ISSN 2277-2685
IJESR/April-June. 2026/ Vol-16/Issue-2/233-240
J. Amulya Reddy *et. al.*, /International Journal of Engineering & Science Research
*Full Length Article*

# ML Approach For Detecting Suspicious Digital Interactions

**Ms. K Virija[1], J. Amulya Reddy[2], G. Harshitha Reddy[3], G. Kavya[4]**
[1]Assistant Professor; Department Of Electronics And Communication Engineering Bhoj Reddy Engineering College For Women Hyderabad India
[2,3,4]B.Tech Students; Department Of Electronics And Communication Engineering Bhoj Reddy Engineering College For Women Hyderabad India
Mail Id; jamulya2504@gmail.com[2],harshitagaddam2003@gmail.com[3]
kavyagouru0@gmail.com[4]

***Accepted 08-04-2026***


***Abstract***
*The rapid expansion of digital communication platforms and online services has significantly increased the risk of cyber threats, particularly phishing attacks and malicious web interactions. Attackers frequently exploit deceptive URLs and fraudulent websites to obtain sensitive information such as login credentials, financial data, and personal details. Traditional security mechanisms based on rule-based filtering and blacklists often struggle to detect newly emerging threats due to the constantly evolving nature of cyberattacks. To address these limitations, this study proposes a machine learning-based approach for detecting suspicious digital interactions through phishing URL classification.The proposed system analyzes structural and behavioral characteristics of URLs and applies machine learning techniques to classify them as legitimate or malicious. Several supervised learning algorithms, including Random Forest, Support Vector Machine (SVM), and XGBoost, were implemented and evaluated using labeled datasets. In addition, deep learning models such as Deep Neural Networks (DNN) and One-Dimensional Convolutional Neural Networks (CNN-1D) were employed to capture complex patterns and sequential relationships within the data. The dataset was preprocessed through feature extraction, normalization, and data cleaning techniques to improve model performance.Experimental results demonstrate that both machine learning and deep learning models are capable of effectively detecting phishing URLs with high accuracy. Among the evaluated models, ensemble learning techniques and deep learning architectures showed superior performance in identifying malicious patterns. Furthermore, a user-friendly web application was developed to allow users to input URLs and receive real-time predictions regarding their safety. The results indicate that the proposed approach provides a scalable and efficient solution for improving cybersecurity by automatically detecting suspicious digital interactions. Future enhancements may include integrating larger datasets, real-time threat intelligence, and advanced deep learning architectures to further improve detection accuracy and system adaptability.*

***Keywords***
*Machine Learning, Phishing Detection, Suspicious Digital Interactions, Cybersecurity, URL Classification, Deep Learning, Random Forest, Support Vector Machine (SVM), Convolutional Neural Network (CNN).*

## Introduction

The rapid advancement of digital communication and online commerce has significantly transformed the way individuals, organizations, and governments interact. While these developments have enabled faster connectivity and improved accessibility, they have also created opportunities for malicious activities conducted through digital platforms. Cyber threats such as phishing attacks, account hijacking, online fraud, and coordinated misinformation campaigns have become increasingly sophisticated. These threats often exploit social media platforms such as Facebook and X (formerly Twitter), messaging services like WhatsApp, and large-scale e-commerce platforms including Amazon, making detection more challenging.As the volume, speed, and diversity of digital data continue to expand, the challenge has shifted from simply collecting information to effectively analyzing large datasets in real time. Traditional rule-based security mechanisms struggle to detect evolving attack patterns because they rely on predefined signatures or manually created rules. Consequently, advanced analytical approaches are required to identify hidden patterns and emerging threats.Machine Learning (ML) has emerged as a powerful solution for detecting suspicious digital interactions. Unlike static detection systems, ML-based approaches can automatically learn behavioral patterns from historical data and adapt to new types of cyber threats. By analyzing large datasets, ML models can identify anomalies, suspicious activities, and deviations from normal behavior, enabling organizations to respond proactively to cyber risks.

