Contents

[**1.** **Introduction** 2](#_Toc49090495)

[**2.** **Related Work** 5](#_Toc49090496)

[**3.** **Methodology** 18](#_Toc49090497)

[3.1 URL Dataset: 19](#_Toc49090498)

[3.2 Feature Extraction & Selection: 19](#_Toc49090499)

[3.3 Model Selection and Training: 19](#_Toc49090500)

[3.4 Evaluation and Comparison: 20](#_Toc49090501)

[**4.** **Implementation** 20](#_Toc49090502)

[**5.** **Results and Discussion** 21](#_Toc49090503)

[**6.** **Conclusion and Future work** 26](#_Toc49090504)

[**References** 27](#_Toc49090505)

Detection of Malicious URL using Deep LSTM Neural network

**Abstract**

Malicious URLs are highly dangerous for the computer users. From various surveys it has been noted that one-third of all the generated URLs are used for performing malicious activities. Various anti-virus companies have used the blacklisting and signature-based methods to detect the malicious URL. However, these methods are useless as it cannot be utilized to detect the new URLs, which are not available in the list or database. Therefore, machine learning methods have been used for feature-based analysis of data. Detection of such URLs with low False positive rate is a challenging task. In this work, we are using the Deep LSTM neural network to improve the classification performance. Along with the LSTM neural network, we have also implemented 4 different machine learning models as a proof of concept. Using LSTM neural network, the highest accuracy of 97.82% and PRF score of 0.98 have been achieved.

# **Introduction**

In the modern world, connectivity is ruling the world and internet has been a part in our day to day life. World Wide Web (WWW) connects all corners of the word and assists people in multiple ways such as online digital financial transactions, trading, digital marketing, travel bookings, video conferences, entertainment such as social media, OTT (Over-the-top media) services and the list goes on. Most of the people from developed and developing countries are highly dependent on internet which is making it more vulnerable to fraudulent activities such as hacking into personal social networking or bank accounts, confidential working databases, competitor company databases, digital transaction schemes/scams based on fake URLs and various credit related robberies. A report from Cybersecurity Ventures sponsored by Herjavec Group generated in 2019 predicts an excess of $6 trillion annual loss to the world by 2021 due to cybercrime [1]. This loss is increased from $3 trillion to $6 trillion from 2015 to 2021.Hacking can be done through different means such as malware, spamming, DDoS (distributed denial of service), phishing and MITM (man in the middle, drive by download etc., Most common hacking is done through phishing and spamming using malicious URL.

Uniform Resource Locator (URL) acts as a web address for a web resource, which indicates its location and retrieving mechanism. These URLs are used for accessing the websites through a client browser.URL consists of protocol identifier, domain name and other detailed information, which makes it complete web address used to open a specific web page of a website. Each web page in the internet will have unique web address. Protocol identifier of URL (ex: http, ftp) specifies the protocol through which a webpage or an executable file can be fetched. Domain name in the URL is considered as the nick name of an IP address which specifies the location of a web source. Various devices on internet will navigate through IP address. The World Wide Web consists of huge number of URLs out of which one third of the URL's are one or the other way malicious [2]. Most of the users become bait to cybercrime through the malicious URL. URL is considered as malicious which is created with the purpose of promoting cybercrimes. Through a malicious URL, an individual personal data or a company’s confidential data can be accessed by unauthorized source. Cybercrime is considered as one of the major threats to mankind. In order to prevent the users to use malicious URL’s, which is the most used means of many cybercrimes in the world, an approach called ‘Blacklisting’ is introduced. In this approach, the URL which is considered as malicious will be blacklisted or blocked. This can be done through different techniques such as web crawlers, heuristics or manual reporting by users. On an average, Google blacklists 10,000 website URLs every day on security grounds. The biggest challenge is that the existing techniques are not capable of dealing with large number of URLs as there are over 1.7 billion websites are present on the World Wide Web (WWW) as of today. In order to make internet a safe place, numerous URLs across web needs to be monitored, detected and deleted frequently and ensure digital data security. For this purpose, many researchers came up with a new technique popularly known as Machine Learning (Artificial Intelligence) which is used extensively in Data Science. Data Science is a stream of science which deals with data analytics and statistics.

Machine Learning is a technique which uses mathematical and statistical algorithms to study and analyze the data and converts it into understandable pattern [3]. Machine Learning has the capability of dealing with huge data sets and also can process the real time data based on the patterns and structural properties of provided data. In machine learning, a model or system will be trained in such a way that it takes decisive actions and predict the future behavior of a system based on the past historical and statistical data. In our case, the machine learning algorithms will be run on huge sets of URLs, which can detect and list the URLs as malicious and benign based on the provided data. Illegal URLs and websites with phishing and spamming content are considered as malicious whereas Benign URLs are the URLs which are listed as harmless. Along with huge sets of URLs to be scanned for blacklisting approach, some properties of the websites like lexical features (text, words and contextual information) and graphical location of the host needs to be presented to the Machine Learning algorithms to train the real time model which can detect and blacklist or delete malicious URLs. Once the data and extra features are extracted and presented to model, classification algorithms such as Logistic Regression (LR) and Support Vector Machine (SVM) which are used in our case of research are applied to the trained model to get the results.

Machine Learning (ML) plays a prominent role in the technical areas where big data processing is required to run a business model such as digital marketing, stock market trending, hardware maintenance, communication, medicine and customer services etc., Mathematical statistics and deep learning (Artificial Intelligence) algorithms are the foundation for core Machine Learning. For applying machine learning on big data, one needs large storage memory which in turn requires processing hardware (computers or workstations) to compute the data and to perform numerous functions on it to give future predictions and take decisive actions if necessary. To overcome the hardware and storage limitation, distributed machine learning is introduced. The concept of distributed machine learning consists of a collection of nodes (workstation or computer) which are connected to the network. These nodes serve as storage unit and computing hardware which processes subset of data present on the node and analyses it to take necessary action. In order to deal with such big data sets, scaling up of machine learning algorithms is necessary and few such algorithms are proposed and implemented by [4] , [5] & [6] etc.,

In this research we are improving the classification performance of URLs using the Deep LSTM neural network. In order to prove our concept, we have also implemented 4 different machine learning models to perform a comparative analysis. The main objective of this research is to reduce the false positive and false negative errors from the system. The performance for false positive and false negative values has been calculated using precision, recall and f1-score. Among the machine learning algorithms, the training time of each algorithm has also been calculated. Whereas the training time of deep LSTM neural network mainly depends on the number of epochs. Therefore, it has not been calculated. Choosing the optimal Parameters for algorithms is another important step. Which has been performed iteratively for an optimal result. In this work, the Malicious URL will be identified based on the lexical and host-based properties of URL.

