iJRASET
International Journal For Research in Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 9 **Issue:** VI **Month of publication:** June 2021

DOI: https://doi.org/10.22214/ijraset.2021.35284

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: 7.429
Volume 9 Issue VI Jun 2021- Available at www.ijraset.com

Detecting Malicious URLs using R-CNN and Cloud

Miss. Priyanka Vasant Ambilwade[1], Dr. Sagar Tambe[2]
[1, 2]*M.E Computer Eigg. Dept, P.R.E.C. Loni, Ahmednagar, Maharashtra-413736., Savitribai Phule Pune University.*

Abstract:** In today's information age as use of websites, mobile apps and all forms of information sharing forms have increased which gave rise to malicious URL forms. These malicious URLs are forwarded and users attention is diverted from the main course for what he is searching to other non-necessary and harmful content, thus wasting a lot of time and money. Theses malicious URLs have given rise to authentication thefts, money thefts and bullying of a user who falls in to a trap set by hackers by accessing these URLs. To resolve and find a solution to this kind of menace there is need to detect and prevent users from accessing these URLs. So, while studying various techniques put forward by various authors in different research papers, we found a few techniques quite interesting and useful. The first is detecting malicious URLs using CNN and GRU. The second is where a text mining technique is proposed using Natural Language Processing (NLP) which can be used for classification. The third is a combination of CNN and NLP. By studying them we came to understand that there should be a combination of both NLP and CNN together to implement a successful malicious URL detection system. So, in our paper we are proposing a fusion of R-CNN, NLP and Cloud together. The main work in our paper is to collect malicious and healthy URL which will be done using internet and multiple sources and combined as one dataset. Thus, we will use Google cloud to create a blacklisted URL database of our own and not depend upon multiple sources internet for them. In our system first we will create a blacklist database on cloud and then apply classification on it using NLP and machine learning algorithm SVM. The second step will be to use same URL dataset to train a R-CNN AI algorithm and get an output in form of malicious identified URLs. Then in the final phase we will compare the final results from SVM and R-CNN and analyse which one is efficient and highs and lows of the technique.

Keywords:** Malicious URL, Cloud Computing, NLP, CNN, R-CNN, SVM.

I. INTRODUCTION

In recent years internet has developed rapidly thus bringing a lot and lot of new users on the platform. As the use of internet increased so hacking activities also increased. In recent years hackers have stolen a large amount of personal and financial theft of a user which has cost an individual and big enterprises dearly. The process of data theft begins by sending offers in the form of URLs to users and luring a user to access them, thus when a user accesses and fills his personal information for offers the information is stolen and used to blackmail a user. So malicious URL by hackers have become a menace and there arises a need to address it with a good technique or else misuse of personal information will go on and on.

Today a large number of website URLs can be found with executable commands, SQL injections, XSS simply by embedding executable code or malicious code in URLs which can be used to lure and steal personal information of the user. So there arises a need to develop a full proof technique which can make use of power of computing, machine learning and artificial intelligence technology to detect malicious URLs at early stage and block the user from using them. These techniques if used together can create a lethal firewall for a website and personal information of a user and how he accesses it on the internet. Trends of malicious attacks can also be detected early. The process of malicious URL attack is demonstrated in the Fig.1.

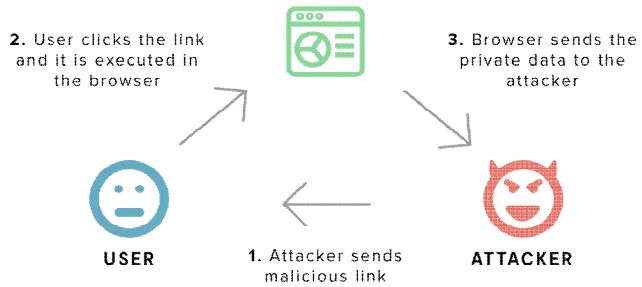


Fig. 1 Malicious URL attack.