## Literature Survey

The detection of suspicious digital activities has attracted considerable research attention due to the rapid growth of online services and cyber threats. Early cybersecurity systems primarily relied on rule-based and signature-based techniques to detect

malicious activities. While these approaches were effective in identifying known threats, they often failed to detect previously unseen attacks or evolving cyber strategies.With the advancement of machine learning, researchers have increasingly adopted data-driven approaches for threat detection. Supervised learning techniques have been widely applied in areas such as spam filtering, fraud detection, and phishing identification. Algorithms including Support Vector Machines (SVM), Random Forests, and Logistic Regression have demonstrated strong performance when trained on labeled datasets containing examples of legitimate and malicious activities.

**Software Requirements**

**Software Requirement Analysis**

Software Requirement Analysis (SRA) represents a fundamental stage in the Software Development Life Cycle (SDLC). This phase focuses on identifying and documenting both functional and non-functional requirements before system development begins. Proper requirement analysis ensures that the system operates effectively within defined constraints while fulfilling its intended objectives.In the proposed machine learning-based system for detecting suspicious digital interactions, requirement analysis plays a crucial role in defining system functionality, performance expectations, security considerations, and data management strategies. Since the system processes sensitive data related to cybersecurity threats, it is essential to establish well-defined requirements to maintain system reliability and accuracy.The requirement analysis process involves gathering information from stakeholders, identifying system boundaries, and determining the hardware and software infrastructure needed for implementation

**Software Tools Used**

Several software tools and platforms were utilized to develop and implement the proposed system.Python serves as the primary programming language due to its simplicity, flexibility, and extensive ecosystem of libraries for machine learning, data processing, and visualization.Dataset Sources include publicly available cybersecurity datasets obtained from repositories such as Kaggle and the UCI Machine Learning Repository. These datasets contain examples of phishing URLs and legitimate web addresses used for training and evaluation.Jupyter Notebook and Anaconda were employed as the development environment. Jupyter Notebook provides an interactive platform for coding, data visualization, and experimentation, while the Anaconda distribution simplifies dependency management and package installation.TensorFlow was used for developing deep learning models such as Convolutional Neural Networks (CNN-1D). It provides powerful tools for building and training neural networks.Scikit-learn was applied for implementing traditional machine learning algorithms and evaluating model performance using metrics such as accuracy, precision, recall, and F1-score.

Google Cloud services were used for large-scale data processing and storage, enabling integration with analytics platforms such as BigQuery and AI tools.

**Simulation and Analysis Tools**

Simulation and analysis tools play an important role in developing machine learning systems designed to detect suspicious digital behavior. Simulation environments allow researchers to generate synthetic datasets that replicate real-world interaction patterns, including both normal user activities and malicious behavior. These simulated environments help address the scarcity of labeled data, particularly for rare cyberattack events.Platforms such as MATLAB and SimPy are commonly used to model user behavior, network traffic, and attack scenarios in controlled experimental settings. These simulations enable researchers to test detection algorithms under different conditions and evaluate system performance.For analytical processing, machine learning frameworks including TensorFlow and Scikit-learn provide tools for feature extraction, model training, classification, and clustering. Performance metrics such as precision, recall, and F1-score are commonly used to evaluate detection models.Advanced network simulation tools like NS-3 and OMNeT++ can simulate large-scale network interactions and intrusion scenarios. These platforms allow researchers to analyze evolving attack strategies and evaluate how detection systems respond to adaptive adversaries.In addition, large-scale data processing platforms such as Apache Spark and Keras support distributed computation and rapid model prototyping. Integration with Explainable Artificial Intelligence (XAI) techniques further improves system transparency by enabling analysts to understand the reasoning behind model predictions.

**System Architecture**

The expansion of digital communication platforms has led to an enormous increase in online interactions, including email communication, social media activity, financial transactions, and network communications. While these technologies offer numerous benefits, they also expose systems to cyber threats such as phishing, identity theft, fraudulent transactions, and unauthorized access.Traditional cybersecurity systems typically rely on predefined rules and manual monitoring methods. However, these approaches are limited in their ability to detect sophisticated attacks and emerging threats. Machine learning-based systems provide a more adaptive solution by analyzing

patterns within digital interactions and identifying suspicious activities automatically.