## **Research Questions**

In this problem we are trying to provide the answers for following research questions. The research questions are as follow:

* How efficiently Deep LSTM neural network reduces the false positive and false negative prediction rate and improves the model performance?
* Does high accurate machine learning algorithm requires, high amount of time to train the model ?

# **Related Work**

This research aims at running distributed machine learning algorithms as discussed in the previous section on large data sets to de-list the malicious URLs in order to make internet a safe place. In this case big datasets consists huge number of URLs which are classified as malicious and non-malicious. Many researchers have proposed and implemented multiple techniques to classify the malicious URLs and benign URLs. These techniques include both classic methods and modern machine learning algorithms. This section mainly discusses about the machine learning algorithms developed and implemented by the past decade researchers to detect malicious URLs.

An experimented with a public dataset consisting of 2.4 million URLs and 3.2 million features. In this, different classification techniques such as Decision Trees, k-nearest neighbor, Bayesian networks (Naive Bayes classifiers), Random forest, Support vector machine and Multi-Layer perceptron are used on 2 million entries which has 3 million variety of attributes [10]. The main aim of this research is to compare the accuracy, efficiency and precision of different classification algorithm mentioned above by categorizing the datasets based on their features like binary and real value attributes. Datasets are categorized as the dataset containing only real value attributes, dataset containing only binary attributes and dataset containing both binary and real value attributes. A total of 121 datasets with multiple probability of malicious and benign URLs are presented to the training model (uses different classification algorithm) in this experiment and compared the predicted results for accuracy and precision of the classification algorithm used. The experiment made sure that the results across different feature sets are fair and concluded that all the classification algorithms yield better results in accuracy and ranking wise Random Forest is considered as most accurate classification algorithm for this problem statement. The accuracies obtained in this experiment are obtained without using advanced feature selection algorithms [10].

Another fundamental research [11] done on classification algorithms using semi supervised learning. The main aim of this research is to estimate the importance of unlabeled data in improving the performance of classification algorithms. This analysis uses semi supervised learning technique for classifiers based on Bayesian networks. This research has experimented with image processing applications which require classification such as face detection, facial expression recognition and came up with results. To get better results with unlabeled data, Stochastic Structure Search (SSS) method is used. This study summarizes that the semi supervised learning model performance gets degraded with unlabeled data using on Naïve Bayes method. To overcome this, algorithms to be improvised to run on Bayesian network structure or generate more labelled data to yield better results in classification problems.

Another author has proposed an adaptive learning method to detect malicious URLs online. This method targets the detection of malicious URLs in dynamic environment which in turn changes the statistical parameters of the target parameter [12]. For example, hackers can change website domains or IP addresses or URLs of a malicious website. To overcome the challenges faced with unstable or non-stationary environments, the machine learning model is trained with drift detection concept in this research. Concept drift detection is a real time supervised learning scenario when the target variables are changing in relation with the inputs fed to the model. To detect the concept drifts, partial feedback is used with manually labelled data. In this research they experimented by collecting IP blocks, Destination port, and Domain feature from the huge dataset of URLs and inputting them to the training/predicting module. In order to calibrate the efficiency of the proposed technique in detecting malicious URLs, the model is trained with different algorithms such as Decision Tree (DT), Gradient Boosted Tree (GBT), Linear support Vector Machine (LSVM), Logistic regression (LR), Naïve Bayes (NB) and Random Forests (RF). Out of these classifier algorithms Decision Tree, Gradient Boosted Tree and Random Forests are better in accuracy. Coming to the proposed technique in this research used the concept of drift detection through nonparametric test approach. This approach is demonstrated using two algorithms cumulative sum and Wilcoxon Rank-Sum Test on artificial and real time datasets. Wilcoxon Rank-Sum nonparametric test algorithm outperformed the cumulative sum in speed and accuracy when applied on both artificial and original datasets in dynamic changing environment [12]. This research contributed in introducing and developing an adaptive learning system to detect benign and malicious URLs which runs in backbone networks with less delay and more accuracy.

Deep learning is a branch of machine learning; deep learning yields better results in taking decisive action and predicting the behavior of any system based on the datasets. One such research [13] demonstrates the implementation of the deep learning algorithms in malicious URL detection. This research designs a convolutional gated recurrent unit (GRU) neural network based on keywords detection algorithms to classify malicious and benign URLs. Here, deep learning technique is used to detect malicious URLs by gathering different features of the dataset. A model equipped with deep learning techniques can learn from very primitive inputs. In this experiment, convolutional neural network and gated recurrent units are deployed for run time extraction of features from large datasets and apply them on data classification algorithms. This study shows the efficiency of neural networks when compared to the models where the features are extracted manually. Manual extraction of features from huge datasets results in consuming more time and less accuracy. Here the proposed system considers file paths, domain names and registries as keywords which are used as base for the classification of malicious URLs. This experiment setup specially uses Keyword based URL character embedding module which can easily extract the feature based on the keywords present in the URL. This research proved that application of neural networks/ deep learning to extract features across big datasets compared to manual extraction yield good results in terms of accuracy, response time, precision and flexibility with different classification algorithms in real time and dynamically updating environments.

A research proposed a method to deal with big datasets, in this case hundreds of millions of URLs, using two filtering models based on descriptive and lexical features of URLs [14]. It also makes use of short span features of an URL efficiently in detecting the malicious URLs. In this technique, researchers used filtration system that can work in real time by extracting dynamic and static features from the URL strings. The lexical characteristics such as delimiters, zero weight words, IP address and words generated in certain timeframe of the URL are extracted to classify the URLs based on blacklist rule but the lifetime of malicious URLs is very short and hackers comes up with different URL once it is reported as malicious and descriptive features such as length, country of the domain, letter, digit and symbol count, length ratio are extracted from big datasets. These extracted features are fed to the proposed framework as shown in the figure, which consists of a learning model based on the passive-aggressive algorithm (PA) and Confidence Weighted learning algorithm (CW). Passive-Aggressive (PA) algorithm supports as the learner from descriptive features whereas confidence weighted (CW) algorithm is chosen as the learner algorithm for lexical features since the nature of these features are similar to Natural Language Processing (NLP). In this experiment, the online learning model is fed with large real time imbalanced data set nearly one million URLs is chosen. This research summarizes that as per the results of this experiment, the model which is learned from lexical features is dynamic in nature and this needs to be updated every time with adaptive learning technique whereas the filter with descriptive feature is stable but gives poor results in detection of malicious URLs when compared to lexical feature filter [14].

