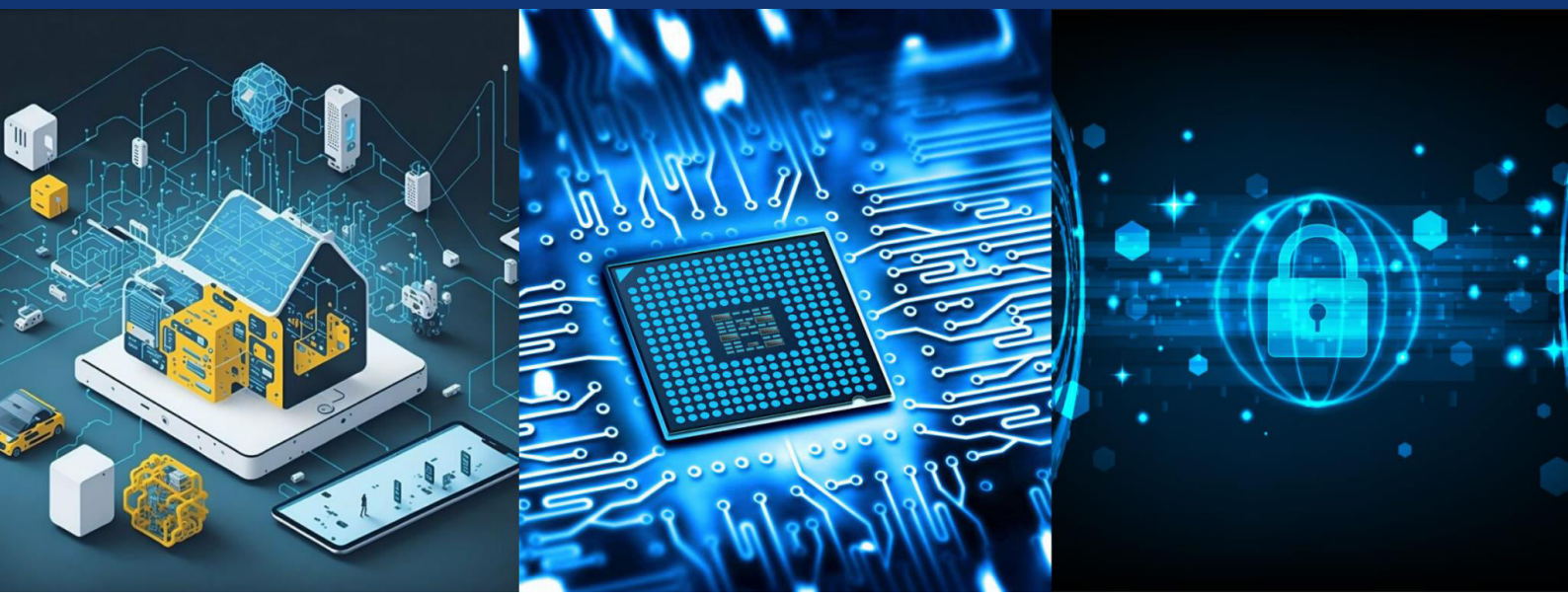
IJIRCCE

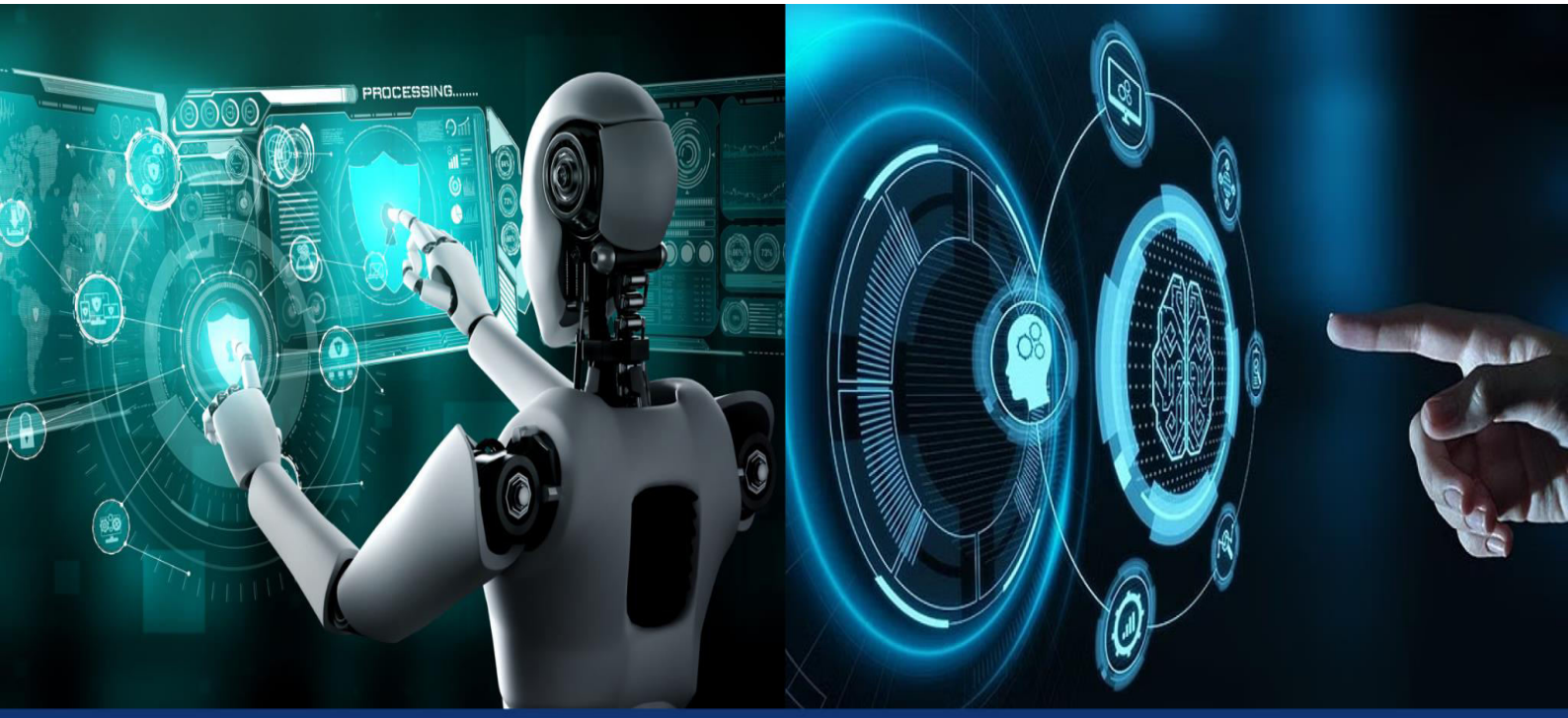
ISSN(O): 2320-9801

ISSN(P): 2320-9798



International Journal of Innovative Research in Computer and Communication Engineering

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)



Impact Factor: 9.927

Volume 14, Issue 7, July 2026

© 2026 IJIRCCE | Volume 14, Issue 7, July 2026 DOI: 10.15680/IJIRCCE.2026.1407027

www.ijircce.com | e-ISSN: 2320-9801, p-ISSN: 2320-9798| Impact Factor: 9.927| ESTD Year: 2013|

International Journal of Innovative Research in Computer and Communication Engineering (IJIRCCE)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

Malicious URL Detection Using Machine Learning

Dr. T.V.S Sriram[1], Mr. Suresh Gopi[2], K. Ashok[3]

Sr. Asst. Professor, Dept. of MCA, NSRIT, Visakhapatnam, AP, India[1]

Asst. Professor, Dept. of MCA, NSRIT, Visakhapatnam, AP, India[2]

Dept. of MCA, NSRIT, Visakhapatnam, AP, India[3]

ABSTRACT: Malicious URLs are among the most common vectors used to deliver phishing, malware, spam, and defacement attacks, affecting millions of internet users worldwide and causing significant financial and data loss. Traditional detection methods such as blacklisting and signature-based filtering are often reactive, unable to detect newly generated malicious URLs, and require frequent manual updates. To address these limitations, this paper presents a Malicious URL Detection framework that utilizes machine learning techniques to analyze URL-related parameters such as lexical features, host-based features, domain age, and page content characteristics. The collected data undergoes preprocessing and feature extraction to improve prediction accuracy. Multiple classification algorithms are employed to identify URL categories including benign, phishing, malware, and defacement URLs. The framework provides automated analysis and reliable predictions, reducing dependence on manual blacklist maintenance. Experimental evaluation demonstrates the effectiveness of the proposed approach in detecting malicious URLs with high accuracy.

