IARJSET ISSN (O) 2393-8021, ISSN (P) 2394-1588

International Advanced Research Journal in Science, Engineering and Technology

Impact Factor 8.311 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 13, Issue 3, March 2026

DOI: 10.17148/IARJSET.2026.13351

DETECTING MALICIOUS URLs USING DATA ANALYTICS AND MACHINE LEARNING

Vignesh S[1], Dr. K. Santhi[2]

Student, Department of Information Technology, Dr. N.G.P Arts and Science College, Coimbatore[1]

Professor, Department of Information Technology, Dr. N.G.P Arts and Science College, Coimbatore[2]

Abstract: The exponential growth of internet services and digital transactions has significantly increased exposure to cyber threats, particularly phishing attacks, malware dissemination, and fraudulent web activities. Malicious URLs serve as a primary attack vector in these cyber incidents, enabling adversaries to manipulate users, extract confidential information, and compromise enterprise security infrastructures. Conventional detection mechanisms, primarily based on static blacklists and signature matching, are inadequate in identifying newly generated or zero-day malicious URLs, thereby necessitating intelligent and adaptive detection strategies.

To address these limitations, this work proposes an advanced malicious URL detection framework built upon data analytics and machine learning methodologies. The system employs comprehensive feature engineering techniques to extract meaningful lexical, statistical, and structural characteristics from URLs. These features include entropy measurement, character frequency distribution, URL length, special character density, domain-related indicators, and hierarchical path depth analysis. Extracted features are standardized using structured preprocessing pipelines to ensure stability and consistency during model training and inference.

The classification core of the system is implemented using the XGBoost algorithm, selected for its robustness, high predictive performance, and capability to model complex nonlinear relationships. To further enhance reliability, a heuristic-based red flag detection layer is integrated alongside the machine learning model, forming a hybrid detection architecture. This layered approach improves resilience against obfuscation techniques and stealthy phishing strategies that attempt to evade automated detection.

The backend infrastructure is developed using Python, leveraging Scikit-Learn pipelines for preprocessing and model integration. An interactive dashboard interface enables real-time URL analysis, risk scoring, and visualization of feature contributions. Experimental evaluation demonstrates high classification accuracy, strong generalization capability on unseen datasets, and reduced false positive rates.

Overall, the proposed system delivers a scalable, efficient, and explainable malicious URL detection solution, contributing to strengthened cybersecurity defenses in modern web environments.

Keywords: Malicious URL Detection, Machine Learning, XGBoost, Data Analytics, Cybersecurity, Phishing Detection, Feature Engineering, Entropy Analysis, Scikit-Learn, SHAP.

I. INTRODUCTION

In the digital era, the internet has become a fundamental part of communication, commerce, education, and governance. However, this rapid digital transformation has also increased the risk of cyberattacks. Among various cyber threats, malicious URLs are one of the most common attack vectors used in phishing campaigns, malware distribution, identity theft, and financial fraud.

Traditional security mechanisms such as signature-based detection and static blacklists are insufficient in detecting newly generated malicious URLs. Attackers continuously modify domain names, use URL shortening services, and apply obfuscation techniques to evade detection systems. Therefore, there is a need for an intelligent and data-driven approach capable of identifying suspicious URLs in real time.

IARJSET ISSN (O) 2393-8021, ISSN (P) 2394-1588

International Advanced Research Journal in Science, Engineering and Technology

Impact Factor 8.311 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 13, Issue 3, March 2026

DOI: 10.17148/IARJSET.2026.13351

This project proposes a Machine Learning-based Malicious URL Detection System that uses data analytics techniques to extract meaningful features from URLs and classify them using supervised learning algorithms. The system applies a hybrid intelligence approach combining XGBoost classification and heuristic red flag detection.

Feature extraction includes lexical features (length, special characters), statistical measures (entropy, randomness), and structural properties (subdomain depth, path complexity). The extracted features are processed using StandardScaler within a Scikit-Learn pipeline to ensure robust inference.