©IJRASET: All Rights are Reserved

Thus, this kind of attack as shown in Fig.1 can be avoided by designing a mechanism by using machine learning and artificial intelligence techniques together. The first technique is natural language processing which can interpret a meaning of word and its grammar which can be used with various other techniques. The first technique we found useful by studying research papers by various authors is machine learning which can classify a URL in to malicious and safe classes. The other very powerful technique is to make use of artificial intelligence technique called as deep learning which can analyze any data deeply and predict good results thus help in detecting malicious URLs early. The mentioned techniques will also a database to keep track of the URLs and block them, thus making use of cloud computing mandatory for such kind of applications as cloud data can be accessed any time anywhere. So, in other words the main objective of this paper is to:

A. Focus on security of personal data of users.
B. Alert a user of malicious URLs at early stage.
C. Study Natural Language Processing techniques which is base of text mining.
D. Study various machine learning and artificial intelligence techniques.
E. Propose a new fusion frame work with a combination of NLP, Machine Learning, Artificial Intelligence and Cloud together.
F. Create a blacklist database of our own without depending on the internet for it.
G. Evaluate and analyze the new malicious URL detection framework and its strengths.

Thus, the rest of the paper is structured as follows:

1) Section II. explains literature survey which studies various techniques with their advantages and drawbacks.
2) Section III. explains the methodology i.e., mathematical model and algorithms to be used by the system.
3) Section IV. explains proposed system with block diagram or system architecture and working of the system.
4) Section V. shows the results of how the application are implemented and how they can be used.

II. LITERATURE SURVEY

This section describes the fundamentals of various techniques that can be used in designing a malicious URL detection system. It helps in understanding various ideas put forward by various technical papers published by various authors and how they put forth a more accurate and secured techniques. Some of the ideas with technique and drawbacks are mentioned below:

In 2020 Kumar et al. [1] presented the paper focusses mainly on malicious URLs using R-CNN. This technique is quite good and covers all the things needed for a successful malicious URL detection and prevention. But main drawback of this system is that it concentrates only on URL dataset that is found on the internet and does not make use of cloud computing technology to create a URL dataset of its own that can be perfected every while from the user end.

In 2018 Aloufi et al. [2] presented the paper focusses mainly on sentiment analysis of short text specifically twitter using machine learning algorithms on football specific tweets. This technique is quite good and covers all the things needed for sentiment analysis in a tweet on football. It can be implemented to detect malicious URL. But main drawback of this system is that it concentrates only on twitter dataset that is found on the internet and does not make use of cloud computing technology to create a tweet dataset of its own that can be perfected every while from the user end.

In 2019 Peng et al. [3] presented the paper focusses mainly on text classification using CNN. sentiment analysis of short text specifically twitter using machine learning algorithms on football specific tweets. This technique is quite good and covers all the things needed for a successful text classification of simple text and emotions behind it. But main drawback of this system is that it concentrates only on URL dataset that is found on the internet and does not make use of cloud computing technology to create a URL dataset of its own that can be perfected every while from the user end.

In 2020 Das et al. [4] presented the paper focusses mainly on malicious URL classification using CNN using RNN. This technique is quite good and covers all the things needed for detecting and finding malicious URL using a third-party dataset. But main drawback of this system is that it concentrates only on URL dataset that is found on the internet and does not make use of cloud computing technology to create a URL dataset of its own that can be perfected every while from the user end.

In 2020 Hussain et al. [5] presented the paper focusses mainly on fake news detection using naïve bayes and SVM. This technique is quite good and covers all the things needed for successful fake news detection. It can be implemented to detect malicious URL. But main drawback of this system is that it concentrates only on news article dataset that is found on the internet and does not make use of cloud computing technology to create a news dataset of its own that can be perfected every while from the user end.

In 2020 Mr. Ramraj S et al. [6] presented the paper focusses mainly on resume classification on LinkedIn using CNN and SVM.

©IJRASET: All Rights are Reserved

This technique is quite good and covers all the things needed for successful resume classification. It can be implemented to detect malicious URL. But main drawback of this system is that it concentrates only on LinkedIn dataset that is found on the internet and does not make use of cloud computing technology to create a news dataset of its own that can be perfected every while from the user end. In 2020 De Souza et al. [7] presented the paper focusses mainly on sentiment analysis of short text specifically tweet to detect offensive language in a tweet using machine learning algorithms SVM and naïve Bayes. This technique is quite good and covers all the things needed for sentiment analysis to find offensive tweets. It can be implemented to detect malicious URL. But main drawback of this system is that it concentrates only on twitter dataset that is found on the internet and does not make use of cloud computing technology to create a tweet dataset of its own that can be perfected every while from the user end.