KEYWORDS: Malicious URL Detection, Artificial Intelligence, Machine Learning, Cybersecurity Analytics, Phishing Detection, Predictive Modeling, Feature Extraction, URL Classification.

I. INTRODUCTION

The Uniform Resource Locator (URL) is the primary means by which users access web resources, and it has become a favored attack vector for cybercriminals. Malicious URLs such as those used for phishing, malware distribution, and website defacement affect millions of internet users worldwide, leading to financial loss, data breaches, and compromised systems.

Traditional detection techniques like blacklisting and heuristic rule-based filtering provide reasonable results but require constant updates, cannot generalize to unseen threats, and often lag behind rapidly evolving attack patterns. Due to these limitations, many newly created malicious URLs remain undetected during their early stages of distribution.

Recent developments in Artificial Intelligence (AI) and Machine Learning (ML) have enabled the development of intelligent cybersecurity systems capable of analyzing URL-related data efficiently. These technologies can identify hidden patterns in lexical structure, domain properties, and network behavior to support early detection of malicious URLs.

This paper proposes a Malicious URL Detection framework that analyzes URL parameters and predicts URL categories using machine learning algorithms. The system aims to provide accurate, fast, and scalable malicious URL identification to strengthen web security.

II. LITERATURE REVIEW

Several studies have explored the use of Artificial Intelligence and Machine Learning techniques for malicious URL detection. Traditional detection methods such as blacklisting provide reasonable results but are reactive and require constant maintenance.

Researchers have applied machine learning algorithms including Support Vector Machine (SVM), Decision Tree, and Random Forest to analyze lexical and host-based URL features and classify URLs as benign or malicious. These approaches have shown promising results in improving detection accuracy and reducing response time.

www.ijircce.com | e-ISSN: 2320-9801, p-ISSN: 2320-9798| Impact Factor: 9.927| ESTD Year: 2013|

International Journal of Innovative Research in Computer and Communication Engineering (IJIRCCE)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

Recent advancements in deep learning models such as Convolutional Neural Networks (CNN) and Long Short-Term Memory (LSTM) networks have further enhanced character-level and sequence-based URL pattern analysis. These models can automatically identify complex relationships within URL strings and support early detection without relying on handcrafted features.

Based on the findings of previous studies, the proposed framework utilizes AI-based techniques to provide an efficient, accurate, and scalable system for malicious URL identification and monitoring.

III. SYSTEM OVERVIEW

The proposed system is an AI-driven cybersecurity solution developed to identify malicious URLs using machine learning techniques. The system analyzes URL-related parameters such as lexical features, host-based features, domain registration information, and page content characteristics to detect abnormalities indicative of malicious behavior.

The framework consists of four major stages: data collection, data preprocessing, model training, and prediction. After processing the input data, machine learning algorithms classify each URL into categories such as Benign, Phishing, Malware, or Defacement.

The system is designed to provide accurate, fast, and scalable predictions, enabling early detection of malicious URLs and supporting proactive web security monitoring.

IV. METHODOLOGY

The development of the proposed system follows a structured approach to ensure accurate malicious URL identification. The methodology consists of the following phases.

4.1 Data Collection

URL-related parameters such as lexical features (URL length, special character count, subdomain count), host-based features (domain age, WHOIS information, IP address), and content-based features are collected from labeled URL datasets for analysis.

4.2 Data Preprocessing

The collected data is cleaned by handling missing values, removing inconsistencies, and normalizing features to improve data quality and model performance.

4.3 Feature Extraction

Important attributes influencing URL maliciousness, such as URL length, presence of IP addresses, use of URL shortening services, special character frequency, and domain age, are identified and selected to enhance the efficiency and accuracy of the prediction model.

4.4 Model Training

Machine learning algorithms such as KNN, SVM, Random Forest, Decision Tree, and Neural Networks are trained using the processed dataset to learn patterns associated with malicious URLs.

4.5 Prediction and Classification

The trained model analyzes input URL data and classifies each URL into categories such as Benign, Phishing, Malware, or Defacement.

4.6 Performance Evaluation

The system performance is evaluated using metrics such as Accuracy, Precision, Recall, and F1-Score to determine the effectiveness of the proposed framework.

www.ijircce.com | e-ISSN: 2320-9801, p-ISSN: 2320-9798| Impact Factor: 9.927| ESTD Year: 2013|

International Journal of Innovative Research in Computer and Communication Engineering (IJIRCCE)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

V. SYSTEM ARCHITECTURE

The architecture of the proposed system follows a layered design that enables efficient processing and classification of URL-related data.