The proposed system also integrates explainable AI techniques using SHAP (SHapley Additive Explanations) to interpret prediction results, increasing transparency and trust in cybersecurity decision-making.

Overall, this system demonstrates how machine learning and data analytics can significantly enhance modern web security infrastructures.

II. LITERATURE REVIEW

The rapid growth of internet usage has led to a significant rise in cyber threats such as phishing attacks, malware distribution, and online financial fraud. Malicious URLs are widely used by attackers to deceive users into visiting fraudulent websites designed to steal sensitive information or install malicious software. Traditional security solutions, particularly blacklist-based detection systems, often fail to detect newly generated malicious URLs. As a result, researchers have increasingly focused on developing intelligent detection systems based on data analytics and machine learning techniques [1], [2].

Early research in malicious URL detection primarily focused on analyzing the structural and lexical characteristics of URLs. Researchers observed that malicious URLs often exhibit suspicious patterns such as unusual string lengths, excessive use of special characters, abnormal subdomain structures, and misleading tokens designed to mimic legitimate websites. By extracting these features, machine learning models could effectively distinguish between legitimate and malicious URLs. Studies demonstrated that supervised learning algorithms could detect previously unseen threats by identifying patterns within large datasets, thereby outperforming traditional blacklist-based approaches [1].

As research progressed, classification-based approaches using machine learning algorithms such as Decision Trees, Random Forests, and Support Vector Machines were widely explored. These methods relied on carefully engineered features extracted from URL structures and domain metadata. Experimental results showed that such models significantly improved detection accuracy while reducing false positives when compared with rule-based filtering systems. However, these approaches required continuous feature engineering and frequent retraining to remain effective against evolving cyberattack techniques [2].

With the advancement of deep learning technologies, researchers began applying neural network architectures such as Convolutional Neural Networks (CNNs) and Recurrent Neural Networks (RNNs) to malicious URL detection. These models are capable of automatically learning complex patterns from raw URL text without requiring extensive manual feature engineering. Deep learning approaches have demonstrated promising performance in identifying subtle patterns and sequential relationships within URLs. However, they typically require large annotated datasets and significant computational resources for training and deployment. Furthermore, deep learning models often operate as “black box” systems, providing limited interpretability regarding how classification decisions are made [3].

Recent research has emphasized the use of advanced machine learning algorithms such as Gradient Boosting methods to enhance detection performance. One of the most widely used techniques is Extreme Gradient Boosting (XGBoost), which offers high predictive accuracy and efficient model training through optimized tree-based boosting algorithms. XGBoost has been successfully applied in various cybersecurity applications due to its ability to handle large datasets, manage missing values, and provide scalable model performance [4].

In addition to improving detection accuracy, modern research has also focused on enhancing model interpretability. Explainable Artificial Intelligence (XAI) techniques such as SHAP (SHapley Additive exPlanations) allow researchers to understand how individual features influence model predictions. By providing transparent explanations for classification outcomes, SHAP helps cybersecurity analysts better interpret model decisions and increases trust in automated detection systems [5].

IARJSET ISSN (O) 2393-8021, ISSN (P) 2394-1588

International Advanced Research Journal in Science, Engineering and Technology

Impact Factor 8.311 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 13, Issue 3, March 2026

DOI: 10.17148/IARJSET.2026.13351

Machine learning frameworks such as Scikit-learn have played a crucial role in implementing and evaluating malicious URL detection models. These frameworks provide robust tools for data preprocessing, model training, and performance evaluation, enabling researchers to develop efficient detection pipelines [6]. Similarly, programming environments like Python have become the dominant platform for implementing machine learning-based cybersecurity solutions due to their extensive libraries and strong community support [7].

For deployment purposes, modern detection systems increasingly integrate web-based frameworks and database management systems. Tools such as Streamlit allow developers to create interactive dashboards for real-time URL analysis and visualization of prediction results [8]. Meanwhile, database systems such as PostgreSQL are commonly used to store analyzed URLs, prediction outcomes, and system logs for further monitoring and analysis [9]. Additionally, cybersecurity guidelines provided by organizations such as OWASP highlight best practices for preventing phishing attacks and strengthening web security infrastructures [10].