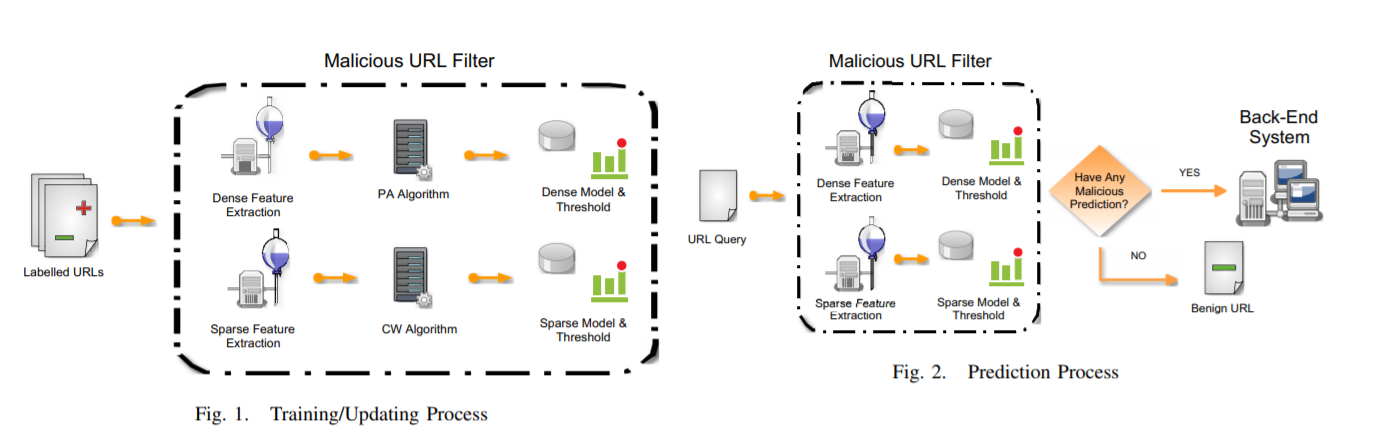


Figure 1 : Proposed URL detection architecture by [14]

A different proposal concentrates on host related features such as location (IP address, geographical features, numbers and prefix), owners, registers, registering date and lexical features such as notations, delimiters and keywords those frequently appears to classify the malicious URLs [15]. Host based features are extracted from WHOIS data through a php script which extracts details of the registrars such as geographic location. Lexical features can be extracted from the URL format. For this implementation all the features are collected daily and create as vectors which are fed to the learning model to classify the URLs based on the labelled data. The proposed framework executes and compares different learning techniques such as Perceptron, Logistic Regression (LR), Passive Aggressive (PA) and Confidence Weight (CW). When compared to simple algorithms like perceptron, recent algorithms are efficient with online learning models with labelled data. This experiment concludes Confidence Weight (WF) algorithm on large and dynamic featured datasets yield good accuracy when compared to other machine learning classifier algorithms [15]. It proved that modern machine learning techniques based on mathematical statistics are faster, flexible, accurate and reliable than the classic methods in binary probabilistic problems like classification of malicious and non-malicious URLs.

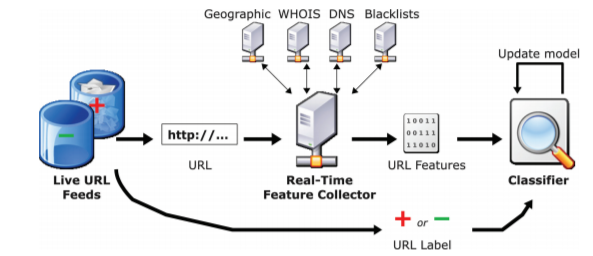


Figure 2 : URL classifier Proposed by [15]

In this social media era, twitter is a popular microblogging service for normal people, celebrities, politicians, entrepreneurs and all kinds of people to express regional, political, religious, economic, nationalist and social views. Many hackers are finding this as bait to introduce malicious content and spamming the accounts [16]. Their experiment determines the suspicious twitter accounts based on few features. After collecting the data, Page Rank algorithm is used to sort the popularity of an account and remove the reciprocity. In this experiment, the entire twitter is scanned and obtained 41.7 million user profiles, 4,262 trending topics, 1.47 billion social relations and 106 million tweets. From this big dataset, multiple features are extracted such as retweets, number of followers and user participation in trending topics. In order to yield reciprocity of users needs to be removed from the collected data. This is done based on the short span accounts and pattern of tweets. Based on all the features extracted, rankings are given using page rank algorithm and the accounts which are highlighted are considered as suspicion accounts.

An author proposed a method to detect malicious URLs on social media platforms such as Twitter, Instagram, Google plus, Facebook, Myspace and Snapchat etc., People uses these platforms to connect each other, to communicate, to spread news, to express various views and money transactions (recently introduced with some platforms). In this experiment, researchers picked a microblogging platform popularly known as Twitter. The main aim of this research is to classify the URLs shared in Twitter by collecting the feature based on URL redirection [16].

Social networks (Facebook, Twitter, Google plus, Myspace etc.,) are catered to different platforms which can be run through various devices such as mobile handsets, computers, tabs, smart watches, notebooks and all devices which are capable of interacting with internet and equipped with a browser or platform which can run different applications (Android apps, IOS apps). This makes the social networking more vulnerable to cyber-attacks. In this research, malicious URLs are detected posted in Twitter, through which user can post 140 characters length string. Due to the limitation of 140 characters Tweet, twitter uses a service called shortening of URLs and these shortened URLs can be used only with Twitter platform and they can’t be used externally. Shortened URLs posted on twitter are secured by t.co wrapper. This also helps to keep track of the user clicks on a tweeted URL. URL is posted on twitter wall to communicate a message or to share some information. Sometimes hackers use these URLs to introduce virus into the system, once any follower (person who follows another user on twitter by clicking on follow button which makes him to know the updates posted by the user) clicks on the malicious URL introduced by the hacker, the virus attacks the system of the follower. One twitter account can have multiple number of followers and again each follower can have followers which makes a big connectivity. Due to this connectivity, such malicious URL can cause a lot of damage. For the implementation of the proposed technique, real time data from twitter is collected. From this data, the tweets which are having URL are collected and the shortened URL is converted into long URL. Through this, the correlation of URL redirection is extracted from the database. The suspicious URLs are detected by connecting the most frequently re-tweeted URLs to a crawler browser (static and dynamic), which can detect the malicious content. The proposed system has client and server setup. On client side, user can register an account through which he can see the tweets, followers, tweeting wall in which he can post and other topics on his point of interest. On registration, the tweets with URLs tweeted by the user will be stored into the database. The URLs also forwarded to crawler browser and stores the report. On the other hand, server collects the database and performs the actions to extract the features of the URL such as domain group, re-direction frequency and entry point by observing the actions done by client and crawler browser. Server can do the actions on the account such as downloading tweets and fed them to foreground running crawler, which converts the shortened URL to long URL. Once it is converted the system compares this URL is related to any other URL. If it finds any correlation, then it decides malicious or not malicious based on previous collected URLs. The URLs which are benign are forwarded to training model and it gets added to the already existing dataset, which helps to yield better results by running machine learning algorithms on the big sets of data.

