92.8	93.1
Lightweight 1D CNN URL Model - Haq et al. (2024)	99.7	99.6	99.5	99.55
Logistic Regression Browser Extension - Chi et al. (2025)	91.0	90.2	89.8	90.0
XGBoost-Based Detection System - Shajahan et al. (2025)	95.0	94.6	94.1	94.3
PhishPatrol (Hybrid ML + PCA + Voting) - Rahane et al. (2025)	93.79	93.2	92.5	92.8
Random Forest Real-Time System - Nerurkar & Naik (2026)	86.1	81.65	93.05	87.1
Feature Fusion + GWO Model - Dandotiya et al. (2026)	97.2	96.8	96.5	96.6

7 Conclusion

This paper's research reveals that an artificial intelligence (AI) solution has been created for real-time website identification of phishing websites, using TensorFlow deep learning and a Chrome plugin. The suggested model was tested and analyzed using both malicious and authentic URLs and an accuracy rate of nearly 99% was achieved with high precision, recall and F1 score values near 0.99. The minimum values of loss and maximum values of ROC-AUC indicated that the deep learning model developed was successful in distinguishing phishing website from legitimate website with minimal false predictions.

The results of the experiments demonstrate the successful application of the proposed approach for detecting suspicious URLs by extracting heuristic features, using AI classification and running them on a blacklist. Light-weight and fast AI inference in the browser for real-time phishing detection was made possible by the integration of ONNX Runtime Web and WebAssembly. The chrome extension developed successfully sent the instant phishing warning notification while accessing suspicious websites, which enhances the cyber security of the browser and increases the protection of the user from the modern phishing attack.

References

[1] T. O. Ojewumi, G. O. Ogunleye, B. O. Oguntunde, O. Folorunsho, S. G. Fashoto, and N. Ogbu, "Performance evaluation of machine learning tools for detection of phishing attacks on web pages," Scientific African, vol. 17, 2022, Art. no. e01165, doi: 10.1016/j.sciaf.2022.e01165.

[2] R. Parameswari and S. Akashkumar, "Phish shield: AI-powered browser extension for phishing detection," International Journal of Advance Research in Multidisciplinary, vol. 3, no. 2, pp. 155–158, 2025.

[3] L. Thaçi, A. Halili, K. Vishi, and B. Rexha, "Efficient Chrome extension for phishing detection using machine learning techniques," Springer Nature, 2021.

[4] B. Lim, R. Huerta, A. Sotelo, A. Quintela, and P. Kumar, "Enhancing phishing detection through explainable AI and LLM-powered interpretability," arXiv preprint arXiv:2503.20796, 2025.

[5] P. G. Chi, L. C. Keat, N. Y. Phing, and T. B. Sian, "An efficient phishing website detection plugin service based on website trustworthiness evaluation," International Journal of Research and Innovation in Social Science (IJRISS), vol. 9, no. 8, Aug. 2025, doi: 10.47772/IJRISS.

[6] M. M. Adake and A. M. Belekar, "Real-time phishing website detection using machine learning and updating phishing probability with user feedback," International Journal of Recent Technology and

Engineering (IJRTE), vol. 12, no. 1, May 2023, doi: 10.35940/IJRTE.A7608.0512123.
[7] M. Subathra and B. K. Sivavarsni, "A machine learning approach for AI-based phishing detection and classification of malicious URLs," International Journal of Engineering Research & Technology (IJERT), vol. 15, no. 3, Mar. 2026.
[8] Q. E. ul Haq, M. H. Faheem, and I. Ahmad, "Detecting phishing URLs based on a deep learning approach to prevent cyber-attacks," Applied Sciences, vol. 14, p. 10086, 2024, doi: 10.3390/app142210086.
[9] A. U. Rehman, I. Imtiaz, and S. Javaid, "Real-time phishing URL detection using machine learning," Engineering Proceedings, vol. 107, p. 108, 2025, doi: 10.3390/engproc2025107108.
[10] S. Shajahan, T. George, J. P. V., and S. K. M., "Browser extension for phishing website detection using machine learning," in AIJR Proceedings, Kerala, India, Apr. 2025, doi: 10.21467/proceedings.7.5.
[11] A. M. Ghime, S. Bolla, and O. Kamble, "Anti phishing extension using AI and ML," International Journal of Advanced Research in Computer and Communication Engineering, vol. 14, no. 5, May 2025, doi: 10.17148/IJARCCE.2025.145119.
[12] V. Jalan, M. H. Pawshe, and M. R. Oswal, "Chrome extension for detection and prevention of phishing websites and related attacks by using machine learning," IJCRT, vol. 11, no. 5, May 2023.
[13] A. Gurung, H. Kousar, and G. Reddy, "AI-powered phishing detection and prevention system for large-scale email data security," International Journal of Novel Research and Development (IJNRD), vol. 11, no. 1, Jan. 2026.
[14] A. Rahane, D. Kolhe, and P. Pagar, "International Journal of Research Publication and Reviews," International Journal of Research Publication and Reviews, vol. 6, no. 5, pp. 10706–10716, May 2025.
[15] G. B. Prasad, S. Deepshika, and S. S. Keshavardhan, "AI-driven real-time phishing detection system," JAAFR, vol. 4, no. 4, Apr. 2026.
[16] R. W. A., L. K. U., and K. Vikas, "Phishing website detection browser extension using ML," International Journal of Recent Trends in Multidisciplinary Research, vol. 6, no. 1, pp. 93–100, Jan.–Feb. 2026, doi: 10.59256/ijrtmr.20260601011.
[17] A. N. T. S., B. P., N. Suresh, and P. V. S., "PHISHSNAP-A Chrome extension tool used for detection of phishing applying machine learning," Journal of Artificial Intelligence and Capsule Networks, vol. 6, no. 1, pp. 105–121, Mar. 2024, doi: 10.36548/jaicn.2024.1.008.
[18] S. Adhikary and U. Setia, "Real-time phishing detection using lightweight deep learning models," International Journal of Engineering Research & Technology (IJERT), vol. 15, no. 3, Mar. 2026.
[19] G. P. Nerurkar and A. Naik, "Real-time phishing website detection using a browser extension with random forest classifier," International Journal of Engineering Research & Technology (IJERT), vol. 15, no. 3, Mar. 2026.
[20] M. Dandotiya, N. K. Goyal, and A. Khunteta, "Real time identification of phishing attacks through machine learning enhanced browser extensions," Scientific Reports, vol. 16, p. 6612, 2026, doi: 10.1038/s41598-026-35655-7.