The proposed system builds upon these existing research efforts by integrating an XGBoost-based classification model with heuristic rule-based validation and explainable AI techniques. By combining machine learning predictions with interpretable analysis and scalable deployment mechanisms, the system aims to provide a robust and practical solution for malicious URL detection.

In summary, malicious URL detection techniques have evolved from simple rule-based filtering systems to advanced machine learning and hybrid detection frameworks. While significant progress has been achieved, challenges remain in ensuring scalability, interpretability, and adaptability against emerging cyber threats. The proposed approach addresses these challenges by combining high-performance classification models with explainable AI and structured deployment architecture.

III. PROBLEM STATEMENT

Malicious URLs pose a serious cybersecurity threat to individuals and organizations. Traditional detection systems rely heavily on blacklists and signature-based techniques, which fail to detect newly created or obfuscated malicious URLs. Attackers continuously generate dynamic URLs using domain generation algorithms, URL shortening, and encoding techniques to bypass security filters. As a result, users remain vulnerable to phishing attacks, credential theft, and malware infections.

Existing machine learning solutions may suffer from high false positives, lack of explainability, or poor generalization to unseen data. Therefore, there is a need for a scalable, explainable, and hybrid detection system capable of accurately identifying malicious URLs using data-driven approaches.

IV. METHODOLOGY

The proposed malicious URL detection framework follows a structured and systematic pipeline that integrates data analytics, supervised machine learning, and heuristic validation. The overall workflow is designed to ensure high detection accuracy, robustness against obfuscation techniques, and explainability in prediction outcomes. The methodology is divided into seven major stages as described below.

Step 1: Data Collection

- The first stage involves constructing a comprehensive and labeled dataset consisting of both benign and malicious URLs. Benign URLs represent legitimate websites obtained from trusted sources, while malicious URLs include phishing links, malware distribution domains, and fraudulent web addresses collected from publicly available cybersecurity repositories and threat intelligence feeds.

- To ensure dataset diversity and generalization capability, URLs from different categories such as banking phishing, e-commerce fraud, social media impersonation, and domain spoofing are included. Each URL is labeled as either *benign (0)* or *malicious (1)*. The dataset is then divided into training and testing subsets to evaluate model performance objectively.

- Proper data balancing techniques are applied where necessary to avoid bias toward the majority class, ensuring that the classifier learns meaningful decision boundaries rather than relying on skewed distributions.

IARJSET ISSN (O) 2393-8021, ISSN (P) 2394-1588

International Advanced Research Journal in Science, Engineering and Technology

Impact Factor 8.311 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 13, Issue 3, March 2026

DOI: 10.17148/IARJSET.2026.13351

Step 2: Feature Extraction

- In this stage, raw URLs are transformed into structured numerical representations suitable for machine learning algorithms. The system performs feature engineering to extract lexical, statistical, and structural attributes from each URL.
- The extracted features include:
- **URL Length:** Malicious URLs often use excessively long strings to hide suspicious tokens.
- **Special Character Count:** Characters such as "@", "-", "_", "=", "%", and multiple slashes may indicate obfuscation attempts.
- **Digit Count:** High frequency of numeric characters may suggest automatically generated domains.
- **Entropy Score:** Shannon entropy is computed to measure randomness within the URL string. Higher entropy values often indicate domain generation algorithms (DGA) or encoded patterns.
- **Subdomain Depth:** The number of subdomains is analyzed, as phishing URLs frequently use deep subdomain structures.
- **Path Depth:** Excessively nested directory structures may signal malicious intent.
- **Suspicious Keyword Detection:** The system scans for commonly misused terms such as "login", "verify", "update", "secure", or brand impersonation tokens.
- **Institutional Domain Whitelisting:** Domains ending in trusted extensions such as *.edu*, *.gov*, or *.ac.in* are evaluated under a whitelist validation mechanism to reduce false positives.
- These features collectively capture both statistical irregularities and structural anomalies present in malicious URLs.

