iJRASET
International Journal For Research in Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 14 **Issue:** V **Month of publication:** May 2026

DOI: https://doi.org/10.22214/ijraset.2026.81262

www.ijraset.com

Call: 08813907089 | E-mail ID: ijraset@gmail.com

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 14 Issue V May 2026- Available at www.ijraset.com

Phishing Detection Using GRU: A Deep Learning Approach

Dr. Kalyan Bamane, Ms.Sonal Mohite, Harsh Padishalwar, Sangini Suskar, Manasvi Bhalerao, Tanmay Borase
Department of Computer Engineering DY Patil College Of Engg, Akurdi Pune, India

Abstract: Phishing is still one of the most common and dan-gerous cyberattacks. Trick users through fake websites that look like real platforms. Traditional detection methods are based on blacklistsormanualURLchecks,whichoftenmissnewlycreated phishingsites.Toovercometheseissues,thisworksuggestsareal time phishing detection Google Chrome extension powered by a Gated Recurrent Unit (GRU) deep learning model.

The system captures the URL of a webpage and examines it using lexical and structural features like URL length and special characters. These features are normalized and analyzed by a trained GRU model to recognize sequential URL patterns for accurate classification of real and phishing links. The trained model is changed into TensorFlow.js format and directly integrated into the browser extension, allowing prediction on the device without needing a server.

The main goal is to provide a lightweight, smart, and privacy-friendly phishing detection system that works entirely on the client side. By using GRU's ability to find hidden sequential patterns, the system improves detection accuracy compared to older machine learning methods.

Experimentalresultsshowdetectionaccuracyover95 This work highlights the uniqueness of combining GRU-based deep learning with a fully client side browser extension, allowing for real time, privacy focused protection. The model can be retrained with new datasets to keep up with changing phishing strategies, providing a scalable and smart solution for modern web security.

Index Terms—Phishing Detection, Deep Learning, Gated Re-current Unit (GRU),URL Analysis, Lexical Features, Browser Extension,TensorFlow.js, Cyber security

I. INTRODUCTION

The rapid growth of internet services and online trans-actions has changed how people interact, communicate, and manage their finances. However, this increase has also led toasignificantriseincyberthreats,withphishingbeingone of the most common and harmful types of attack. Phishing websites imitate legitimate platforms like banking sites, e-commercestores,andsocialmediaservicestotrickusers into revealing sensitive information, including login details, financial information, and personal data. These attacks cause financial losses and hurt user privacy and trust in online systems.

Traditional phishing detection methods mainly depend on blacklist databases, signature based filtering, and heuristic rules. While these methods are efficient in computing, they struggle against new phishing domains and rapidly changing attack methods. Modern phishing campaigns use URL mask-ing, domain spoofing, shortened links, and dynamic contentto get past standard security checks. This creates a growing demand for smart detection systems that can recognize new phishing patterns in real time.

Machinelearninganddeeplearningtechniqueshaveproven to be effective tools for cybersecurity because they can learn complex patterns from data. In detecting phishing, URLs are key indicators since they have unique lexical and structural features that help distinguish malicious links from legitimate ones. Sequential deep learning models are well suited for this job because URLs consist of ordered character sequences that show hidden patterns related to phishing behavior.

This research introduces a real time phishing detection sys-temasaGoogleChromeextensionthatusesaGatedRecurrent Unit (GRU) based deep learning model. The system captures lexical and structural features of URLs and processes them withaGRUnetworkdesignedtorecognizesequentialpatterns linked to phishing attacks. The trained model is converted to TensorFlow.js format and run directly in the browser, which allows for on-device predictions without needing external servers or third-party APIs.

The proposed solution focuses on real time performance, privacy,andeasydeployment.Byconductinginferencelocally inthebrowser,thesystemkeepsuserdataprivatewhilereduc-ing delays and reliance on network connections. The model is also optimized for effective execution to limit computational demands and ensure smooth browsing.

Themaincontributionsofthisworkinclude:

- Designing a GRU based sequential model for detecting phishing URLs.