**Overall System Architecture**

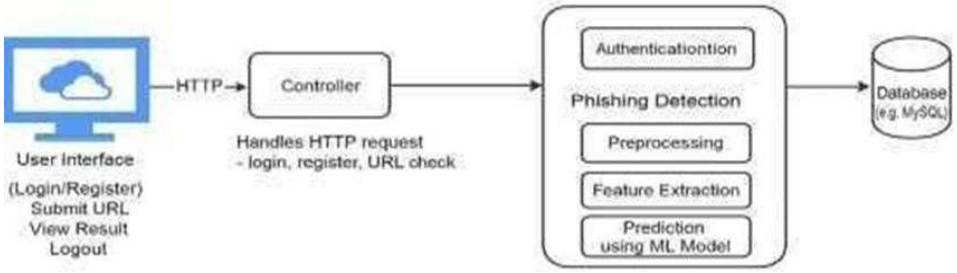


**Figure 1: System Architecture**

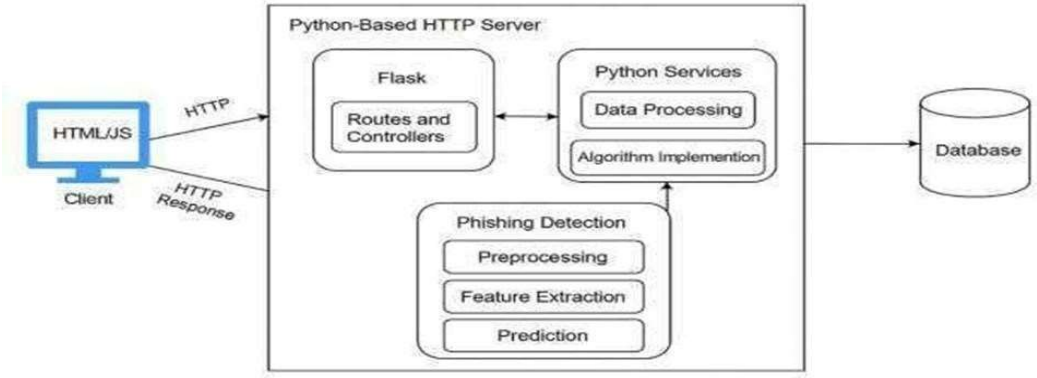


**Figure 2: Technical Architecture**

The proposed system architecture consists of multiple layers responsible for data collection, processing, model training, and decision-making. The process begins with the data ingestion stage, where interaction data such as user activity logs, network traffic, and communication records are collected.This data is then processed through preprocessing stages, which include data cleaning, normalization, and feature extraction. These steps transform raw data into structured inputs suitable for machine learning algorithms.The processed data is fed into machine learning models developed using frameworks such as TensorFlow and Scikit-learn. These models analyze interaction patterns to identify anomalies and suspicious activities. A decision-making component evaluates model predictions and flags potential threats based on predefined thresholds.The results are visualized through monitoring dashboards, allowing analysts to investigate suspicious activities. Feedback mechanisms allow the system to improve over time by incorporating expert validation.

**Data Processing Layer**

The data processing layer prepares raw interaction data for analysis. Raw datasets often contain inconsistencies, missing values, and noisy entries that may negatively affect model performance.Data cleaning techniques remove duplicate records, corrupted entries, and irrelevant information. Outliers in user behavior patterns are carefully examined to ensure that abnormal patterns are accurately captured without introducing bias.Normalization techniques scale numerical attributes into standardized ranges, improving algorithm convergence and model stability.

**Model Layer**

The model layer represents the core analytical component of the system. In this layer, both traditional machine learning algorithms and deep learning models are used to detect suspicious interactions.Machine learning algorithms such as Random Forest, Support Vector Machine (SVM), and XGBoost are used for classification tasks based on engineered features. These algorithms are effective at identifying patterns and anomalies in structured datasets.Deep learning models including Deep Neural Networks (DNN) and 1D Convolutional Neural Networks (CNN-1D) are also implemented to capture complex patterns in sequential interaction data.Model training involves adjusting parameters using labeled datasets to minimize prediction errors. Hyperparameter tuning is performed to improve performance by optimizing parameters such as learning rates and network architecture. Cross-validation techniques ensure that models generalize well to unseen data.Performance is evaluated using metrics such as accuracy, precision, recall, and F1-score.