Step 3: Data Preprocessing

- After feature extraction, preprocessing is performed to prepare the dataset for model training. Since feature scales may vary significantly (e.g., entropy values versus URL length), normalization is required to prevent bias during model learning.
- A **StandardScaler** is implemented within a Scikit-Learn pipeline to standardize features by removing the mean and scaling to unit variance. This ensures consistent distribution during both training and inference phases.
- Additionally, missing values (if any) are handled, and data integrity checks are performed. The preprocessing pipeline is saved to ensure reproducibility and consistent transformation of new incoming URLs during deployment.

Step 4: Model Training

- The core classification engine of the system is based on the **XGBoost (Extreme Gradient Boosting)** algorithm. XGBoost is selected due to its high performance, ability to handle structured data efficiently, and robustness against overfitting through regularization mechanisms.
- The model is trained using the preprocessed feature set. During training, XGBoost constructs multiple decision trees sequentially, where each tree attempts to correct errors made by the previous ones. This gradient boosting process enables the model to capture complex nonlinear relationships between features and classification labels.
- Hyperparameter tuning techniques such as grid search or cross-validation are employed to optimize parameters including learning rate, tree depth, and number of estimators. The final model is evaluated using performance metrics such as:
- Accuracy
- Precision
- Recall
- F1-Score
- ROC-AUC Score
- This ensures that the classifier achieves balanced detection performance without excessive false positives or false negatives.

Step 5: Hybrid Heuristic Layer

- To enhance detection reliability, a rule-based heuristic validation layer is integrated alongside the machine learning model. This hybrid approach strengthens the system against adversarial manipulation and edge cases.
- The heuristic module checks for:
- Suspicious domain patterns not captured by the model
- Known phishing keyword combinations
- Abnormal character sequences

IARJSET ISSN (O) 2393-8021, ISSN (P) 2394-1588

International Advanced Research Journal in Science, Engineering and Technology

Impact Factor 8.311 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 13, Issue 3, March 2026

DOI: 10.17148/IARJSET.2026.13351

- Mismatch between domain name and brand identity
- Institutional whitelist verification
- If heuristic indicators exceed a defined threshold, the system can adjust the final risk score accordingly. This layered defense strategy improves robustness and reduces the probability of false negatives.

Step 6: Prediction and Risk Scoring

- Once the trained model receives a new URL input, the following process occurs:
- Feature extraction is applied.
- Standardization is performed using the saved preprocessing pipeline.
- The XGBoost classifier outputs a probability score representing the likelihood of maliciousness.
- Instead of returning only a binary result, the system provides a **risk probability score** (e.g., 0–100%). This probabilistic interpretation allows more flexible decision-making in real-world deployments.
- To enhance transparency, **SHAP (SHapley Additive Explanations)** analysis is used to compute feature importance for each prediction. SHAP values quantify the contribution of individual features toward the classification outcome, enabling users to understand why a specific URL was flagged as malicious.

Step 7: Output Display and Visualization

- The final stage presents the detection results through an interactive dashboard interface. The user interface displays:
- URL classification (Benign / Malicious)
- Risk probability score
- Feature contribution visualization
- Highlighted suspicious attributes
- The dashboard is designed for clarity and usability, enabling cybersecurity analysts and institutional administrators to quickly interpret results and take appropriate action.
- The integration of predictive analytics with explainable visualization ensures that the system is not only accurate but also interpretable and practical for deployment in enterprise environments.