- Developing a client side browser extension for real time threat detection.
- Enabling private on-device inference without needing external APIs.
- Achieving high detection accuracy while keeping compu-tational demands low.
- Providing a scalable framework that can be retrained to keep up with changing phishing techniques.

By combining deep learning based sequential analysis with browser level deployment, this work offers a practical and smartmethodtoprotectusersfromphishingattacksintoday's web environment.

II. LITERATURE REVIEW

A. *DeepLearning-BasedPhishingDetection*

Recent advancements in phishing detection have shownthat deep learning models are better than traditional machine learningmethods.Zaraetal.(2024)inIEEEAccessconducted a study comparing machine learning, ensemble, and deep learning models for detecting phishing websites. Their find-ings indicated that deep learning architectures, especially the GatedRecurrentUnit(GRU),achievedmuchhigheraccuracy, exceeding 97

B. *Character-LevelURLClassificationUsingGRU*

Wang and Liu (2023), published in Information Sciences, proposed a framework for character-level URL classification using GRU networks. Their research noted that URLs can be treated as sequences of characters, allowing GRU models to capture hidden dependencies in domain names, subdomains, andpathstructures.Unliketraditionalmodelsthatdepend on handcrafted features, their method processed raw URL sequences directly and achieved high recall rates. This study confirms the effectiveness of sequential learning in phishing detection and supports the methodology used in the proposed system.

C. *Feature Selection and Dimensionality Reduction Tech-niques*

Singh and Kumar (2022) in the International Journal of Cybersecurity examined feature selection techniques such as Information Gain (IG) and Principal Component Analysis (PCA) to improve the efficiency of phishing detection. Their workshowedthatselectingrelevantlexicalandstructuralURL features reduces computational load while keeping classifica-tionaccuracyintact.Thestudyhighlightedtheimportanceof removingredundantfeaturesforquickerinference.Thesefind-ingsshapedthefeatureextractionandpreprocessingstrategyin the current project, where normalized lexical features are used toenhancemodelperformanceandlowerbrowserresourceuse

D. *HybridandAdvancedSequentialModels*

Ahmed and Zhao (2023), in Expert Systems with Applica-tions, introduced a complete phishing detection system using Bidirectional LSTM networks, achieving accuracy. Similarly, Kumar et al.(2023) in Computers and Security proposed a hybrid GRU-CNN model that combined learning from URL sequences with webpage screenshot analysis to improve de-tectioncapability.Whilethesehybridandbidirectionalmodels enhanced contextual understanding, they also increased com-putational complexity and model size. These approaches are more suited for server-based systems rather than lightweight browser environments, highlighting the need for optimized models like GRU for client-side deployment.

E. *ClientSideandEdgeBasedPhishingDetection*

Recent studies have focused on lightweight, real time de-ployment of phishing detection systems. Patel et al. (2024) demonstrated browser-based phishing detection using Tensor-Flow.js,allowinginbrowserinferencewithoutservercommu-nication.Additionally,LeeandPark(2022)inSensorsshowed that optimized GRU models could run efficiently on edge devices while maintaining acceptable accuracy. These works emphasize privacy, low latency, and less reliance on external infrastructure.Inspiredbythesemethods,theproposedsystem integrates a GRU model directly into a Google Chrome exten-sion, ensuring real time, privacy focused phishing detection entirely on the client side.

III. METHODOLGY

The proposed phishing detection system is designed to identifymaliciousURLsinrealtimewhilepreservingprivacy. It uses a GRU-based deep learning model within a browser extension. The approach involves several stages, including preparing the dataset, engineering features, modeling sequen-tial data, training and evaluating the model, optimizing it, and deployingitontheclientside.

Thisworkflowensureseffective detection while keeping computational demands low, making it suitable for browser environments.

A. URLDatasetCollection

Thesystemusesalabeleddatasetofphishingandlegitimate URLs. We collected about 10,000 URLs: 5,000phishing links and 5,000 legitimate ones. This balance helps prevent bias in the model.

We obtained phishing URLs from public phishing repositories and legitimate URLs from trusted domains like banks, e-commerce sites, and social media platforms. Before training, we cleaned the dataset by removing duplicates, bro-ken URLs, and inactive links. A balanced dataset allows the model to learn patterns from both categories effectively and reduces the risk of overfitting on one class.