Server works as standalone so that it can detect malicious URLs during processing itself which helps in stopping the huge spread of attack. All the modules in the proposed system should run in synchronization on the collected datasets to be consistent with the results. The entire system is portable and can be deployed across the platforms which make it operating system independent. This feature of this application is very important in the recent technically advanced world where different scale of devices is in market running on different platforms and being hosted on different operating systems. Existing systems collect the data for a fixed amount of time span like 6 months or for one year and feed those to the learning model to detect the malicious URLs which makes them less efficient in dynamic environments and also not capable enough to redirect the URLs on runtime. The proposed system extracts the features from dataset which is collected in real time and feed them to the dynamic crawling. This helps in detecting the malicious URLs online more efficiently. This makes the system more adaptable to the imbalanced and dynamically unpredicted environments and helps in detecting malicious content at early stages. The dynamic detection property of the proposed system makes it more popular and can be used with any search engine capable of storing more data. Also, the experiment shows that the system yields 90 percent accuracy as the systems gets the real time data on run time when a user attempts to post a tweet on his wall. Hence the proposed system in this research is more efficient and has great adaptability nature where in this system comes handy with day to day flooding malicious URLs introduced by hackers through social networking platforms. Implementing this system can reduce the rate of cyber-attacks introduced through social media to a greater extent.

The paper published by [17] discusses about a technique to classify the malicious and non-malicious URLs by taking care of class imbalance. In this case, various features are extracted from large datasets and feature selection is implemented. The extracted features are ranked based on their technical importance in detecting malicious URLs and associated with rules to define the relation between different features. The demonstration of proposal goes through different phases such as collection of data, pre-processing of data, taking care of class imbalance, feature selection and association rule making.

In data collection phase, data collected using different tools such as country servers with WHOIS. This experiment is conducted on stored datasets with 1781 URL samples which are having 20 features. The 20 features of the dataset used for this research are URL, URL Length (number of characters in the URL), Number of special symbols (/,@,%...), character set (character encoding standard), server system (server system’s operating system extracted from the packet), domain content length (number of characters in HTTP header), geographical state (website origin state), geographical country (website hosting country or country of origin), website registration date, URL update date, number of TCP packets exchanged, number of various remote ports used which are different from TCP, number of remote connected IP’s, application bytes ( number of bytes transmitted to and for), remote application packets (packets received and acknowledged from server), source application packets (packets exchanged between server and client), application packets (number of packets generated), source application bytes (number of bytes transmitted between server and client), remote application bytes (received bytes from server), DNS server query timings (number of DNS packets sent) and types (class) of websites. The data collected initially misses some values and comprises of class imbalances.

Pre-processing phase re-scale and normalize the large range of values received from the collected data to a particular range. Here the missing values are replaced with most used items. The next phase takes care of class imbalance. Class imbalance is the scenario where the collected data has different number of samples. In this experiment, the dataset collected consists of two classes of data, one is malicious URLs and other is benign URLs. One class has more samples (1781 malicious URLs) than another class which is having very less number of samples (216 benign URLs). Machine learning algorithms yields better results, when applied on balanced classes. Over sampling is used to balance the class imbalance by achieving the same number of samples for each class. To achieve this, Synthetic Minority Over-Sampling Technique (SMOTE) is used. SMOTE synthesizes the samples for minority class based on the previous occurrence’s history of existing minority samples.

After balancing the class imbalances, the pre-processed data is served as input to several classification algorithms such as Random Forest (RF): a supervised learning algorithm which divides the data into different trees and combine them for accurate results and clear prediction, Decision Trees (DT) approach : which classifies based on various parameters, Logistics Regression (LR): this approach is followed when the classification is binary type, K Nearest Neighbors algorithm (KNN): this is non parametric machine learning approach in which different class of data is divided into number of clusters and predicts the class of new data sample and last algorithm is Support Vector Machines (SVM): this is based on the hyper plane to categorize the class of the data. When this is used in combination of Random Forest (RF), the supervised learning model yields optimal results.

Feature selection phase uses Recursive Feature Elimination method, which consists of a trained model to eliminate the non-significant features. The trained model uses a ranking system based on the coefficient of model or a recursive loop which detects the less important features iteratively and eliminates them. After eliminating non-significant features, the data should go through associative rule mining phase. Associative rule mining is a rule-based machine learning approach which defines or establishes a relation between different feature variables extracted from datasets which helps in grouping the data. For example, bucketing approach is introduced to group the data into different buckets and in turn it optimizes the classification algorithm functionality. In this experiment, the researchers used two different algorithms to achieve associative rule mining. They are FP-Growth, Apriori, Decision Tree Rule making. Apriori is an algorithm which runs through hash tables of dataset and sort or contrasts them with the other datasets. These scores are used to determine the frequency of the occurrences. This approach traverses through bottom up and make use of the hash tables to access the data from the database. Apriori is used in combination with other pre-defined algorithms to sort or differentiate data and label them accordingly. FP-Growth algorithm accesses the data through Depth First method. This algorithm groups the data based on the frequent dataset mining and eliminates the explicit generation of datasets. Decision Tree Rule algorithm is also used to sort the data. In this phase all the generated rules are used across the samples collected from the dataset. The samples are related to yield better accuracy instead of coverage.