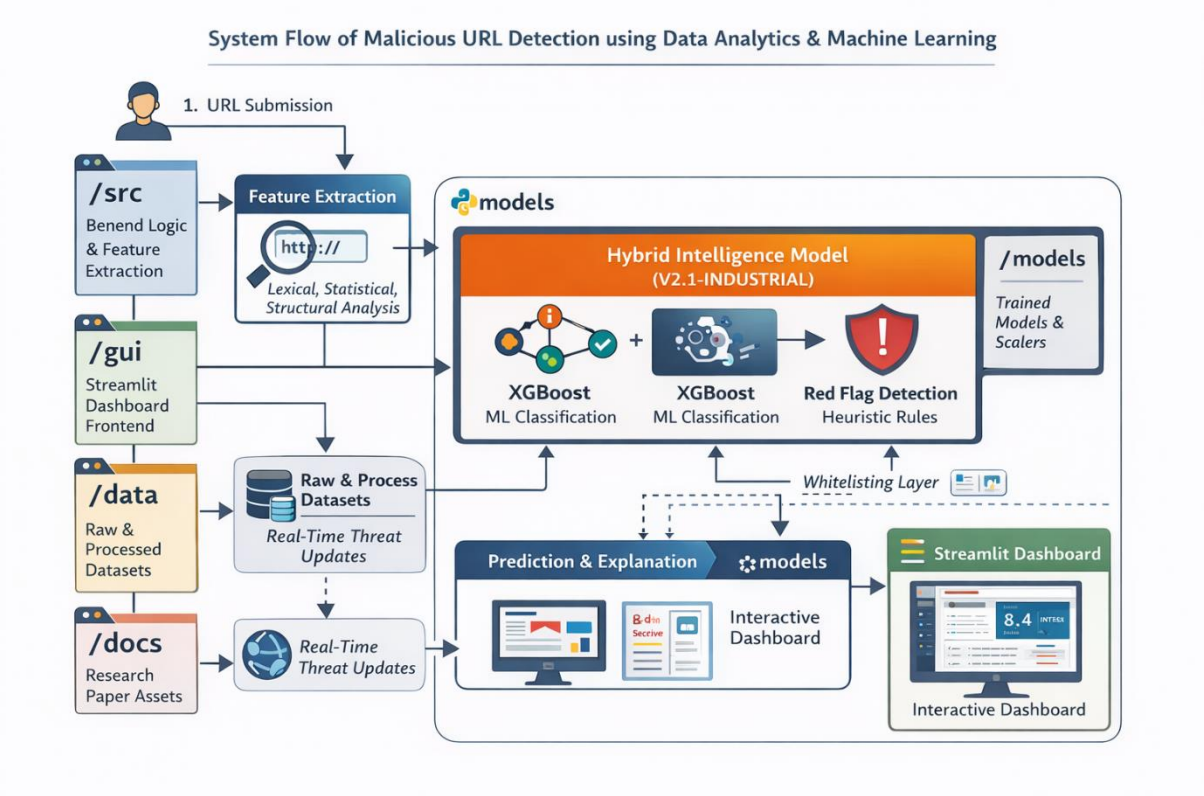

Fig:1.1: (System Flow Diagram of AI Powered Food Nutrition Analyzer Using Image Recognition)

IARJSET ISSN (O) 2393-8021, ISSN (P) 2394-1588

International Advanced Research Journal in Science, Engineering and Technology

Impact Factor 8.311 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 13, Issue 3, March 2026

DOI: 10.17148/IARJSET.2026.13351

V. RESULTS AND DISCUSSION

The proposed system was tested on a diverse dataset containing phishing, malware, and benign URLs. The XGBoost classifier achieved an average accuracy of approximately 96%, with high precision and recall values.