**Methodology**

The increasing reliance on digital communication platforms and online transaction systems has

resulted in a massive growth of user-generated interactions. Monitoring these interactions manually has become increasingly difficult, especially when malicious activities such as phishing, fraudulent links, and suspicious user behavior are continuously evolving. Conventional rule-based security mechanisms are often limited because they rely on predefined patterns and cannot easily adapt to new attack strategies. Consequently, intelligent data-driven approaches are required to automatically identify abnormal or potentially harmful activities in digital environments.To address these challenges, this research adopts a machine learning-based methodology for detecting suspicious digital interactions, particularly phishing URLs. The proposed framework processes large volumes of interaction data, extracts meaningful features, and trains predictive models capable of distinguishing between legitimate and malicious patterns. Machine learning techniques allow the system to learn hidden relationships within the dataset and identify anomalies that may not be observable through traditional analysis methods.The methodology includes several major stages: data collection, preprocessing, model training, and performance evaluation. Multiple machine learning and deep learning algorithms are implemented to determine the most effective approach for suspicious interaction detection. Special attention is given to dataset imbalance, which commonly occurs in cybersecurity applications where malicious samples are significantly fewer than legitimate ones.

**Data Collection**

Data collection represents a critical component in developing an effective machine learning model. The quality, diversity, and representativeness of the dataset directly influence the accuracy and generalization capability of the trained models. In this study, datasets related to phishing URLs and legitimate websites were obtained from publicly available cybersecurity repositories such as Kaggle and the UCI Machine Learning Repository.The collected data includes several attributes that characterize URL behavior and structural properties. These attributes enable the detection system to analyze patterns associated with phishing attacks. Data was gathered from various digital environments where suspicious interactions may occur, including web browsing activity, communication platforms, and network logs.The main objective of the data collection stage is to obtain sufficient information to represent both normal and malicious interaction patterns. The dataset contains labeled examples that allow supervised machine learning algorithms to learn classification boundaries between legitimate and phishing URLs.

**Types of Data Collected**

To ensure comprehensive analysis, multiple types of data features were considered.

**Structured Data**

Structured data refers to organized information stored in predefined formats such as tables or databases. In the context of phishing detection, structured features include URL length, domain age, number of special characters, presence of HTTPS protocols, redirection behavior, and the number of subdomains. These features can be directly processed using machine learning algorithms.

**Unstructured Data**

Unstructured data includes text-based information such as webpage content, messages, or email text associated with URLs. Detecting malicious intent from textual information requires natural language processing techniques such as tokenization, vectorization, and feature extraction. These techniques transform textual information into numerical representations suitable for machine learning models.

**Behavioral Data**

Behavioral data describes patterns related to user activity and interaction sequences. Examples include browsing patterns, click frequency, login activity, and session duration. Sudden deviations from typical user behavior may indicate suspicious activity or compromised accounts.

**Challenges in Data Collection and Handling**

While collecting data for machine learning applications, several challenges may arise that can affect system performance. These challenges include data noise, missing information, and large data volumes.

**Data Noise**

Noise refers to irrelevant or incorrect information that may distort actual patterns within the dataset. Noise can originate from incorrect labeling, data corruption, or environmental disturbances. If not handled properly, noisy data may reduce model accuracy and lead to misleading predictions.

**Missing or Incomplete Data**

Missing values occur when certain data attributes are not recorded or become unavailable due to technical issues such as network failures or logging errors. Incomplete datasets can negatively affect model training and reduce the reliability of predictions. Various data imputation methods such as interpolation or statistical estimation are used to address missing values.

**Data Preprocessing**

Raw datasets cannot be directly used for machine learning tasks because they may contain inconsistencies, redundant information, or irregular values. Therefore, several preprocessing operations were performed to prepare the dataset for analysis.Initially, the dataset was cleaned by removing duplicate entries and correcting inconsistent values. Missing attributes were handled using suitable imputation techniques to preserve data continuity. After cleaning, normalization techniques were applied to scale numerical attributes

within a consistent range. This step improves the convergence of algorithms such as Support Vector Machines and neural networks.Feature selection was also performed to identify the most relevant attributes influencing phishing detection. Selected features were then converted into structured formats appropriate for different machine learning models.