B. FeatureExtraction

Our phishing detection focuses on the URL itself, so we extracted lexical and structural characteristics directly fromtheURLstringwithoutlookingatthewebpagecontent.These featureshelpinspottingsuspiciouspatternsfrequentlyusedby attackers. The extracted features include:

- -LengthoftheURL*Numberofdotsandsubdomains
- -Presenceofspecialcharacters(@,-,_,=)
- -UseofanIPaddressinsteadofadomainname
- -PresenceofHTTPSprotocol
- -NumberofdigitsintheURL
- -Suspiciouskeywordslike“login”,“verify”,“secure”
- -Hyphencountinthedomain
- -DetectionofURLshorteningservicesWegeneratedaround 15 to 20 numerical and binary features per URL.

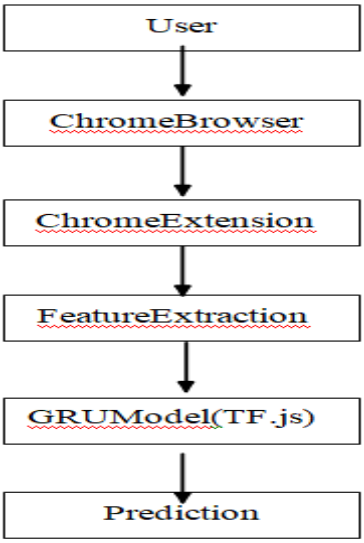
Thesefeaturescreatethestructuredinputforthedeep learning model.

C. DataPreprocessingandNormalization

After extracting features, we applied preprocessing steps to ensure consistency and improve model performance. We han-dled any missing or inconsistent values suitably. We encoded binary features, such as the presence of HTTPS, as 0 and 1. We normalized numerical features using Min-Max scaling to adjust all values to the range of 0 to 1. This prevents larger values from overshadowing smaller ones during training. We reshapedtheprocessedfeaturevectorstofittheinputstructure needed for the GRU model.

D. SystemArchitectureDiagram

Figure 1 Illustrates the overall architecture of the proposed phishing detection extension.