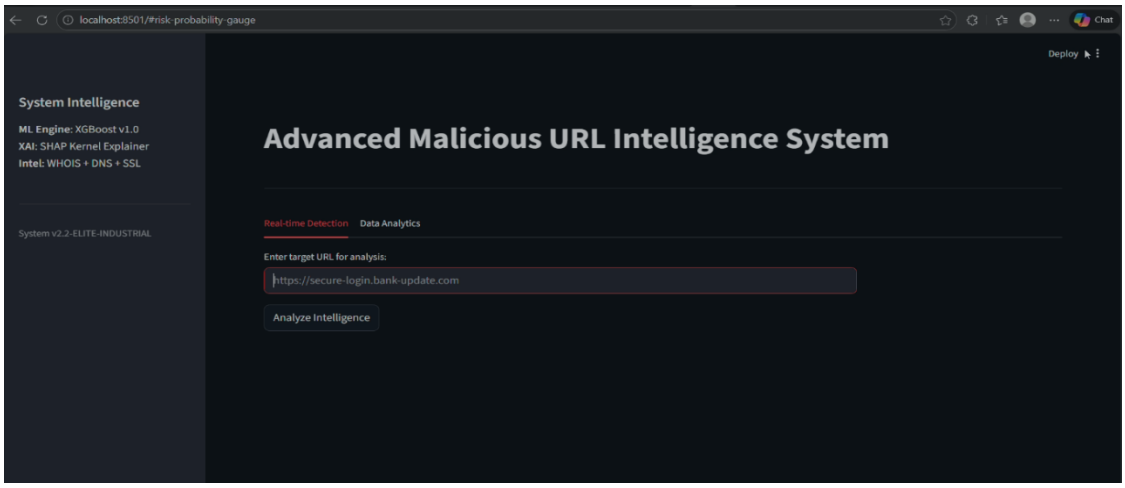

Fig: 1.2: (Malicious URL Intelligence Dashboard Page)

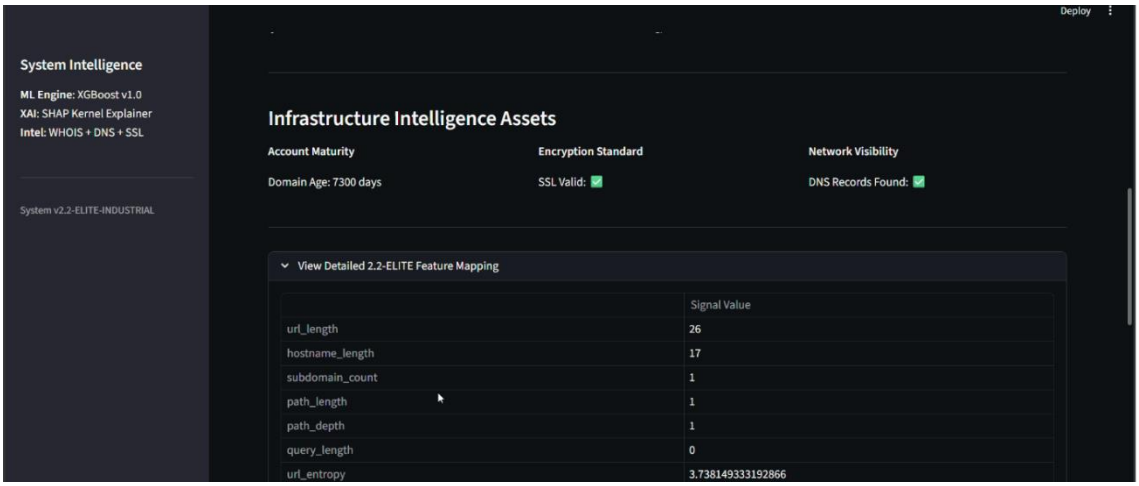

Fig: 1.3: (Data Analysis Page)