In 2020 Fesseha et al. [8] presented the paper focusses mainly tagging of news articles to a specific label using machine learning algorithm SVM. This technique is quite good and covers all the things needed for tagging of a regional language news article. It can be implemented to detect malicious URL. But main drawback of this system is that it concentrates only on news article dataset that is found on the internet and does not make use of cloud computing technology to create a news dataset of its own that can be perfected every while from the user end.

III.METHODOLOGY

This section will study the mathematical conditions and algorithms to be used for designing a secured and multilayer malicious URL detection framework. These are explained as follows:

A. Mathematical Model

Our malicious URL detection technique can be explained in two sets with probability, success and failure conditions.

1) Classification Module

Set (C)= {C0, C1, C2, C3, C4, C5}

C0∈C = Fetch blacklisted dataset from cloud.

C1∈C = Extract features using OPEN-NLP.

C2∈C = Create training dataset for classification using two Classes safe and malicious.

C3∈C = Pass testing URL for classification

C4∈C = Apply SVM algorithm and perform classification.

C5∈C = View classification results in the form of prediction in two classes safe and malicious.

2) Deep Learning Module

Set (D)= {C0, C1, D0, D1, D2, D3}

C0∈D = Fetch blacklisted dataset from cloud.

C1∈D = Extract features using OPEN-NLP.

D0∈D = Design a text vector for text classification using CNN.

D1∈D = Train a R-CNN with various layers.

D2∈D = Pass a URL and get a prediction from R-CNN.

D3∈D = View prediction in two labels safe and malicious.

So, by studying the sets we come to notice that many elements are common in both modules and used in coordination in both sets so they be placed as

$$x \in C \cap D \text{ if } x \in C \text{ and } x \in D$$

Thus, the probability of intersection of elements in both modules can be given as

$$P(C \cap D) = P(C)+P(D)$$

So, intersection of common elements can be shown as

$$C \cap D = \{C0, C1\}$$

The conditional probability of both modules using the same element can be shown as

$$P(C|D) = \frac{P(C \cap D)}{P(D)}$$

©IJRASET: All Rights are Reserved

ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.429
Volume 9 Issue VI Jun 2021- Available at www.ijraset.com

Thus, we conclude that our malicious URL detection framework's success and failure will depend upon the internet, i.e., if the internet is not present blacklisted URL dataset cannot be fetched from the cloud and further classification and deep learning cannot give results thus this is a case of failure, so our framework supports NP-Hard and not NP-Complete.

B. Algorithms Used

Our malicious URL classification process can be explained using following algorithmic steps.

```
1: procedure CLASSIFY
2:     urls[]=fetchURLDatabase
3:     urlsCount=urls.length()
4:     if urlsCount == 0 then
5:         status=addTrainUrl(newUrl)
6:         addTestUrl(newUrl)
7:         if status == True then
8:             Create deep learning model
9:             Train model
10:            saveStatus=Save Model()
11:            if saveStatus == True then
12:                Load testing dataset
13:                Apply model
14:                classify URLs
15:            else
16:                result ←Exit procedure
17:            end if
18:        else
19:            result ←Exit procedure
20:        end if
21:    else
22:        Repeat Step 2:
23:    end if
24:    View URL classification results.
25: end procedure
```

.

IV. PROPOSED SYSTEM

This This section is mainly divided in 3 main modules with other sub parts in them. The text that follows explains the modules with a block diagram or system architecture as shown in Fig.2. to illustrate them. The working of the framework is explained as:

1) *Blacklist URL Dataset:* In this module we are proposing to use Google Drive as Cloud. The Google Drive sub cloud Google Sheets will be used to store blacklisted URL's and feature keywords. The google sheets is free cloud where the information can be stored and retrieved at any time anywhere. To store URL and keywords we will have a form in desktop application which will use google sheets as backend. The communication with the cloud and desktop application will be done using Google Drive and Google Sheets API. This cloud model will be used by both techniques i.e., Malicious URL detection using SVM and R-CNN.
2) *SVM:* In this module we first propose to fetch the backlisted dataset from google cloud. Then feature extraction will be done on blacklisted URLs using OPEN-NLP. The extracted feature keywords will be used to create a training dataset which will train SVM. Then a testing URL will be passed from which a testing dataset will be created. Both training and testing datasets will be passed to SVM algorithm which will give a classification results in the form of two classes i.e., safe and malicious.
3) R-CNN: This this module we also first propose to fetch the backlisted dataset from google cloud. Then feature extraction will be done on blacklisted URLs using OPEN-NLP. The extracted features will be used to create a text vector which will be used to train a R-CNN model. The R-CNN model will have all the layers necessary to predict the results properly. The layers will be fine-tuned for better predictions if necessary. Then after training the R-CNN a URL from dataset or any other URL will be passed to it for prediction. The prediction will be in the form of two labels i.e., safe or malicious