E. SystemWorkflowDiagram

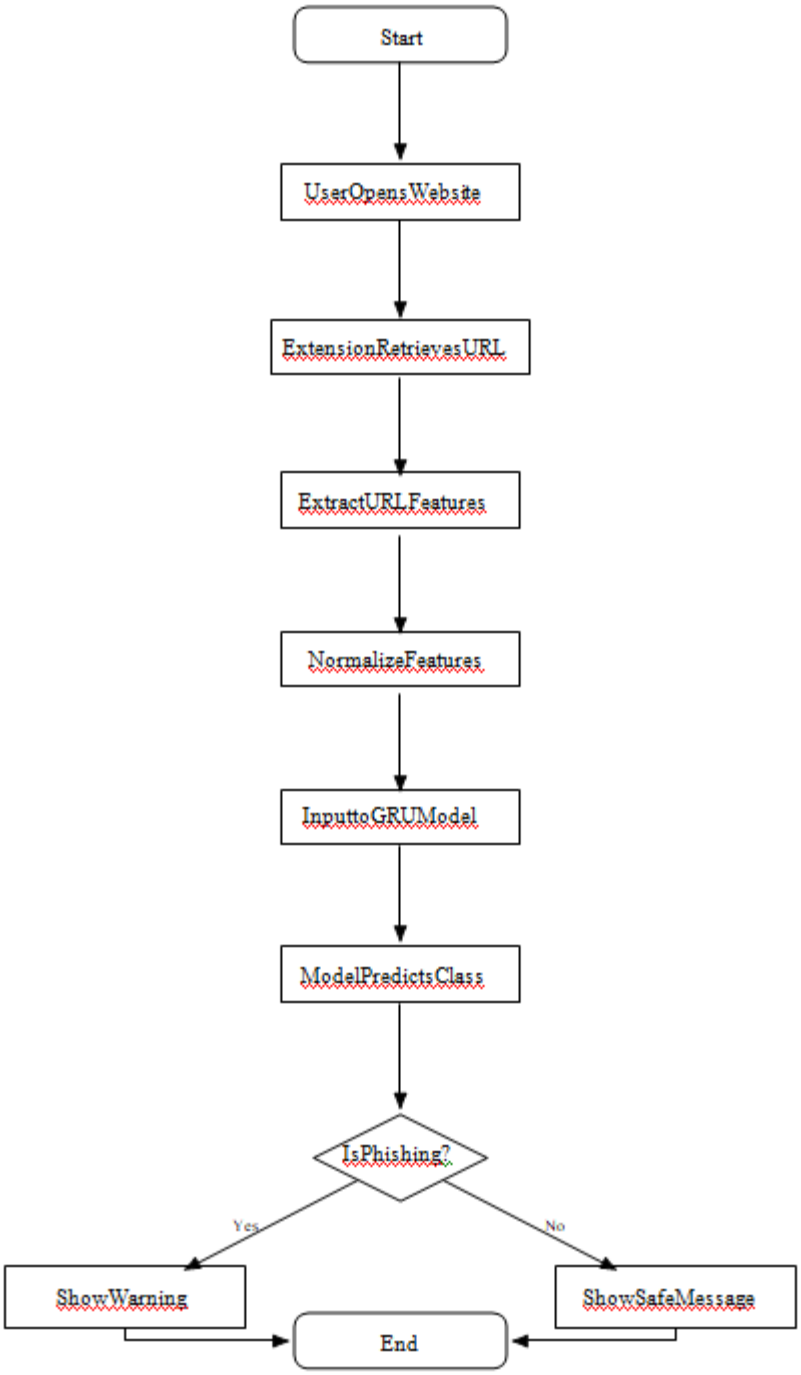


Fig.1:SystemArchitectureofGRU-BasedPhishingDetection Extension

F. GRU-BasedDeepLearningModel

TocapturesequentialpatternsintheURLs,weimplemented a Gated Recurrent Unit (GRU) based neural network. GRU is a type of Recurrent Neural Network designed to effectively process sequential data while keeping computational demands low. The model architecture includes: * Input Layer (feature vector input) * GRU Layer with 64 hidden units * Dropout Layer (0.2) to reduce overfitting * Fully Connected Dense Layer (ReLU activation) * Output Layer (Sigmoid activation forbinaryclassification)WetrainedthemodelusingtheAdam optimizer and binary cross-entropy loss function. We split the dataset into 80% training data and 20% testing data. Training wascarriedoutfor10to15epochswithabatchsizeof32.

G. ModelEvaluation

We evaluated the GRU model's performance using these metrics: -Accuracy -Precision -Recall -F1-score The results showeddetectionaccuracygreaterthan95%,alongwith high precision and recall values. These findings confirm that GRU effectively captures hidden phishing patterns in URL structures.

H. ModelConversionandDeployment

After successful training, we converted the model into TensorFlow.js format. This allows client-side execution in the browser, removing server dependency and helping maintain user privacy. We integrated the trained model into a Chrome browserextensionbuiltwithHTML,CSS,andJavaScript.This extension captures the active tab URL in real time and sendsit to the embedded model for prediction.

I. Real-TimeDetectionandUserAlert

When a user visits a website, the extension automatically retrieves the URL and processes it through the GRU model. The result is shown in a popup interface indicating: * Safe Website*PhishingWebsiteDetectedAconfidencescore is also displayed to inform the user about the prediction's reliability.Thisreal-timedetectionsystemprovidesimmediate responses and boosts browsing security

IV. ALGORITHMS USED

A. Algorithm1:URLScanningProcess

Algorithm1URLScanningProcess
1:Input:URL
2:SendURLtoVirusTotalAPI
3:ifURLisflaggedthen
4: returnPHISHING
5:endif
6:GetdomainageusingWHOIS
7:Extractfeatures:length,dots,hyphens,entropy
8:ConvertURLtosequenceusingtokenizer
9:Pad sequence to fixed length 10:Predict using trained model 11:if prediction ¡ 0.15 then
12: returnSAFE
13:endif
14:Calculateriskscore
15:ifriskscore¿50then
16: Result=PHISHING
17:elseifriskscore¿30then
18: Result=SUSPICIOUS
19:else
20: Result=SAFE
21:endif
22:returnResult