The system consists of six major layers: User Input Layer, Data Preprocessing Layer, Feature Extraction Layer, Machine Learning Layer, Prediction and Analysis Layer, and Reporting Layer.

The User Input Layer collects URL data submitted by users or crawled from web sources. The Data Preprocessing Layer improves data quality through cleaning and normalization techniques. The Feature Extraction Layer identifies the most relevant lexical and host-based attributes. The Machine Learning Layer applies classification algorithms to learn URL patterns and predict threat categories. The Prediction and Analysis Layer generates classification results along with confidence scores. Finally, the Reporting Layer presents the prediction results and threat insights in a user-friendly format.

VI. MODULES

User Module: In this module

- User can register and login into the system
- User can submit URLs for analysis
- User can view prediction results
- User receives alerts and recommendations for suspicious URLs

Feature Selection Module: In this module:

- Data collection from URL datasets and live submissions
- Feature extraction and processing (URL length, special characters, domain age)
- Selection of important features affecting URL maliciousness

VII. IMPLEMENTATION DETAILS

7.1 Technology Stack

The proposed system is developed using Python as the programming language and Django as the web framework. Machine learning models are implemented using Scikit-learn, while data processing is performed using NumPy and Pandas. SQLite is used for database management.

7.2 Frontend Implementation

The user interface is designed to provide a simple and interactive environment where users can submit URLs and view prediction results easily.

7.3 Backend Implementation

The backend manages data processing, model execution, prediction generation, and communication between the user interface and database.

7.4 Database Design

The database stores user information, submitted URL data, prediction results, and generated reports. SQLite is used to ensure efficient data storage and retrieval.

7.5 Model Implementation

Machine learning algorithms such as KNN, SVM, Decision Tree, Random Forest, and Neural Networks are trained using the processed dataset. The best-performing model is selected for malicious URL prediction.

7.6 Security Measures

Input validation, secure data handling, and controlled database access mechanisms are implemented to ensure data integrity and system reliability.

© 2026 IJIRCCE | Volume 14, Issue 7, July 2026 DOI: 10.15680/IJIRCCE.2026.1407027

www.ijircce.com | e-ISSN: 2320-9801, p-ISSN: 2320-9798| Impact Factor: 9.927| ESTD Year: 2013|

International Journal of Innovative Research in Computer and Communication Engineering (IJIRCCE)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

VIII. EXPERIMENTAL DETAILS AND RESULTS

8.1 Experimental Setup

The proposed system was implemented and tested using Python, Django, Scikit-learn, NumPy, and Pandas. The experiments were conducted on a system with Intel Core i5 processor, 8GB RAM, and Windows operating system. URL-related parameters such as lexical features, host-based features, domain age, and content characteristics were used for model training and evaluation.

8.2 Test Results

Several machine learning models were evaluated for malicious URL classification. The performance of the models was measured using standard classification metrics.

Test Case	Description	Result	Status
User Authentication	Login / Registration with valid and invalid credentials	Passed	✓
URL Data Input	Enter URL samples for lexical and host-based feature extraction	Passed	✓
Data Preprocessing	Cleaning, normalization and feature processing	Passed	✓
ML Model Execution	Training and testing machine learning models	Passed	✓
URL Classification	Classifying Benign, Phishing, Malware, and Defacement URLs	Passed	✓
Report Generation	Generate prediction results and threat insights	Passed	✓
Database Management	Store user details, URL records and prediction history	Passed	✓
Response Time	Prediction generated within acceptable time	Passed	✓
Dashboard Visualization	Display URL analysis graphs and results	Partial	~
Mobile Responsiveness	System display on multiple devices	Partial	~

Test Case Results for the Malicious URL Detection System

8.3 Performance Evaluation

The system performance was evaluated using Accuracy, Precision, Recall, and F1-Score metrics. Experimental results indicate that the proposed framework effectively identifies malicious URLs with high reliability and minimal prediction delay.

8.4 Observations

The proposed system successfully classified URLs such as Benign, Phishing, Malware, and Defacement categories. The use of machine learning algorithms improved detection accuracy and reduced the complexity associated with traditional blacklist-based approaches.