**Results and Discussion**

The rapid expansion of digital communication and online services has significantly increased the risk of cyber threats, particularly phishing attacks and malicious websites. Cybercriminals often create deceptive URLs to trick users into revealing confidential information such as login credentials, banking details, and personal data. Traditional security mechanisms frequently struggle to detect newly emerging threats because they rely on predefined rules and signatures. To address this limitation, this study proposes a machine learning-based system that detects suspicious digital interactions by identifying phishing URLs.The developed system analyzes the structural and behavioral characteristics of URLs and classifies them as either legitimate or malicious. Several machine learning and deep learning models were trained and evaluated using labeled datasets. The system was also integrated with a web-based interface that allows users to input URLs and receive instant predictions regarding their safety. The results of the experiments are presented in two phases. In the first phase, traditional machine learning models such as Random Forest, Support Vector Machine (SVM), and XGBoost were evaluated and compared. In the second phase, deep learning models including Deep Neural Networks (DNN) and Convolutional Neural Networks (CNN-1D) were implemented to analyze their ability to capture complex patterns in the dataset. The comparative evaluation helps determine the most effective model for detecting phishing URLs and suspicious digital interactions.

**Performance Metrics Overview**

To evaluate the effectiveness of the proposed detection system, several performance metrics were used. These metrics provide a comprehensive understanding of how accurately the models classify URLs and how well they detect malicious activities. Since phishing detection is essentially a binary classification problem, relying on a single metric such as accuracy may not provide sufficient insight into model performance, especially when datasets are imbalanced.Accuracy measures the overall proportion of correctly classified URLs among all predictions made by the model. Although accuracy provides a general indication of model performance, it may be misleading in situations where legitimate URLs significantly outnumber phishing URLs. Therefore, additional evaluation metrics were considered to obtain a more reliable assessment.Precision measures the proportion of URLs predicted as malicious that are actually malicious. A model with high precision generates fewer false alarms, which is important in cybersecurity systems to avoid unnecessary warnings for users. Recall, also known as sensitivity, evaluates the ability of the model to correctly identify actual phishing URLs. A high recall value indicates that the system successfully detects most malicious links and minimizes the risk of missed attacks.The F1-score represents the harmonic mean of precision and recall and provides a balanced evaluation when both false positives and false negatives must be minimized. Another useful evaluation metric is the Receiver Operating Characteristic – Area Under Curve (ROC-AUC), which measures the model's ability to distinguish between legitimate and malicious URLs across various classification thresholds. Additionally, confusion matrix analysis was used to observe the distribution of true positives, true negatives, false positives, and false negatives. These metrics collectively provide a comprehensive framework for evaluating the reliability and effectiveness of the proposed detection models.

**Accuracy Comparison Analysis**

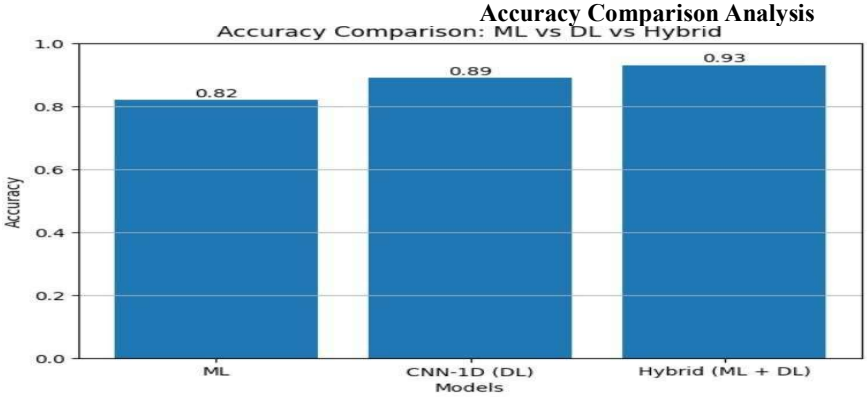


**Fig 1: Accuracy Comparison of ML and DL Models**

The accuracy comparison analysis evaluates how effectively each model classifies URLs as legitimate or phishing. Experimental results show that both machine learning and deep learning models achieve