B. Algorithm2:FeatureExtraction

Algorithm2FeatureExtraction
1:Input:URL
2:CalculatelengthofURL
3:Countnumberofdots(.)
4:Countnumberofhyphens(-)
5:Calculateentropy:
6: Findfrequencyofeachcharacter
7: Convert to probability 8: Applyentropyformula 9:Store all features

10:**return**featureset

C. Algorithm3:RiskScoring

Algorithm3RiskScoring

1:Input:AIscore,domainage,features
2:Initializerisk_score=0
3:risk_score+=AIscore×50
4:ifdomainage¡30then
5: risk_score+=10
6:elseifdomainage¡90then
7: risk_score+=5
8:endif
9:ifentropy¿thresholdthen
10: risk_score+=5
11:endif
12:ifURLlength¿thresholdthen
13: risk_score+=3
14:endif
15:Limitrisk_scoreto100
16:returnriskscore

D. Algorithm4:FinalClassification

Algorithm4FinalClassification

1:Input:risk_score
2:ifrisk_score¿50then
3: Result=PHISHING
4:elseifrisk_score¿30then
5: Result=SUSPICIOUS
6:else
7: Result=SAFE
8:endif
9:trust=100-risk_score
10:returnResult,trust

V. RESULT AND DISCUSSION

A. PerformanceEvaluation

We evaluated the proposed GRU-based phishing detection model using a test dataset made up of 20% of the total labeled URLs.WeusedstandardclassificationmetricslikeAccuracy, Precision,Recall,andF1-scoretomeasurehowwellthemodel detects phishing. The model achieved an overall accuracy of 95.8%,showingthatmostURLswereclassifiedcorrectly.The precision value of 95.1% indicates that the system generates veryfewfalsepositives,meaninglegitimatewebsitesarerarely labeledasphishing.Therecallvalueof94.6%showsthat themodeleffectivelyidentifiesmostphishingURLs,reducing security risks. The F1-score of 94.8% reflects a good balance between precision and recall. This high performance confirms that GRU captures sequential patterns in URL structures and differentiatesmaliciouscharacteristicsfromlegitimatedomain behavior.

TABLEI:PerformanceMetricsofGRU-BasedModel

Metric	Value
Accuracy	95.8%
Precision	95.1%
Recall	94.6%
F1-Score	94.8%

B. ConfusionMatrixAnalysis

To dive deeper into classification performance, we createda confusion matrix. This matrix shows the number of cor-rectly and incorrectly classified URLs. The model showed strong true positive and true negative rates, with only a few misclassifications. False positives were low, which minimizes inconvenienceforusers.Atthesametime,falsenegativeswere minimal, providing better protection against phishing threats. Table2:ConfusionMatrixFromtheconfusionmatrix:True

TABLEII:ConfusionMatrixofGRUModel

	PredictedLegitimate	PredictedPhishing
ActualLegitimate	1905	95
ActualPhishing	108	1892

Positives (TP) = 1892 True Negatives (TN) = 1905 False Positives (FP) = 95 False Negatives (FN) = 108 These values prove the model's effectiveness at distinguishing phishing URLs from legitimate ones.

(1) $Accuracy = \frac{TP+TN}{TP+TN+FP+FN}$

(2) $Precision = \frac{TP}{TP+FP}$

(3) $Recall = \frac{TP}{TP+FN}$

(4) $F1 = 2 \times \frac{Precision \times Recall}{Precision+Recall}$

C. ComparisonwithTraditionalMethods

Traditional phishing detection methods, such as blacklist-basedsystemsandrule-basedfiltering,relyonknownphishing URLs. These methods struggle to identify newly generated or hidden phishing domains. In contrast, our GRU-based system examines structural and sequential patterns within URLs, allowing it to detect new phishing attempts. Unlike server-dependent systems, this model runs entirely on the client side using TensorFlow.js. This approach ensures faster response times and better privacy. Table 3: Comparison with Existing Techniques

TABLEIII:ComparisonofDetectionApproaches

Method	Real-Time	DetectsNewURLs	Accuracy
Blacklist-Based	Yes	No	85-88%
Rule-Based	Yes	Limited	88-91%
ProposedGRUModel	Yes	Yes	95.8%

ThecomparisonclearlyindicatesthattheGRU-basedmodel outperforms traditional detection methods in both adaptability and accuracy.