IX. APPLICATIONS

Web Security Monitoring:
The system helps security teams monitor incoming URL traffic and identify possible malicious links at an early stage.
Early Threat Detection and Prevention:
AI-based prediction helps in detecting threats such as phishing, malware distribution, and website defacement before they cause harm.
Browser and Email Protection:
Users can be warned in real time when clicking links in browsers or emails through intelligent URL analysis.

www.ijircce.com | e-ISSN: 2320-9801, p-ISSN: 2320-9798| Impact Factor: 9.927| ESTD Year: 2013|

International Journal of Innovative Research in Computer and Communication Engineering (IJIRCCE)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

Integration with Security Tools:
The system can be integrated with firewalls, proxy servers, and endpoint protection platforms for real-time URL filtering.
Enterprise Network Protection:
It enables organizations to filter malicious traffic and protect employees from web-based attacks.
Research and Threat Intelligence:
Collected URL data can assist researchers in studying evolving attack patterns and improving detection methods.
Personalized Security Alerts:
The system can provide customized warnings based on individual browsing behavior and risk levels.

X. LIMITATIONS

Data Dependency:
The accuracy of the system depends on the quality, quantity, and reliability of the collected URL data.
Limited Dataset Availability:
Insufficient or imbalanced datasets may affect the performance and generalization ability of machine learning models.
Evasion Techniques:
Attackers may use obfuscation and URL manipulation techniques to bypass detection models.
Evolving Threat Landscape:
New attack patterns emerge continuously, which can make accurate classification of previously unseen URLs challenging.
Need for Continuous Retraining:
Effective prediction requires regular model retraining with updated threat data.
Clinical—Operational Validation Requirement:
AI-based predictions should be further validated by security analysts before final decisions in high-stakes deployments.
Privacy and Security Concerns:
URL and browsing-related data require secure storage and protection from unauthorized access.

XI. FUTURE ENHANCEMENT

The future scope of the proposed system focuses on improving the accuracy, reliability, and usability of the malicious URL detection framework. Advanced deep learning techniques such as CNN, LSTM, and transformer-based models can be integrated to enhance character-level and sequence-based prediction performance. The system can be further developed with real-time monitoring capabilities by integrating with browser extensions and network gateways. Using larger and more diverse threat intelligence datasets can improve the generalization of the model. Future enhancements may also include personalized security recommendations, mobile application support, and integration with enterprise security platforms to assist analysts in better threat monitoring. Additionally, advanced adversarial-robustness mechanisms can be implemented to reduce susceptibility to evasion attacks.

XII. CONCLUSION

The proposed Malicious URL Detection framework provides an effective AI-based approach for identifying malicious URLs using machine learning techniques. The system analyzes URL-related parameters and applies data preprocessing, feature extraction, and classification methods to detect categories such as benign, phishing, malware, and defacement URLs. Compared with traditional blacklist-based approaches, the proposed system offers a faster, more adaptive, and scalable solution for web security. The experimental results demonstrate that machine learning models can successfully identify malicious URL patterns and support early threat detection. The framework can assist security teams and individuals in better understanding URL-based threats and improving overall cybersecurity posture. With further improvements in real-time monitoring, advanced AI models, and security platform integration, the proposed system can become a reliable tool for future intelligent web security systems.

REFERENCES

[1] Symantec Corporation, Internet Security Threat Report, Symantec, 2019.
[2] Anti-Phishing Working Group (APWG), Phishing Activity Trends Report, APWG, 2023.

www.ijircce.com | e-ISSN: 2320-9801, p-ISSN: 2320-9798| Impact Factor: 9.927| ESTD Year: 2013|

International Journal of Innovative Research in Computer and Communication Engineering (IJIRCCE)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