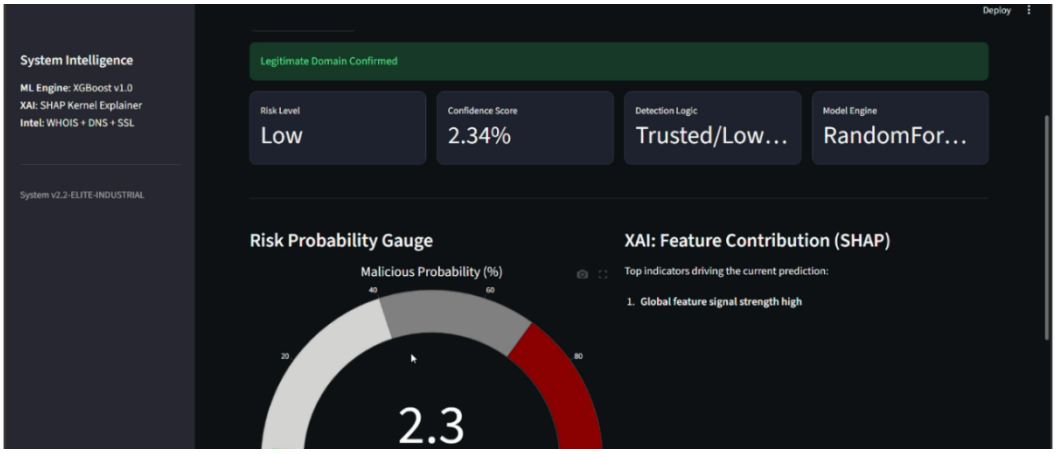

Fig: 1.4: (Infrastructure Intelligence Assets Page)

IARJSET ISSN (O) 2393-8021, ISSN (P) 2394-1588

International Advanced Research Journal in Science, Engineering and Technology

Impact Factor 8.311 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 13, Issue 3, March 2026

DOI: 10.17148/IARJSET.2026.13351

The hybrid detection layer improved detection reliability for stealthy phishing URLs that attempt to mimic legitimate domains. The integration of heuristic rules reduced false negatives significantly.

SHAP-based explainability enabled clear visualization of influential features such as entropy score, suspicious keywords, and domain depth. This transparency enhances trust in automated cybersecurity systems.

The system demonstrated strong generalization capability against unseen URLs, indicating robustness and practical applicability in real-world environments.

VI. CONCLUSION

The Detecting Malicious URLs using Data Analytics and Machine Learning system demonstrates that hybrid intelligence models can significantly enhance cybersecurity defenses. By combining XGBoost classification, heuristic red flag detection, and explainable AI techniques, the system achieves high accuracy and reliability.

The solution overcomes limitations of traditional blacklist-based approaches and provides scalable, transparent, and efficient malicious URL detection. It offers practical value for institutions, enterprises, and cybersecurity researchers.

VII. FUTURE WORK

A modern cybersecurity framework can be envisioned as a cohesive system that integrates several advanced capabilities. It begins with **real-time browser extensions**, which provide immediate protection at the user's point of interaction with the web. This is supported by a **cloud-based microservices architecture**, ensuring scalability, resilience, and rapid deployment of security updates. At its core, the system leverages **deep learning-based URL embedding techniques** to detect malicious patterns with high accuracy, while **continuous learning from threat intelligence feeds** keeps defenses adaptive against evolving threats. To enhance enterprise visibility, the framework seamlessly connects with **SIEM (Security Information and Event Management) systems**, enabling centralized monitoring and incident response. Finally, the model incorporates **adversarial attack resistance enhancements**, strengthening its robustness against attempts to bypass detection mechanisms. Together, these elements form a layered, intelligent, and future-ready defense strategy.

REFERENCES

[1]. J. Ma et al., "Beyond Blacklists: Learning to Detect Malicious Web Sites from Suspicious URLs," ACM SIGKDD, 2009.
[2]. D. Sahoo, C. Liu, and S. Hoi, "Malicious URL Detection using Machine Learning," IEEE Access, 2017.
[3]. A. Singh and R. Kumar, "Deep Learning-Based Phishing URL Detection," International Journal of Computer Applications, 2020.
[4]. T. Chen and C. Guestrin, "XGBoost: A Scalable Tree Boosting System," ACM SIGKDD, 2016.
[5]. S. Lundberg and S.-I. Lee, "A Unified Approach to Interpreting Model Predictions," NeurIPS, 2017.
[6]. F. Pedregosa et al., "Scikit-Learn: Machine Learning in Python," Journal of Machine Learning Research, 2011.
[7]. Python Software Foundation, "Python Language Reference," 2025.
[8]. Streamlit, "Streamlit Documentation," 2026.
[9]. PostgreSQL Global Development Group, "PostgreSQL Documentation," 2026.
[10]. OWASP Foundation, "OWASP Phishing Prevention Guidelines," 2025.