This research concluded that the Random Forest (RF) classification algorithm gives 96 percent accuracy when compared to other classifier algorithms when the sample dataset processed through various times of training and verification iteratively. The feature selection techniques used in this research shown better results with only 8 features out of 20 in detecting malicious URLs. Those features are source application bytes (number of bytes transmitted between server and client), application bytes (number of bytes exchanged), State (website origin state), Number of special characters (/,@,%...), URL Length (string length of the URL), number of various remote ports used which are different from TCP ), remote app packets (packets received from server) and remote app bytes (received bytes from server). The rules generated in this experiment by machine learning algorithms like FP Growth, Apriori, and Decision Tree Rule making are efficiently contributed in detecting malicious URLs with high accuracy.

Another author came up with a new approach which can detect the malicious URLS in real time. Proposed system detects a malicious URLs based on machine learning algorithms like Logistic Regression (LR) and it also provides a plug-in which alerts the user about the nature of the URL i.e., malicious or genuine. The proposed model addresses the limitations of the existing classification systems such as dealing with dynamically changing environment (URLs getting added to the system more frequently), alerting the users about malicious URLs which looks similar to genuine website’s URL, rapid change in the structure of the URLs and addressing the issue of personal data theft such as bank credentials by making users aware of the malicious, phishing and spamming content beforehand by introducing a plug-in which can alert the user on accessing malicious URL [18].

The proposed system uses machine learning algorithms to classify malicious and benign URLs and plug-in as well. This is implemented in three phases Tokenization, Classification and Plug-In. In Tokenization stage, the lexical and text features of website URL are tokenized or labeled. In Classification stage, the classification algorithm Logistic Regression (LR) is applied on the labeled or tokenized features collected in first phase. Logistic Regression is a machine learning algorithm which is based on the binary classification. This algorithm predicts and classifies the URL based on the training data (previous historic data) given to the algorithm. The last stage is Plug-In, the front-end implementation of the proposed system. If user enters a suspicious URL in the plug-in it inputs the URL to machine learning algorithm which classifies the URL as malicious or benign and alert the user by tagging the malicious URLs as bad and benign URLs as good. The proposed method is applicable in the real time detection of malicious URL and makes user aware of the threat before resulting in cyber-attack through plug-in, which throws an error message if it detects a malicious or spamming URL. The main advantage is this can be applied throughout the social network which is rampant across the world these days.

In the other research author proposed a multi-layer filtering model to classify the malicious URLs and normal URLs. The existing systems used different blacklisting and machine learning classification algorithms which has their own advantages and disadvantages as discussed in earlier research works. Considering this fact, in the proposed technique of filtering URLs, the model comprises of four-layer filters which used different classification algorithms such as Support Vector Machine (SVM), Naïve Bayes etc., The four layers are Stratified filter, Alpha N-Bayes threshold training, CART decision tree filter and Support Vector machine (SVM) filter. The first level filter, stratified filter uses blacklist method, where it maintains two lists of URLs, blacklist with all malicious URLs and white list with all benign URLs. This model filters the URL by traversing through both lists. The second stage filter uses an N-Bayes threshold training model, which works on the basis of Naive Bayesian classification algorithm. Once the model gets the Bayesian probability beyond threshold, the respective URL will be classified. The third stage filter uses CART decision tree filter, where the model will be trained with URLs samples. Similar to Bayesian filter, a threshold is set based on which the leaf node takes decision and put the URL either in malicious or normal group. The last fourth filter of the model uses Support Vector Machine (SVM) filter. In this stage of filtering, the SVM algorithm derives the classification of the URL from upper layer filter. This works along with Bayesian classifier and CART decision tree classifier and determines the classification [19].

For the demonstration of the multi-layer filtering method, 10000 samples of malicious and normal URLs each are collected. From the big dataset, features are extracted based on the rules such as special characters in domain name (#, $, @, ~, \_, -), popular domain extensions (com, en, net, org, cc), number of full stops (.) in URL name, URL text length, length of the longest URL name segment and meaningful words in the main URL domain name. After extracting these features from the 20000 URLs dataset, a machine learning model is trained with different sample datasets using Bayesian, CART decision tree and Support Vector Machine (SVM) classifier models separately. Using these three classifiers another machine learning model is built and this multi layered filtering model is trained with the big URL dataset by setting reasonable thresholds and the results produced by this model are compared with other machine learning classifiers. The demonstration of this experiment proves conveys that CART decision tree yields better results among all the three classifier techniques. Also, it summarizes that the multilayer filtering model which is formed with the combination of Bayesian classifier, CART decision tree classifier and Support Vector Machine classifier given more accurate and reliable results in classification of malicious URLs and benign URLs when compared to the individual classifiers.

From some research and observations, we have found that deep neural network such as LSTM (Long short-term memory) efficiently classifies the target values for other CTR application [21]. Therefore, to efficiently detect the malicious URLs and to reduce the false negative and false positive values from the predictions we are using deep neural network LSTM and to prove its concept practically, we are also implementing different machine learning models and measuring the performance using the classification metrics such as precision, recall and F1-score.

# **Methodology**

Hackers implants the viruses and trojan. Where, URL is the main source of distribution for performing harmful activities. Therefore, detection of malicious URL becomes an important step. In this section we are proposing a framework, with different machine learning and deep learning approach to accurately classify the malicious or legit URL. The proposed framework consists of various steps which include data cleaning, data normalization, data sampling and many other methods. We will discuss each of the step, shown in Figure 3.

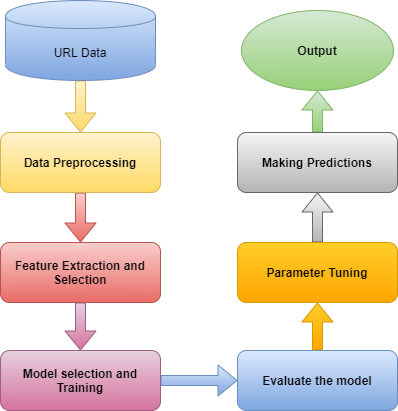


Figure 3: Proposed Framework for URL detection

3.1 URL Dataset: The dataset has been collected from UCI machine learning repository [20]. The dataset contains 2.4 million URLs. The target values in the dataset is -1 and +1. Where the -1 symbol is used for the malicious URL and +1 value represent the benign URL. Due to which it becomes a binary classification problem. To maintain the data privacy, the attributed of URL data are anonymized. The 2.4 million URLs of ICML-09 have been collected in the 120 days of time. URL dataset does not contain the content, inside the webpage.