[3] I. Goodfellow, Y. Bengio, and A. Courville, Deep Learning, MIT Press, 2016.
[4] T. Hastie, R. Tibshirani, and J. Friedman, The Elements of Statistical Learning, Springer, 2017.
[5] C. M. Bishop, Pattern Recognition and Machine Learning, Springer, 2006.
[6] F. Chollet, Deep Learning with Python, 2nd ed., Manning Publications, 2021.
[7] J. Ma, L. K. Saul, S. Savage, and G. M. Voelker, "Beyond Blacklists: Learning to Detect Malicious Web Sites from Suspicious URLs," ACM SIGKDD, 2009.
[8] D. Sahoo, C. Liu, and S. C. H. Hoi, "Malicious URL Detection using Machine Learning: A Survey," arXiv preprint, 2019.
[9] A. Esteva et al., "A Guide to Deep Learning in Healthcare," Nature Medicine, vol. 25, no. 1, pp. 24-29, 2019.
[10] I. Sommerville, Software Engineering, 10th ed., Pearson Education, 2016.
[11] R. S. Pressman and B. R. Maxim, Software Engineering: A Practitioner's Approach, 8th ed., McGraw-Hill Education, 2015.
[12] A. Geron, Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow, 3rd ed., O'Reilly Media, 2022.
[13] J. Han, M. Kamber, and J. Pei, Data Mining: Concepts and Techniques, 3rd ed., Morgan Kaufmann, 2012.
[14] F. Pedregosa et al., "Scikit-learn: Machine Learning in Python," Journal of Machine Learning Research, vol. 12, pp. 2825-2830, 2011.
[15] W. McKinney, Python for Data Analysis, 3rd ed., O'Reilly Media, 2022.
[16] T. Mitchell, Machine Learning, McGraw-Hill, 1997.
[17] E. Alpaydin, Introduction to Machine Learning, 4th ed., MIT Press, 2020.
[18] K. P. Murphy, Machine Learning: A Probabilistic Perspective, MIT Press, 2012.
[19] S. Raschka and V. Mirjalili, Python Machine Learning, 3rd ed., Packt Publishing, 2019.
[20] R. Verma and A. Das, "What's in a URL: Fast Feature Extraction and Malicious URL Detection," ACM IWSPA, 2017.
[21] Y. LeCun, Y. Bengio, and G. Hinton, "Deep Learning," Nature, vol. 521, no. 7553, pp. 436-444, 2015.
[22] G. Hinton and R. Salakhutdinov, "Reducing the Dimensionality of Data with Neural Networks," Science, vol. 313, no. 5786, pp. 504-507, 2006.
[23] A. Krizhevsky, I. Sutskever, and G. Hinton, "ImageNet Classification with Deep Convolutional Neural Networks," NIPS, 2012.
[24] S. Hochreiter and J. Schmidhuber, "Long Short-Term Memory," Neural Computation, vol. 9, no. 8, pp. 1735-1780, 1997.
[25] L. Breiman, "Random Forests," Machine Learning, vol. 45, no. 1, pp. 5-32, 2001.
[26] C. Cortes and V. Vapnik, "Support Vector Networks," Machine Learning, vol. 20, pp. 273-297, 1995.
[27] J. R. Quinlan, C4.5: Programs for Machine Learning, Morgan Kaufmann, 1993.
[28] T. Cover and P. Hart, "Nearest Neighbor Pattern Classification," IEEE Transactions on Information Theory, vol. 13, no. 1, pp. 21-27, 1967.
[29] F. Chollet, Deep Learning with Python, Manning Publications, 2018.
[30] B. Ripley, Pattern Recognition and Neural Networks, Cambridge University Press, 2007.
[31] D. Silver et al., "Mastering the Game of Go with Deep Neural Networks and Tree Search," Nature, vol. 529, pp. 484-489, 2016.
[32] R. Collobert and J. Weston, "A Unified Architecture for Natural Language Processing," ICML, 2008.
[33] J. Brownlee, Machine Learning Mastery with Python, Machine Learning Mastery, 2016.
[34] M. Kuhn and K. Johnson, Applied Predictive Modeling, Springer, 2013.
[35] J. Han and M. Kamber, Data Mining: Concepts and Techniques, Morgan Kaufmann, 2011.
[36] S. Haykin, Neural Networks and Learning Machines, 3rd ed., Pearson, 2009.
[37] N. Chou, R. Ledesma, Y. Teraguchi, and J. C. Mitchell, "Client-Side Defense Against Web-Based Identity Theft," NDSS, 2004.
[38] S. Marchal, J. Francois, 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.
[39] Google Safe Browsing, Transparency Report, Google, 2023.
[40] OpenPhish, Phishing Intelligence Feed Documentation, OpenPhish, 2023.

INTERNATIONAL STANDARD SERIAL NUMBER INDIA

INNO SPACE SJIF Scientific Journal Impact Factor

doi crossref

निस्केयर NISCAIR

INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH

IN COMPUTER & COMMUNICATION ENGINEERING

9940 572 462 6381 907 438 ijircce@gmail.com

www.ijircce.com

Scan to save the contact details