relatively high accuracy levels, although performance varies across algorithms.Among the machine learning models, Random Forest achieved an accuracy of approximately 93%, making it the best-performing model in this category. This performance can be attributed to the ensemble nature nature of Random Forest, which combines multiple decision trees to improve prediction stability and reduce overfitting. The Support Vector Machine model achieved an accuracy of around 90%, demonstrating strong classification capability when data features are clearly separable. However, its performance slightly decreases when handling highly complex or nonlinear patterns. XGBoost, although known for its powerful gradient boosting technique, achieved a comparatively lower accuracy of approximately 85% in this study. This may be due to the need for further hyperparameter tuning and more refined feature engineering.Among deep learning models, the Deep Neural Network achieved an accuracy of approximately 90%. While DNNs are capable of learning high-level abstractions in data, their performance depends heavily on network architecture and parameter optimization. The CNN-1D model achieved an accuracy between 92% and 93%, demonstrating strong capability in detecting sequential patterns in interaction data. Because phishing behaviors often involve repeated patterns or structural similarities in URLs, CNN-1D effectively captures these characteristics. The hybrid model, which integrates both machine learning and deep learning techniques, achieved the highest overall accuracy of approximately 93–94%. This indicates that combining feature-based learning with automatic feature extraction can significantly enhance detection performance.

**Precision Comparison Analysis**

Precision analysis evaluates the reliability of model predictions by measuring how many URLs predicted as phishing are actually malicious. High precision is essential in phishing detection systems because excessive false positives can negatively affect user trust and system usability.Experimental results show that Random Forest and CNN-1D achieved the highest precision values of approximately 92%. These models demonstrate strong capability in accurately identifying malicious URLs while minimizing incorrect classifications. The Deep Neural Network achieved a precision of around 89%, indicating relatively reliable predictions but slightly higher false positive rates compared to the top-performing models. The Support Vector Machine achieved a precision value of approximately 89%, which is slightly lower than Random Forest but still demonstrates good classification capability.XGBoost showed the lowest precision among the evaluated models, with a value of around 84%. This indicates that a higher number of legitimate URLs were incorrectly classified as phishing. Although XGBoost remains a powerful algorithm, its performance in this case may require improved feature engineering or parameter tuning.

**Recall Comparison Analysis**

Recall analysis focuses on the ability of the detection system to correctly identify actual phishing URLs. In cybersecurity applications, recall is particularly important because missing a malicious link could lead to severe security breaches.The experimental results show that CNN-1D achieved the highest recall value of approximately 93%, indicating its strong ability to detect most phishing URLs present in the dataset. Random Forest also demonstrated excellent recall performance with a similar value of approximately 93%, confirming its effectiveness as a machine learning baseline model.The Deep Neural Network and Support Vector Machine both achieved recall values of approximately 90%, which indicates a good balance between detection capability and classification stability. However, XGBoost showed a slightly lower recall value of around 85%, suggesting that some phishing URLs were not correctly detected.These results highlight that deep learning models, particularly CNN-1D, are more effective in identifying malicious patterns and detecting suspicious interactions. Their ability to learn complex relationships in data allows them to capture subtle indicators of phishing behavior.