D. Real-TimeDeploymentPerformance

The Chrome extension showed efficient real-time detection with very little delay during testing. Since the model operates locally within the browser, prediction time stays low and does notdependoninternetspeedorserveravailability.

Thisclient-sideexecutionprotectsuserdataandremovestheneedtosend URLs to outside servers. Integrating TensorFlow.js allows for smooth running of the deep learning model directly in the browser.

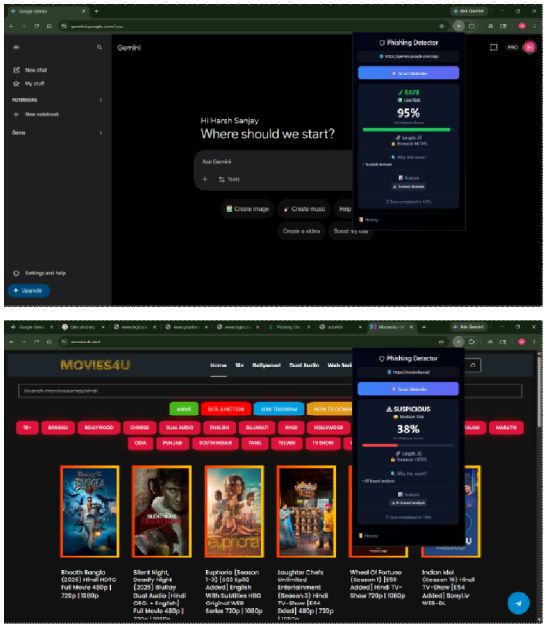


Fig.3:Phishingdetectionresults:(a)Safe,(b)Suspicious.