©IJRASET: All Rights are Reserved

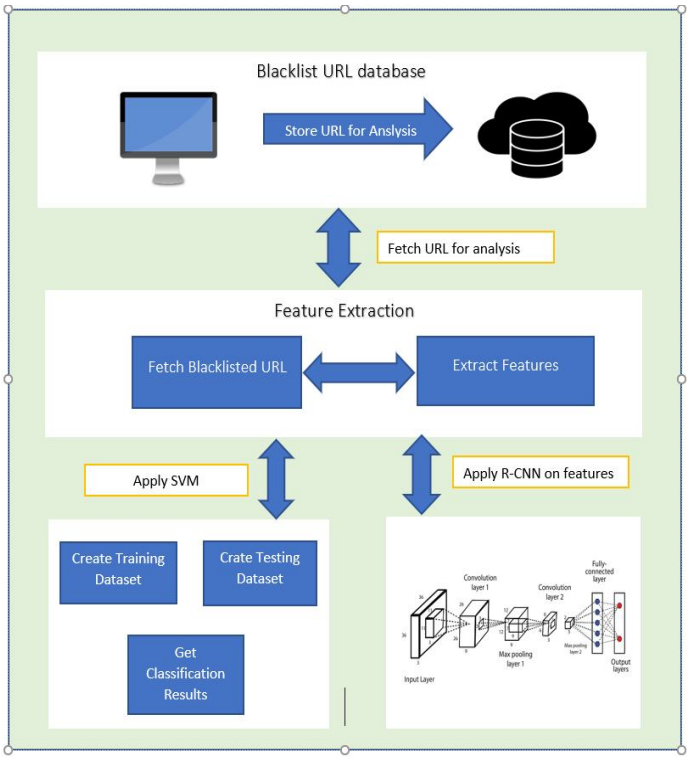


Fig.2. System Architecture Diagram.

V. RESULTS AND DISCUSSION

Thus, to explain the above proposed system we have created 3 applications. One for admin desktop where he can analyze the URLs in 2 categories safe and unsafe and upload the URLs so that a user can browse safely using the online database. Some of the relevant screens are shown below.

Fig.3. Finding malicious URL

In Fig.3 it classifies the URLs in safe and unsafe. First the training URLs are collected using internet and added to training dataset. Then a testing dataset is provided to classify. Then the model is trained and save for testing purpose. After applying the model, it will classify each test URL passed to it using testing dataset which is further uploaded to the cloud so that the users can browse safely.

©IJRASET: All Rights are Reserved

Fig.4. URL Database.

In Fig.4 it shows how the URLs look on the Google sheet after upload. This URL database can be used to safely access the internet from the mobile app by utilizing the URL database.

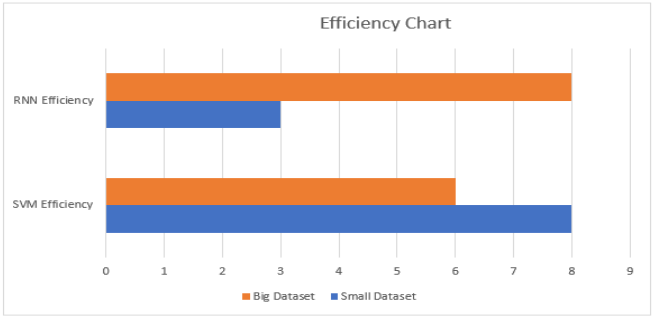


Fig.5. Efficiency chart.

In the fig.5 we come to conclude that efficiency of SVM is greater than RNN when the dataset is small. But the efficiency of RNN is very good when the dataset is large, so we conclude that AI is better than machine learning when it comes to larger datasets.