**Phase-1 Results: URL-Based Detection**

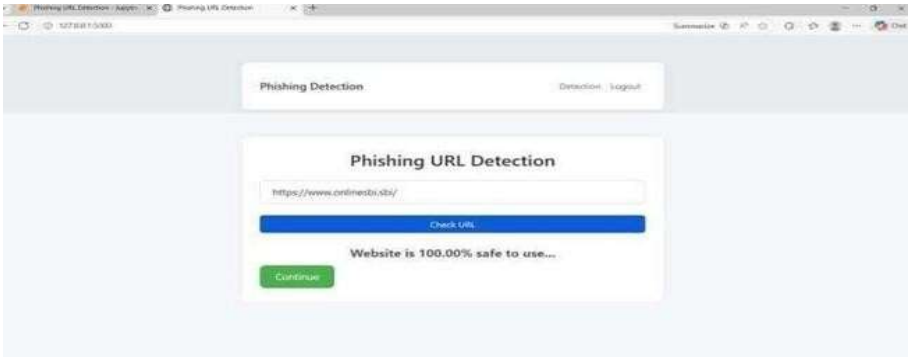


**Fig 2: Phase-1 Result 1**


**Fig 3: Phase-1 Result 2**

In the first phase of the project, the system focused on building and evaluating machine learning models capable of detecting phishing URLs based on extracted features. Users provide a URL as input, and the system analyzes several characteristics such as URL length, presence of special characters, domain structure, use of HTTPS protocols, and redirection behavior.These features are processed and passed through trained machine learning algorithms including Random Forest, SVM, and XGBoost. Based on learned patterns from labeled datasets, the models classify the URL as either legitimate or malicious. This phase primarily concentrated on dataset preparation, model training, and performance evaluation.The results of Phase 1 demonstrated that machine learning techniques are effective for phishing detection, achieving high classification accuracy and providing a reliable baseline for further system development.

**Phase-2 Results: Application Implementation**

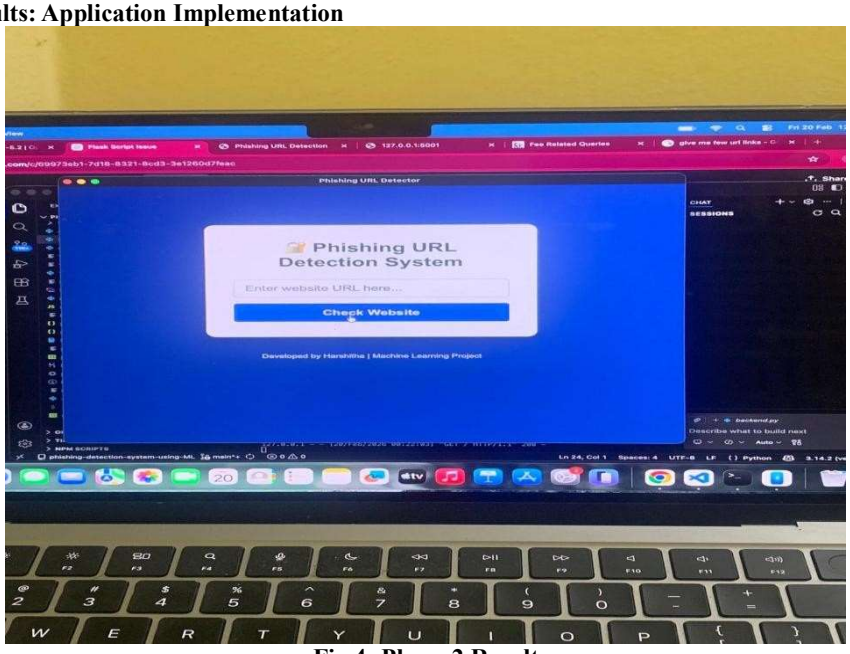


**Fig 4: Phase-2 Result**

In the second phase of the project, the trained detection models were integrated into a real-time application to make the system practical and accessible to users. A user-friendly interface was developed where users can input URLs and receive immediate feedback regarding the safety of the link.When a user submits a URL, the system performs preprocessing, extracts relevant features, and applies the trained classification models to determine whether the URL is safe or potentially malicious. The prediction is generated within seconds, enabling real-time threat detection.The application displays clear results such as “Safe” or “Unsafe” along with confidence scores that indicate the reliability of the prediction. This design allows both technical and non-technical users to easily evaluate suspicious links and avoid potential cyber threats.

**Discussion**

The experimental results demonstrate that machine learning techniques are effective in detecting phishing URLs and suspicious digital interactions. While traditional machine learning models provide reliable baseline performance, deep learning models

offer improved detection capabilities by learning complex relationships within the dataset.The results also highlight the importance of selecting appropriate features for phishing detection. URL characteristics such as domain structure, length, special characters, and redirection behavior play a crucial role in identifying malicious links. Additionally, balancing precision and recall is essential to ensure that the system detects most phishing attacks while minimizing false alarms.Another important observation is the trade-off between model complexity and interpretability. Traditional models such as decision trees and support vector machines are easier to interpret, while deep learning models may achieve higher accuracy but provide less transparency. Therefore, selecting the appropriate model depends on the specific requirements of the cybersecurity system.