3.2 Feature Extraction & Selection: There are 3.2 million features associated with the URL reputation data. The features are either lexical features or host-based features. Bag of words, number of dots, length of URL, domain name, hostname are some of lexical properties of the URL. Whereas, connection speed, WHOIS information, Membership in blacklists, Location and other DNS related properties are the part of host-based features. Total URL reputation dataset contains the 62% lexical features. Whereas the remaining 38% are host-based features. The dataset contains the day wise URL data. Each day file has approximately 20,000 URLs. The dataset has been collected over the period of 120 days. The following table shows the properties of lexical and host-based features.

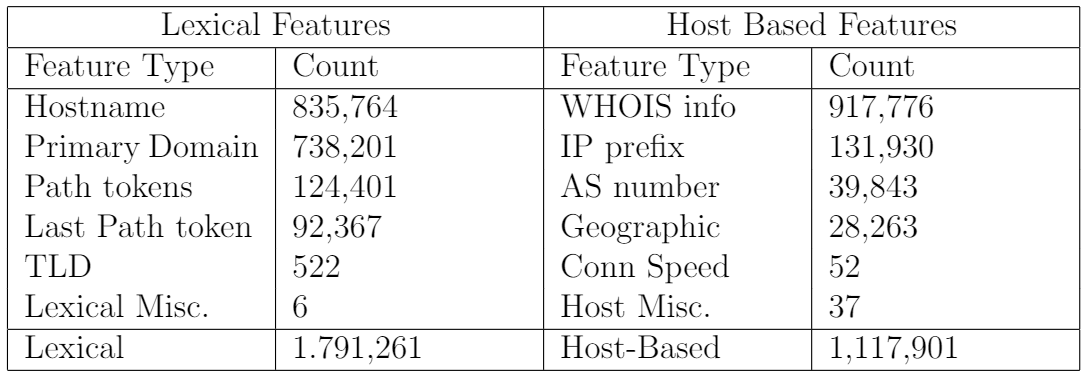


Table 1 : Feature Description of Dataset

3.3 Model Selection and Training: There are various supervised machine learning models, which can classify the URLs as malicious or benign. In order to select the best model for our classification. We have trained the URL reputation dataset over the different machine learning and deep learning classifier. The machine learning based selected algorithms are Random forest, logistic regression, support vector machine and naïve bayes. Whereas, to implement the deep learning model, we are using LSTM (Long short-term memory) neural network algorithm. A subset of data has been extracted and divided in ratio of training and testing set. We have utilized the 80% of URL data for training and remaining 20% of data for testing purposes.

3.4 Evaluation and Comparison:As the detection of malicious URL is a binary classification problem, therefore are various performance metrics which can utilized in order to know the overall performance of the model. Confusion matrix is highly used for evaluating the performance of classification model on test data. The number of correctly classified and incorrectly classified values are summarized using confusion matrix. It provided a deep insight about the type of error made by the classifier. There are four major terms True positive, True negative, false positive and false negative used for confusion matrix. Based on these values, the various score such as precision, recall, f1-score can be calculated. Accuracy is another important metrics, represents the overall performance of the classifier. For deep learning model, we are even calculating the loss, in making predictions by LSTM neural network. The evaluation of model is an iterative approach, as there are various parameters associated with machine learning and deep learning approach. Every time we perform the parameter tuning, in order to achieve the best results from the model.

# **Implementation**

As detection of malicious is a binary classification task. Various classification algorithms have been implemented for achieving better results. All our experiments have been performed in a single machine. The machine is running over windows operating system with installation of latest version of python3. Some of the python libraries have also been utilized which includes matplotlib, numpy, scipy, scikit-learn. For applying deep learning LSTM neural network, keras APIs and libraries have been utilized. For effective handling of dataset pandas framework have been utilized. The following system specifications are required to run the implemented code.

* Operating system : Windows
* Main Memory : 8GB (DDR4)
* GPU : Ryzen 5 2400G
* Hard disk : 30GB
* Programming Language : Python
* Libraries used : Keras, pandas, numpy, matplotlib and scikit-learn

We are using the jupyter notebook as IDE for real time visualization of results. The dataset is stored in the local disk of system.

# **Results and Discussion**

As described the methodology section, we have utilized 5 different models which includes the Random forest, logistic regression, support vector machine, naïve bayes and LSTM neural network. We will train the dataset for each model and calculate their respective accuracy, precision, recall and f1-score. For machine learning models we can compare the training time for each individual algorithm. Whereas, in the case of deep learning the training time of model depends on the number of epochs. Higher epoch value has higher training time. For LSTM neural network, additionally we have also calculated the validation loss. Many authors believe that reduction of loss value on every epoch, indirectly increase the accuracy of the model. The loss graph for LSTM neural network is shown in Figure 4. The loss obtained while training the data is called train loss. Whereas the validation loss is calculated over the test data. On every epoch the training and validation loss are decreasing. The lowest validation loss measured for 50 number of epochs is 0.0174.

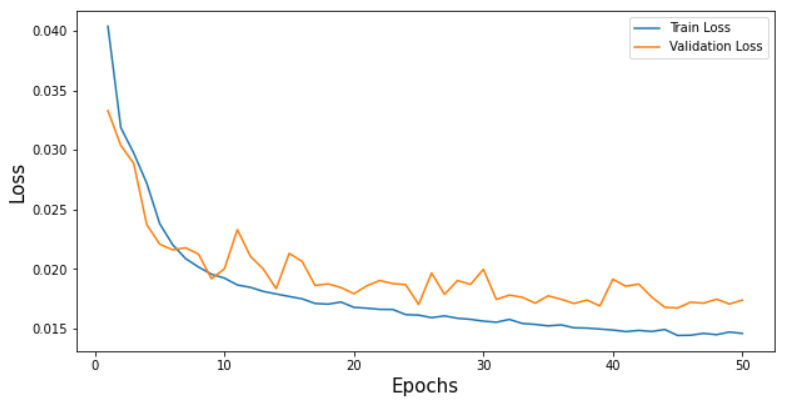


Figure 4: Graph of Validation loss for LSTM neural network

For any classification problem, accuracy is an important metrics. Which informs about the overall performance of the model. We will mainly focus on the validation accuracy, which informs about the performance of model over the test data. As increase in the number of epochs we have observed increase in accuracy. The maximum validation accuracy achieved by LSTM neural network for 50 epochs is 97.82%. The accuracy graph of LSTM neural network is shown in Figure 5. For further discussion about the PRF score, We have plotted a comparison graph which represents the PRF score of every machine learning and deep learning model proposed in this research. The precision score is inversely proportional to false positive rate. Whereas the recall score is inversely proportional to false negative rate. A bar graph shown in Figure 6, represents a comparative analysis of PRF score for LSTM neural network, logistic regression, random forest, naïve bayes and support vector machine. It has been observed that naïve bayes algorithm achieves the lowest precision, recall and f1score value and the highest PRF score has been achieved using the LSTM neural network. LSTM model, which is an deep learning algorithm outperforms than all other machine learning models in terms of precision, recall and f1score. While among the machine learning classifiers Support vector machine achieves the highest precision, recall and F1score of 0.96.