VI. CONCLUSIONS

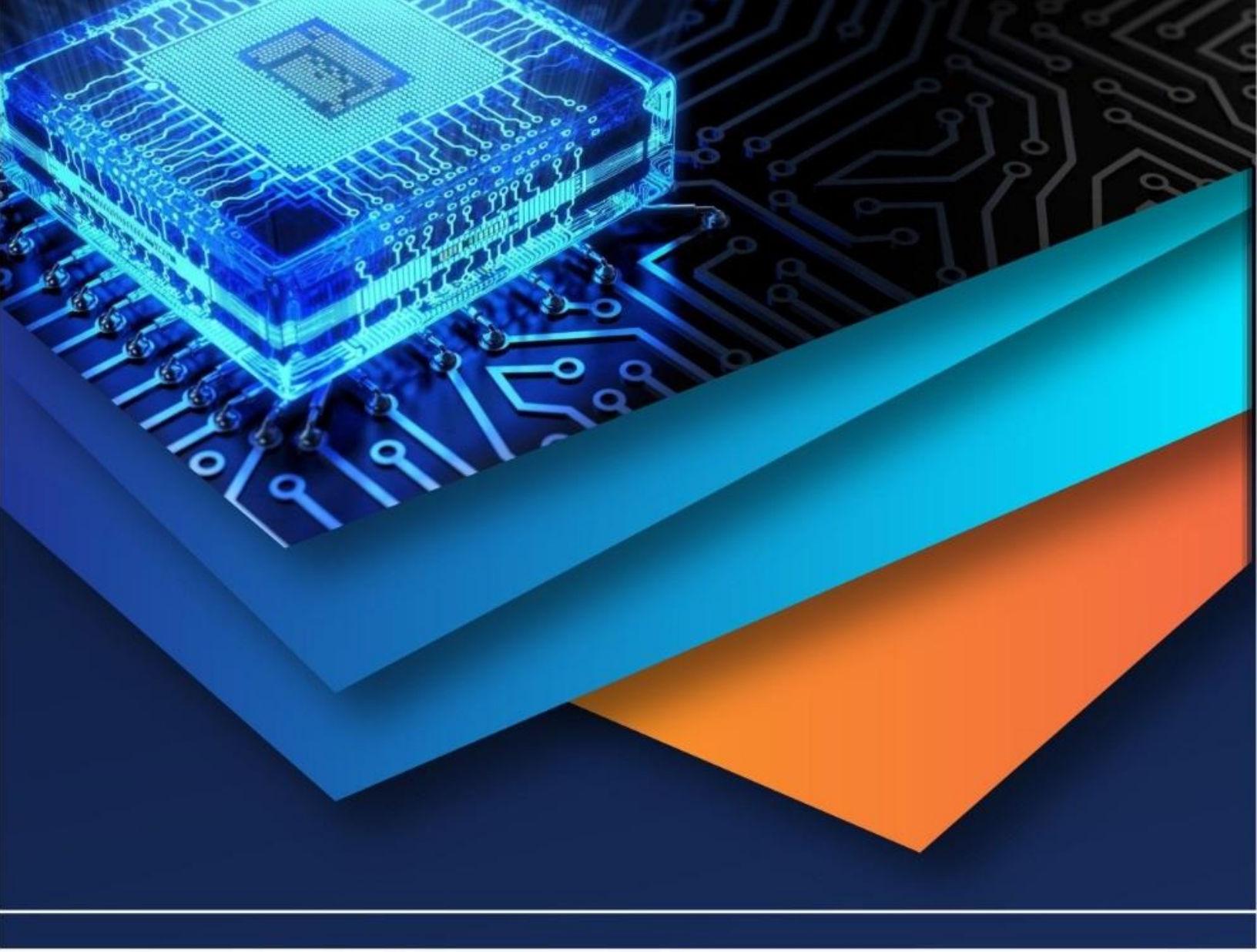
In this paper we conclude to develop a novel approach of malicious URL detection system using machine learning algorithm SVM, artificial intelligence algorithm R-CNN and cloud computing technology together to create a frame work to make use of blacklisted URL with the algorithms. We are making use of both machin elearning and Artificial intelligence techniques together and test them on various sizes of datasets desgined indegenously. We studeied and tried to incorporate studeis from [1][2][3][4][5][6][7][8] studeis slightly as needed.We are concluding to make effective use of malicious URL dataset found on the internet and make a cloud dataset of our own using Google Drive. We are concluding to try to combine and make use of Java and Python languages together to get effective output from the frame work. We conclude to fine tune R_CNN architecture to get the desired results. Thus we conclude that our framework will be helpful in preventing attacks by hackers using malicious URL thus saving a lot of time and money that is wasted by these attacks.

©IJRASET: All Rights are Reserved

REFERENCES

[1] Sudhanshu Kumar, Kanjar De, and Partha Pratim Roy, - Detecting Malicious URLs via a Keyword-Based Convolutional Gated-Recurrent-Unit Neural Network, in IEEE-2020.
[2] Samah Aloufi , and Abdulmotaleb El Saddik., - Sentiment Identification in Football-Specific Tweets, in IEEE-2018.
[3] Song Peng, Li Zhijie and Geng Chaoyang., - Research on Text Classification Based on Convolutional Neural Network, in IEEE-2019.
[4] Arijit Das, Ankita Das, Anisha Datta, Shukrity Si and Subhas Barman5, - Deep Approaches on Malicious URL Classification, in IEEE-2020.
[5] Md Gulzar Hussain, Md Rashidul Hasan, Mahmuda Rahman, Joy Protim and Sakib Al Hasan, - Detection of Bangla Fake News using MNB and SVM Classifier, in IEEE-2020.
[6] Mr. Ramraj S, Dr.V. Sivakumar and Kaushik Ramnath G, - Real-Time Resume Classification System Using LinkedIn Profile Descriptions, in IEEE-2020.
[7] Gabriel Ara´ujo De Souz and M´arjory Da Costa-Abreu, - Automatic offensive language detection from Twitter data using machine learning and feature selection of metadata, in IEEE-2020.
[8] Awet Fesseha, Shengwu Xiong, Eshete Derb Emiru and Abdelghani Dahou, - Text Classification of News Articles Using Machine Learning on Low-resourced Language: Tigrigna, in IEEE-2020.
[9] B. Cui, S. He, X. Yao, and P. Shi, - Malicious URL detection with feature extraction based on machine learning, Int. J. High Perform. Comput. Netw., vol. 12, no. 2, pp. 166–178, 2018.
[10] S. Sheng, B. Wardman, G. Warner, L. Cranor, J. Hong, and C. Zhang, - An empirical analysis of phishing blacklists, in Proc. 6th Conf. Email Anti-Spam (CEAS), Sacramento, CA, USA, 2009, pp. 59–78.
[11] J. Ma, L. K. Saul, S. Savage, and G. M. Voelker, - Beyond blacklists: Learning to detect malicious Web sites from suspicious URLs, in Proc. 15th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining, Paris, France, Jun./Jul. 2009, pp. 1245–1254.
[12] Yazan Alshboul, Raj Kumar Nepali, and Yong Wang. Detecting malicious short urls on twitter. In AMCIS, 2015.
[13] Betul Altay, Tansel Dokeroglu, and Ahmet Cosar. Context-sensitive and keyword density-based supervised machine learning techniques for malicious webpage detection. Soft Computing, 23(12):4177–4191, 2019.
[14] Ram B Basnet, Andrew H Sung, and Quingzhong Liu. Feature selection for improved phishing detection. In International Conference on Industrial, Engineering and Other Applications of Applied Intelligent Systems, pages 252–261. Springer, 2012.
[15] Nicolo Cesa-Bianchi and G´abor Lugosi. Prediction, learning, and games. Cambridge university press, 2006.
[16] Marco Cova, Christopher Kruegel, and Giovanni Vigna. Detection and analysis of drive-by-download attacks and malicious javascript code. In
[17] Proceedings of the 19th international conference on World wide web, pages 281–290, 2010.
[18] Giovanni Vigna Davide Canali, Marco Cova and Christopher Kruegel. Prophiler: a fast filter for the large-scale detection of malicious web pages. Proceedings of the 20th international conference on World wide web. ACM., 2011.

©IJRASET: All Rights are Reserved



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Call : 08813907089 (24*7 Support on Whatsapp)