**Conclusion**

The rapid expansion of digital services has increased exposure to cyber threats such as phishing attacks and malicious websites. Traditional rule-based security systems are often unable to detect evolving threats effectively.

This study presented a machine learning-based approach for detecting suspicious digital interactions, focusing specifically on phishing URL identification. The proposed system performs preprocessing, feature extraction, and classification using multiple machine learning algorithms.Experimental results demonstrate that machine learning models can accurately identify malicious URLs and significantly improve threat detection efficiency. The integration of a user-friendly web interface allows users to verify links in real time, enhancing cybersecurity awareness.Overall, the system provides a scalable and automated solution for detecting phishing attacks and improving online safety.

**Future Scope**

Future research can extend this work in several directions.Advanced deep learning architectures such as recurrent neural networks and transformer models could be explored to improve detection accuracy. Incorporating natural language processing techniques may also enhance the ability to analyze webpage content and detect hidden phishing indicators.Real-time adaptive learning systems could enable models to update automatically as new threats emerge. Integrating machine learning models with cloud computing and edge computing technologies would allow faster detection across distributed environments.Additionally, privacy-preserving techniques such as federated learning could enable collaborative model training without exposing sensitive user data.With continued advancements in artificial intelligence and cybersecurity technologies, machine learning-based detection systems will play a crucial role in protecting digital platforms from evolving cyber threats.

**References**

[1] R. Verma and K. Dyer, “On the Characterization of Phishing URLs Using Lexical and Host-Based Features,” in *Proceedings of the IEEE International Conference on Communications*, London, UK, 2015, pp. 1–6.

[2] M. Sahingoz, E. Buber, O. Demir, and B. Diri, “Machine Learning Based Phishing Detection from URLs,” *Expert Systems with Applications*, vol. 117, pp. 345–357, 2019.

[3] S. Marchal, J. François, R. State, and T. Engel, “PhishStorm: Detecting Phishing with Streaming Analytics,” *IEEE Transactions on Network and Service Management*, vol. 11, no. 4, pp. 458–471, 2014.

[4] A. Le, A. Markopoulou, and M. Faloutsos, “PhishDef: URL Names Say It All,” in *Proceedings of the IEEE INFOCOM Conference*, Turin, Italy, 2013, pp. 191–195.

[5] J. Ma, L. K. Saul, S. Savage, and G. M. Voelker, “Beyond Blacklists: Learning to Detect Malicious Web Sites from Suspicious URLs,” in *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, Las Vegas, USA, 2009, pp. 1245–1254.

[6] A. Blum, B. Wardman, T. Solorio, and G. Warner, “Lexical Feature Based Phishing URL Detection Using Online Learning,” in *Proceedings of the ACM Workshop on Artificial Intelligence and Security*, Chicago, USA, 2010, pp. 54–60.

[7] H. Abutair and A. Belghith, “Using Case-Based Reasoning for Phishing Detection,” *Procedia Computer Science*, vol. 109, pp. 281–288, 2017.

[8] A. A. Almomani, B. B. Gupta, S. Wan, A. Altaher, and E. Manickam, “Phishing Dynamic Evolving Neural Fuzzy Framework for Online Detection ‘Zero-Day’ Phishing Email,” *Indian Journal of Science and Technology*, vol. 6, no. 1, pp. 1–9, 2013.

[9] W. Han, J. Xue, Y. Wang, and X. Zhang, “Phishing Detection Based on Deep Learning,” *IEEE Access*, vol. 8, pp. 134504–134513, 2020.

[10] M. S. Rahman, S. Al-Shaikhli, and S. Al-Shaikhli, “A Machine Learning Approach for Detecting Phishing Websites,” *International Journal of Advanced Computer Science and Applications*, vol. 11, no. 7, pp. 1–8, 2020.