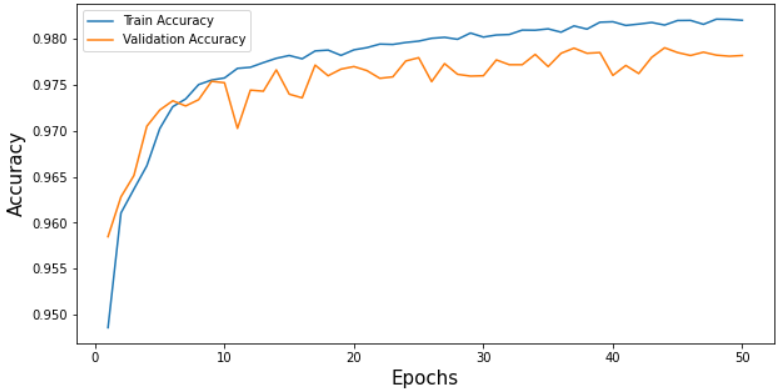


Figure 5: Accuracy Graph of LSTM neural network

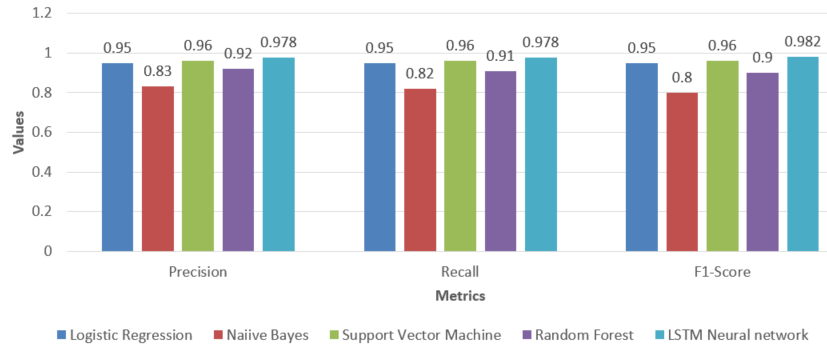


Figure 5: PRF score comparison graph

On further analysis, in terms of accuracy we have observed that LSTM again provides the significantly better accuracy as compared to other models. A highest accuracy of 97.82% has been achieved using LSTM neural network. Whereas the accuracy score for SVM, Logistic regression, random forest and naïve bayes are 96.43, 95.92, 91.18 and 82.07. All the analysis clearly indicates that LSTM neural network has the lowest false positive and false negative value and its should have high number of true positive and true negative values. The activation function plays an important role in performance measure of LSTM model. In LSTM neural network, the activation function is used ‘ReLu’ which helps the LSTM model to achieve the highest precision, recall, f1score and accuracy as compared to other models. Our proposed LSTM model consists of 5 layers. Where the 2 layers are input and output layers and remaining 3 layers are hidden layers. Hidden layers decides the complexity of model.

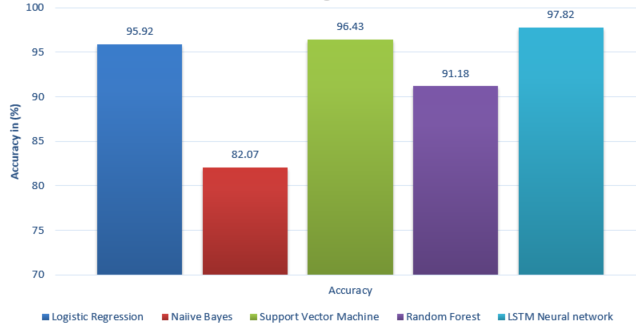


Figure 6: Accuracy Comparison between different algorithms

After calculation of classification performance metrics, we have also calculated the training time of machine learning models. For LSTM neural network, training time cannot be calculated because the training time highly depends on the number of epochs. Therefore, neglecting the LSTM model, we have compared the training time of SVM, naïve bayes, logistic regression and random forest algorithm. A comparison graph has been plotted, which represents the training time of different machine learning algorithms in Figure 7. From the graph it can be observed that naïve bayes takes the least time to train the model and highest time is required by SVM to train the model. The training time required by support vector machine is 1000 times higher as compared to other algorithms. Whereas, the time taken by random forest and logistic regression algorithm is 10.43 seconds and 3.9 seconds. Among the machine learning models, the logistic regression algorithm can be considered as the one of the best model In terms of accuracy. Although it has less precision, recall and f1-score as compared to SVM classifier. The training time taken by SVM is very high therefore, it is not much suitable algorithm to train the large number of URL samples.

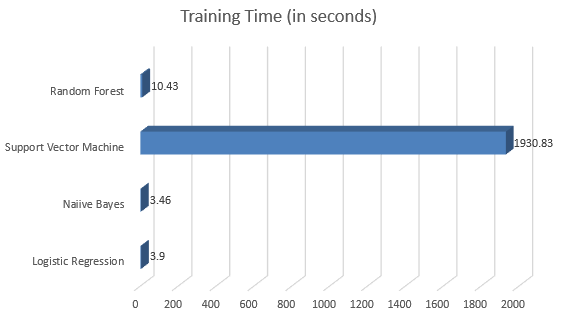


Figure 7: Training time comparison between different algorithms

After performing all the experiments, over the URL dataset we can reach to a conclusion that deep neural network are the best classifiers for detecting malicious URLs. For large number of input samples SVM classifier provides the fair accuracy and PRF score but due to high training time, it cannot be preferred as the best model. Naïve bayes poorly performs as compared to all other models the accuracy achieved by naïve bayes is 82.07%, which is comparatively very low. Deep learning approach using LSTM neural network is an effective approach for accurate classification of URLs, it does not provide the highest accuracy but also provides the low false positive and false negative values, which indirectly increases the precision, recall and f1-score.

# **Conclusion and Future work**

After implementing all the machine learning and deep learning model. We can conclude that LSTM neural network is an efficient solution for classifying the URL as malicious or benign. Using the LSTM neural network we achieved the highest accuracy of 97.82% over test data. In the next place among the machine learning algorithms logistic regression outperforms in terms of accuracy. SVM algorithm also provides the fair accuracy but time taken by SVM for training is very high. The reason behind the high training time of SVM is the number of support vectors. High number of support vectors among the data increases the complexity of the model, which indirectly leads to higher training time. Therefore, support vector machine is not a preferable option for large dataset. As the millions of URLs are generated every day, storing the all the such information will take significantly high computing power and storage capacity and for this relying on single machine/server is not a optimal solution. In this regard for production use, we need a scalable solution which can be achieved using the big data technologies. The framework such as spark is very efficient to process the larger dataset in cluster environment. In future work, more number of url samples can be collected and can be experimented in big data environment using Hadoop and spark technologies.

# **References**

[1] “The 2020 Official Annual Cybercrime Report,” *Herjavec Group*. https://www.herjavecgroup.com/the-2019-official-annual-cybercrime-report/ (accessed Jun. 29, 2020).

[2] B. Liang, J. Huang, F. Liu, D. Wang, D. Dong, and Z. Liang, “Malicious Web Pages Detection Based on Abnormal Visibility Recognition,” in *2009 International Conference on E-Business and Information System Security*, May 2009, pp. 1–5, doi: 10.1109/EBISS.2009.5138008.

[3] T. Kraska, A. Talwalkar, and J. Duchi, “MLbase: A Distributed Machine-learning System,” p. 7.

[4] F. J. Provost and D. N. Hennessy, “Scaling Up: Distributed Machine Learning with Cooperation,” p. 6.

[5] F. Provost and V. Kolluri, “A Survey of Methods for Scaling Up Inductive Algorithms,” *Data Min. Knowl. Discov.*, vol. 3, pp. 131–169, 1999.

[6] F. J. Provost and V. Kolluri, *A Survey of Methods for Scaling Up Inductive Learning Algorithms*. 1997.

[7] “Spark 101: What Is It, What It Does, and Why It Matters | MapR.” https://mapr.com/blog/spark-101-what-it-what-it-does-and-why-it-matters/ (accessed Jun. 29, 2020).

[8] M. Zaharia, M. Chowdhury, M. J. Franklin, S. Shenker, and I. Stoica, “Spark: Cluster Computing with Working Sets,” p. 7.

[9] G. Biau, “Analysis of a Random Forests Model,” p. 33.

[10] F. Vanhoenshoven, G. Nápoles, R. Falcon, K. Vanhoof, and M. Köppen, “Detecting malicious URLs using machine learning techniques,” in *2016 IEEE Symposium Series on Computational Intelligence (SSCI)*, Dec. 2016, pp. 1–8, doi: 10.1109/SSCI.2016.7850079.

[11] I. Cohen, F. G. Cozman, N. Sebe, M. C. Cirelo, and T. S. Huang, “Semisupervised learning of classifiers: theory, algorithms, and their application to human-computer interaction,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 26, no. 12, pp. 1553–1566, Dec. 2004, doi: 10.1109/TPAMI.2004.127.

[12] G. Tan, P. Zhang, Q. Liu, X. Liu, C. Zhu, and F. Dou, “Adaptive Malicious URL Detection: Learning in the Presence of Concept Drifts,” in *2018 17th IEEE International Conference On Trust, Security And Privacy In Computing And Communications/ 12th IEEE International Conference On Big Data Science And Engineering (TrustCom/BigDataSE)*, Aug. 2018, pp. 737–743, doi: 10.1109/TrustCom/BigDataSE.2018.00107.

[13] W. Yang, W. Zuo, and B. Cui, “Detecting Malicious URLs via a Keyword-Based Convolutional Gated-Recurrent-Unit Neural Network,” *IEEE Access*, vol. 7, pp. 29891–29900, 2019, doi: 10.1109/ACCESS.2019.2895751.

[14] M.-S. Lin, C.-Y. Chiu, Y.-J. Lee, and H.-K. Pao, “Malicious URL filtering — A big data application,” in *2013 IEEE International Conference on Big Data*, Oct. 2013, pp. 589–596, doi: 10.1109/BigData.2013.6691627.

[15] J. Ma, L. K. Saul, S. Savage, and G. M. Voelker, “Identifying suspicious URLs: an application of large-scale online learning,” in *Proceedings of the 26th Annual International Conference on Machine Learning - ICML ’09*, Montreal, Quebec, Canada, 2009, pp. 1–8, doi: 10.1145/1553374.1553462.

[16] N. S. Gawale and N. N. Patil, “Real Time Detection System for Malicious URLs,” in *2014 International Conference on Computational Intelligence and Communication Networks*, Nov. 2014, pp. 856–860, doi: 10.1109/CICN.2014.181.

[17] A. S. Manjeri, K. R., A. M.N.V., and P. C. Nair, “A Machine Learning Approach for Detecting Malicious Websites using URL Features,” in *2019 3rd International conference on Electronics, Communication and Aerospace Technology (ICECA)*, Jun. 2019, pp. 555–561, doi: 10.1109/ICECA.2019.8821879.

[18] R. George, R. Jalal, R. M. Raju, S. S. Sunny, and M. Hari, “High Responsive Plug-In for Malicious URL Detection,” in *2019 3rd International Conference on Trends in Electronics and Informatics (ICOEI)*, Apr. 2019, pp. 357–359, doi: 10.1109/ICOEI.2019.8862664.

[19] R. Kumar, X. Zhang, H. A. Tariq, and R. U. Khan, “Malicious URL detection using multi-layer filtering model,” in *2017 14th International Computer Conference on Wavelet Active Media Technology and Information Processing (ICCWAMTIP)*, Dec. 2017, pp. 97–100, doi: 10.1109/ICCWAMTIP.2017.8301457.

[20] Archive.ics.uci.edu. 2020. *UCI Machine Learning Repository: URL Reputation Data Set*.

[21] Chen, Qiao-Hong & Yu, Shi-Min & Guo, Zi-Xuan & Jia, Yu-Bo. (2016). Estimating Ads’ Click through Rate with Recurrent Neural Network. ITM Web of Conferences. 7. 04001. 10.1051/itmconf/20160704001.