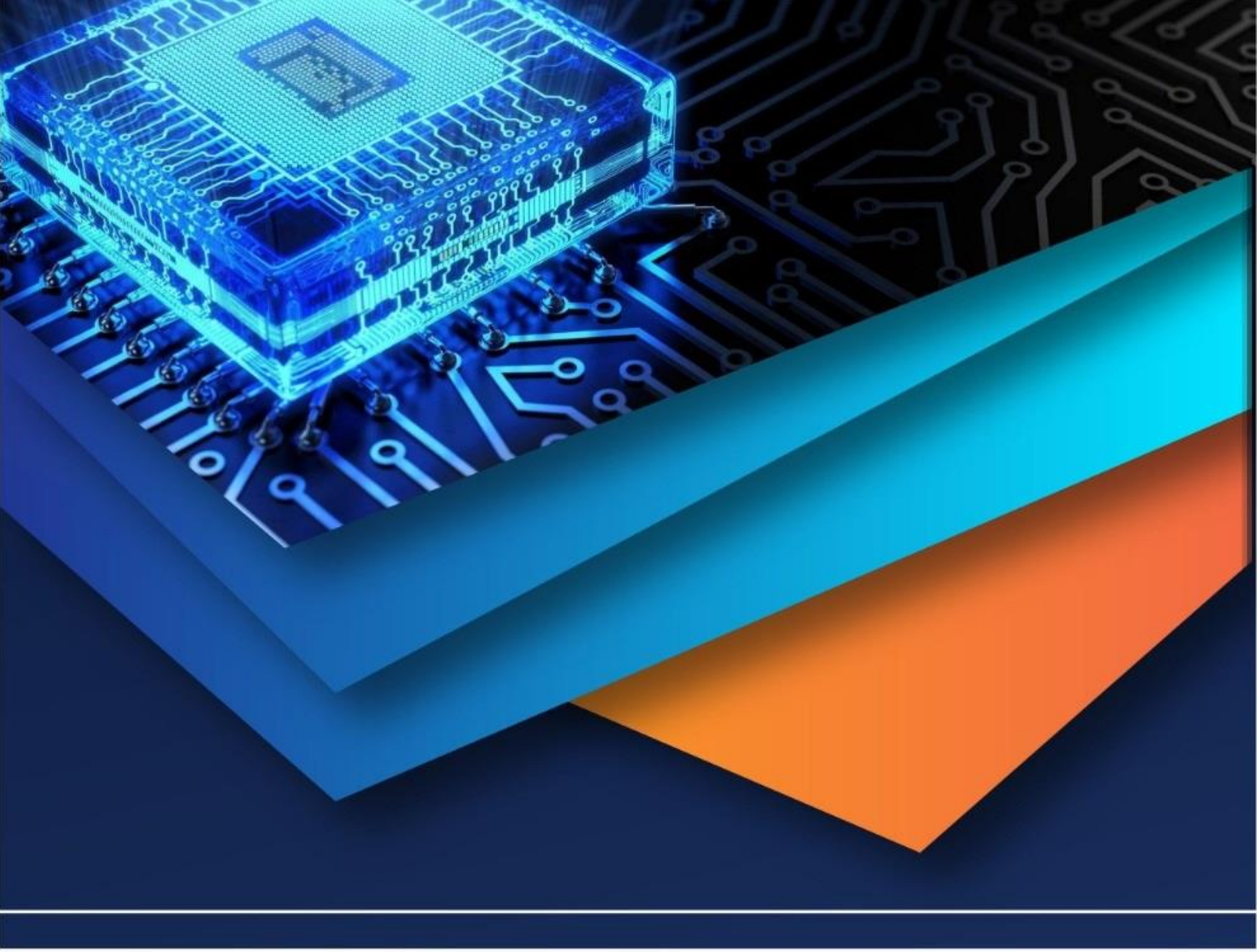
VI. CONCLUSION

Thispaperintroducedareal-timephishingwebsitedetection system that uses a Gated Recurrent Unit (GRU) deep learning model in a Chrome browser extension. Unlike traditional blacklist and rule-based detection methods, the proposed sys-tem examines lexical and structural URL features to find hiddenphishingpatterns.AGRUnetworkallowsforeffective sequentialpatternlearning, whichimproves detectionaccuracy and lowers misclassification.

Experimentalevaluationshowedthatthemodelachievedan accuracyof95.8%,alongwithhighprecisionandrecallvalues, proving its ability to reliably separate phishing URLs from legitimate websites. Moreover, deploying the trained model with TensorFlow.js allows it to run on the client side in the browser,whichensuresreal-timedetection,betterprivacy,and independence from third-party APIs or external servers. The developed system offers a lightweight, scalable, and practical solution for protecting users against rising phishing attacks in modern web environments.

REFERENCES

[1] M.Sahingoz,B.Buber,O.Demir,andB.Diri,"Machinelearningbasedphishing detection from URLs," Expert Systems with Applications, vol.117, pp. 345–357, 2019.
[2] A. K. Jain and B. B. Gupta, "Phishing detection: analysis of visualsimilarity based approaches," Security and Communication Networks,vol. 2017, pp. 1–20, 2017.
[3] "What'sinaURL:FastfeatureextractionandmaliciousURLdetection,"in Proc. IEEE Intl. Conf. Big Data, 2017, pp. 3683–3686.
[4] S. Hochreiter and J. Schmidhuber, "Long short-term memory," NeuralComputation, vol. 9, no. 8, pp. 1735–1780, 1997.
[5] K.Choetal.,"LearningphraserepresentationsusingRNNen-coder–decoder for statistical machine translation," in Proc. EMNLP,2014, pp. 1724–1734.
[6] J. Ma, L. K. Saul, S. Savage, and G. M. Voelker, "Beyond blacklists:Learning to detect malicious web sites from suspicious URLs," in Proc.ACM SIGKDD, 2009, pp. 1245–1254.
[7] M.Young,TheTechnicalWriter'sHandbook.MillValley,CA:Univer-sity Science, 1989.



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY