IJARCCE

ISSN (Online) 2278-1021
ISSN (Print) 2319 5940

International Journal of Advanced Research in Computer and Communication Engineering
ISO 3297:2007 Certified
Vol. 6, Issue 6, June 2017

A Study of Phishing URL Detection using Apriori and FP-Tree algorithm

Vaishali Jaiswal[1], Dr. Pramod Nair[2]

Research Scholar, Department of CSE, MITM, Indore, M.P., India[1]

Professor, Department of CSE, MITM, Indore, M.P., India[2]

Abstract: The internet becomes a primary need or the part of daily life. A significant amount of growth of internet users is observed in recent years. In the similar ratio the internet frauds and phishing cases are also observed. In order to prevent these issues the user awareness is required. In addition of that various anti-phishing tools are also used to identify the phishing cases. There are a number of techniques available for detection of phishing URLs and prevent them to open in the user's browser. In this presented work the data mining technique based phishing URL detection technique is investigated. The data mining approaches are always first evaluating the historical data to recover the patterns to recognize and then the concluded knowledge is used for application. In the similar manner the phish tank data is used to recover the patterns and the mixed data is used for classification performance demonstration. In the proposed work first using the phishing URLs the significance of the URL is estimated. This phase is termed as a feature computation phase. After finding the features the entire URLs are encoded on the basis of these features and a transactional database is prepared. After this the association rule mining algorithms are applied to the dataset. In this experiment the Apriori algorithm and FP-Tree algorithm is used for computing the association rules. These association rules are further utilized for detection of phishing URLs. The implementation of this technique is performed on JAVA technology. After implementation the experimental results with increasing amount of data are performed. The result shows the increasing amount of data for classification impact on the performance of both the algorithms. But the FP-Tree algorithm provides efficient and accurate results as compared to the Apriori algorithm.

Keywords: phishing detection, URL analysis, FP-Tree, Apriori Algorithm, Data mining technique.

I. INTRODUCTION

Use of the computer network or internet for the transmission of data is growing rapidly. But more use of internet severe kind of attack may steal our personal information or any kind of information which flows through it, due to which the security can break. With due to rapid increase in the use of internet technology for communication different kind of attacks can be possible on the network such as DOS (denial of service attack), masquerade, replay and phishing, etc. It is one of the most serious attacks which steals our personal information or hack the website [1]. Social networking sites, such as Facebook, have become part of everyday use. While many individuals and organizations use SNSs to maintain contact and to do a variety of services, attackers may see social networking sites as a prime target for deceiving users, attacking their organizations, or performing different types of attacks, Phishing is a mechanism that tantamount to a crime employing both social engineering and technical subterfuge to steal consumers' personal identity data and financial account credentials [2]. With the advancement of the internet and anti-phishing techniques, the number and types of phishing attacks are also on the rise. The phishing attacks, too, have become more sophisticated.
In this research work, we present a knowledge framework for detecting URL phishing websites. Our study contributes to professional practice by providing a theoretical foundation for developing algorithms that predict users' susceptibility to phishing victimization using data mining techniques.

II. PROPOSED WORK

The section provides the understanding about the solution developed in this work for classifying the phishing URLs. Therefore the different phases of the process involved in the solution are described.

A. System Overview

The Phishing is an act of steeling the confidential data of others for wrong intentions. That is an act of serious cybercrime. Every year a number of phishing attacks are deployed over the internet. In most of cases the users are not aware about the phishing or respectively less time expended on the internet. The attacker either sends phishing URLs by using the emails or mobile phones. If the user provides the details about the email or message, then the data are used for harming the targeted user. On the other side a significant amount of effort for preventing such kind of frauds are

IJARCCE
ISSN (Online) 2278-1021
ISSN (Print) 2319 5940
International Journal of Advanced Research in Computer and Communication Engineering
ISO 3297:2007 Certified
Vol. 6, Issue 6, June 2017

available and developed every year. But still either these methods are not much efficient or inaccurate in nature. Therefore, in this work for detection of phishing URLs a machine learning based approach is proposed for study. The machine learning techniques enable the algorithms to analyses the historical data patterns and compute the features by which the similar behavior data can be recognized.
Therefore, in this work the data mining technique is used for recognizing the phishing URL patterns. The concept is to analyses and extracts the features from the phishing URLs as training of the algorithm and then utilizes these extracted features to identify the phishing URLs [3]. The investigation of the work involves the study of phish tank datasets. In further the Pre-processing techniques are studied for refining the data to utilizing with the algorithms, additionally the different features or heuristics for evaluation of the URLs. In next process the encoding of the data is performed for making its use with mining algorithms. Finally the two association rule mining algorithms are used for finding the associative patterns in the data. These associative rules are used with the data for classifying the URLs in two classes namely legitimate or phishing. This section provides the overview of the proposed data model for study. In the next section the methodology of work is described.

B. Methodology
The proposed data model for phishing URL detection is given using figure 1. In this diagram the different system components are organized to process the data and forward in the next phase of analysis.

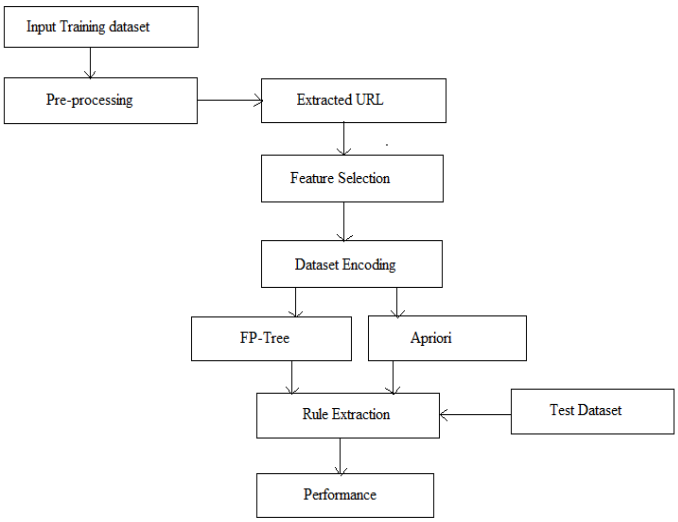

Figure 1 methodology

Input training dataset: As described initially the concept first evaluates the patterns of phishing URLs and use the computed knowledge for identifying the similar kinds of patterns. Therefore the phish tank database is used for evaluation and training purpose. Initially the dataset contains the following fields:
1. **Phish ID:** that denotes the ID assigned by the phish tank database for recognizing the phishing attack.
2. **URL:** that demonstrates the URL which is used to deploy the phishing attack.
3. **Phish detail URL:** the details about the phishing URL is available online in this link.
4. **Submission date:** the date of reporting this URL as phishing is given in this attribute
5. **Verification time:** the time when it is verified as the phishing URL
6. **Online:** that shows the detection type of URL
7. **Target:** the target company or organization for deploying the attack

Initially, all the data is read by the system and represented as the initial dataset.

Pre-processing: pre-processing is a technique for filtering the unwanted data produced as input to the system. Sometimes this technique is used for improving the quality of data by reducing the amount of noise, handling missing values and other. In this phase the data is pre-processed for extraction of the URL data which is reported and verified as phishing URL.

Feature selection: the extracted data from the previous phase is used in here for computing the features of the data. The feature computation is required to evaluate each URL in the training set over the following heuristics.
1. Length of the host URL

IJARCCE

ISSN (Online) 2278-1021
ISSN (Print) 2319 5940

International Journal of Advanced Research in Computer and Communication Engineering
ISO 3297:2007 Certified
Vol. 6, Issue 6, June 2017

2. Number of slashes in URL
3. Dots in host name of the URL
4. Number of terms in the host name of the URL
5. Special characters
6. IP address
7. Unicode in URL
8. Transport layer security
9. Subdomain
10. Certain keyword in the URL
11. Top level domain
12. Number of dots in the path of the URL
13. Hyphen in the host name of the URL
14. URL length

On the basis of these listed features the individual URLs are evaluated and for each of them a value is computed. The computed values are demonstrated using the table 1.

Table 1 example feature computation

URL	1	2	3	4	5	6	7	8	9	10	11	12	13	14
URL1	28	2	3	1	4	5	2	5	2	1	5	7	7	1

Dataset encoding: now the encoding of the dataset is performed. In this context each feature value is associated with some threshold value if the feature value is less then threshold is it assumed the URL feature is legitimate and if higher then then considered as the phishing URL. The associated threshold values of each feature are given in table 2.

Table 2 feature threshold

Feature	Threshold	Phishing	Legitimate
Length of the host URL	25	1	0
Number of slashes in URL	5	1	0
Dots in host name of the URL	4	1	0
Number of terms in the host name of the URL	4	1	0
Special characters	Yes	1	0
IP address	Yes	1	0
Unicode in URL	Yes	1	0
Transport layer security	Http(Yes)	1	0
Subdomain	Yes	1	0
Certain keyword in the URL	Yes	1	0
Top level domain	No	1	0
Number of dots in the path of the URL	2	1	0
Hyphen in the host name of the URL	1	1	0
URL length	75	1	0

According to the values obtained by analysing the URLs are used to encode the dataset in terms of binary strings.

FP-Tree: after construction of final encoded dataset the data is used with the FP-tree and Apriori algorithm. The FP growth algorithm is described in this section. The major steps of FP growth are consisting of the following steps:

Step1- First condenses the database showing frequent item set into a FP - tree.

Step2: It decomposes the FP-tree into a no. of conditional, database and mines each database independently, thus extract frequent item sets from FP-tree exactly. It contains a single root labelled as null, a set of transaction set prefix sub trees as the children of the root, and a frequent item header table. Every node in the item prefix sub tree made up of three domains: name of item, count and node link where-name of item registers which item the node appear as; registers of counts the many of transactions demonstrate by the portion of path arriving this node, node link connects to the next node in the FP- tree. Each item in the header table made up of two fields---item name and head of node link, which points to the first node in the FP-tree compassionate the item name" [4].

IJARCCE

ISSN (Online) 2278-1021
ISSN (Print) 2319 5940

International Journal of Advanced Research in Computer and Communication Engineering
ISO 3297:2007 Certified
Vol. 6, Issue 6, June 2017

Table 3 FP-Growth algorithm

Input: constructed FP-tree Output: set of frequent patterns
Process: 1. If FP Tree accommodate a unique path P then 2. For individual combination do generate patterns β. α with support = minimum support of nodes in β. 3. Else For each header an in the header of the tree does 4. Generate pattern β = AI α with support = AI. Support 5. Evaluate β. S conditional pattern base and then β. S conditional FP-tree Tree β 6. If Tree β = null 7. Then call FP-growth (Tree β, β)

Apriori: During the literature survey, we found a number of frequent set mining algorithms is implemented. Additionally, many other sequences and frequent pattern mining techniques are generated. The Apriori Algorithm is an influential algorithm for mining frequent item-sets of Boolean association rules. The key terminology of the Apriori algorithm is given as follows [5]:

- **Frequent Item-sets:** The group of items which has minimum support (denoted by L_i for it-Item-set).
- **Apriori Property:** Any subset of frequent item-set must be frequent.
- **Join Operation:** To find L_k, a set of candidate k-item-sets are implemented by joining $L_{k-1}$with itself.

Find the frequent item-sets: the sets of items that have minimum support– A subset of a frequent item-set must also be a frequent item-set [9]

- I.e., if {AB} is a frequent item-set, both {A} and {B} should be a frequent item-set
– Iteratively find frequent item-sets with cardinality from 1 to k (k-item-set)
- Use the frequent item-sets to generate association rules.
- Join Step: $C_k$is generated by joining $L_{k-1}$with itself.
- Prune Step: Any (k-1)-item-set that is not frequent cannot be a subset of a frequent k-item-set

Table 4 Pseudocode Apriori Algorithm

Variables: C_k: Candidate item-set of size k L_k: frequent item-set of size k L_1= {frequent items};

Process:
For (k = 1; L_k!=∅; k++) do begin
C_{k+1}= candidates generated from L_k;
For each transaction to in the database do
Increment the count of all candidates in C_{k+1}
Those are contained in t
L_{k+1}= candidates in $C_{k+1}$with min_support
End
Return$\cup_k L_k$;

Rule extraction: both the implemented algorithms are extracted the association rules on the basis of the input item sets and their corresponding values. These rules are used for classifying the data in terms of legitimate and phishing URLs.

Test dataset: initially the data is used for training contains only the phishing URLs. Now a new set of data is prepared by using the mixing up both kinds of patterns phishing URLs and the well-known legitimate URLs.

Performance: each the testing set URL is evaluated using the given association rules and according to the obtained patterns the classification of the URL is performed and their performance is computed.

C. Proposed algorithm

This section summarizes the defined model which is required to implement. In order to summarize the process the algorithm is defined using table 5.

IJARCCE

ISSN (Online) 2278-1021
ISSN (Print) 2319 5940

International Journal of Advanced Research in Computer and Communication Engineering
ISO 3297:2007 Certified
Vol. 6, Issue 6, June 2017

Table 5 proposed algorithm.

Input: training set T_r, testing sets T_s Output: Classified Data C
Process: 1. $R = readTrainingDataset(T_r)$ 2. $P = PreProcessData(R)$ 3. $for(i = 1; i \leq P.length; i++)$ a. $H_i^P = ComputeHuristic(P_i)$ b. $E = encode(P_i, H_i^P)$ 4. end for 5. $A_{rule} = Apriori.FindRules(E)$ 6. $F_{rule} = FPTree.FindRules(E)$ 7. $C = A_{rule}.Classify(T_s)$ 8. $C = F_{rule}.Classify(T_s)$ 9. return C

III. RESULTS ANALYSIS

This chapter provides the evaluation of the performance for both kinds of algorithm, namely FP-growth and Apriori. The obtained performance of the algorithms is compared on different parameters.

A. Time complexity

The algorithm requires a time for performing the computation, this amount of time requirements is termed as the time complexity of algorithm. the time consumption of algorithm can be computed by finding the difference among the algorithm initialization time and process completion time.

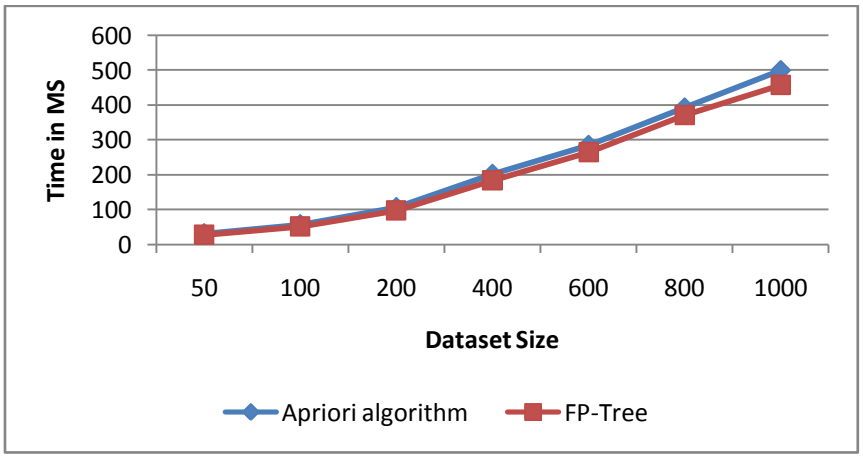

Figure 2 time complexity

The time complexity of both the algorithms (i.e. Apriori and FP-Tree) is denoted in table 6 and figure 2. In order to represent the performance of algorithms the X axis contains the increasing amount of data for evaluation and the Y axis contains the time required to complete the process. Here the time complexity is computed in terms of milliseconds. According to the obtained results the time of both the algorithms with increasing amount of data is increases in similar ratio. Additionally the FP-Tree requires less time to compute the classes of URLs in terms of phishing or legitimate as compared to Apriori algorithm.

Table 6 time complexity

Dataset Size	Apriori algorithm	FP-Tree
50	32	28
100	58	51
200	107	98
400	202	184
600	285	264
800	392	371
1000	499	457

IJARCCE

ISSN (Online) 2278-1021
ISSN (Print) 2319 5940

International Journal of Advanced Research in Computer and Communication Engineering
ISO 3297:2007 Certified
Vol. 6, Issue 6, June 2017

B. Space Complexity

The space complexity of the algorithm demonstrates the amount of main memory space required to compute the outcomes by any algorithm.

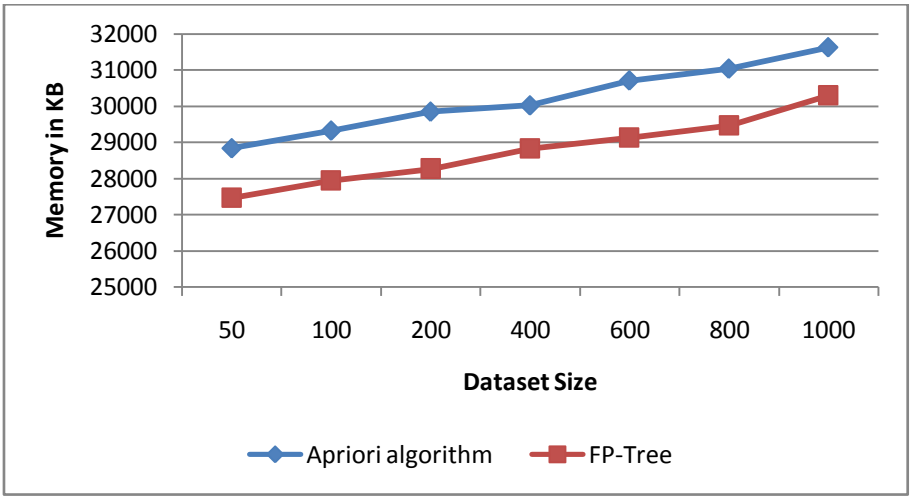

Figure 3 space complexity

Table 7 space complexity

Dataset Size	Apriori algorithm	FP-Tree
50	28847	27474
100	29334	27948
200	29856	28275
400	30028	28837
600	30716	29137
800	31042	29471
1000	31631	30299

The space complexity of algorithms is described using figure 3 and table 7. The space complexity of algorithms is described in Y axis, which is computed in form of KB (kilobytes). Additionally the amount of data supplied for experimentation is denoted in X axis. According to the obtained results the Apriori algorithm requires large amounts of memory as compared to the FP-Tree algorithm. The reason behind large resource consumption is that because the Apriori algorithm initially generates the candidate-sets and places them in main memory for further utilization. In addition of that FP-Tree generates the frequent item sets and the volume of frequent item sets are less as compared to the candidate set.

C. Accuracy

The accuracy is the measurement of the algorithm's correctness of recognitions. That can be computed by finding the ratio between the correctly classified data and the total data samples are produced for classifying. To compute the accuracy of the algorithm the following formula can be used:

$$\text{accuracy} = \frac{\text{Total correctly classified}}{\text{total input for classify}} \text{X100}$$

Table 8 accuracy

Dataset Size	Apriori algorithm	FP-Tree
50	98	98
100	96	97
200	95.6	96.5
400	94.8	96.2
600	93.9	94.4
800	92.3	93.8
1000	91.6	92.3

IJARCCE

ISSN (Online) 2278-1021
ISSN (Print) 2319 5940

International Journal of Advanced Research in Computer and Communication Engineering
ISO 3297:2007 Certified
Vol. 6, Issue 6, June 2017

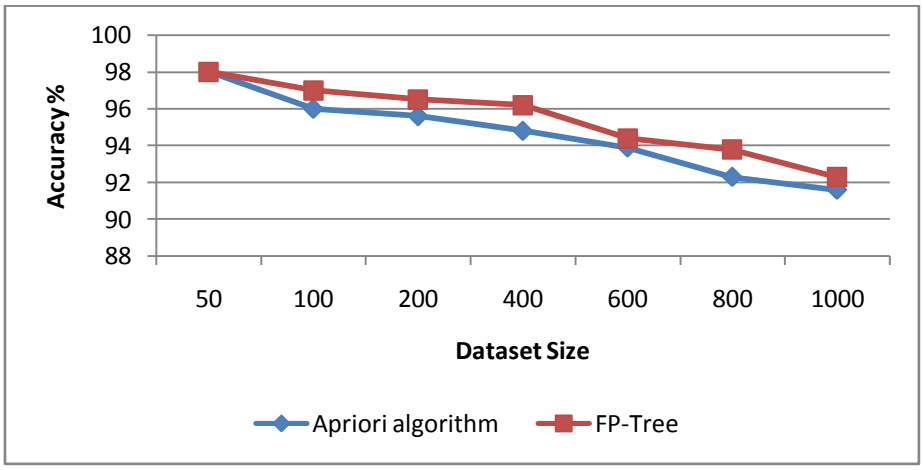

Figure 4 accuracy

The performance of Apriori algorithm and FP-Tree for phishing URL detection in terms of percentage accuracy is given using figure 4 and table 8. In this diagram the X axis contains the amount of dataset instances used for classifying the phishing patterns and the Y axis shows the amount of data correctly recognized by the algorithms. According to the experimental results both the algorithms initially provide similar accuracy but as the amount of data increases the difference in performance is clearly observed. The results show the accuracy of the FP-Tree increases as the amount of patterns increases for classification.

D. Error rate

The error rate is the percentage amount of misclassified data over the total samples provided for classification. The error rate of the algorithm can be measured using the following formula:

$$\text{error rate} = \frac{\text{incorrectly classified data}}{\text{total data input}} \text{X100 Or error rate} = 100 - \text{accuracy}$$

Table 9 error rate

Dataset Size	Apriori algorithm	FP-Tree
50	2	2
100	4	3
200	4.4	3.5
400	5.2	3.8
600	6.1	5.6
800	7.7	6.2
1000	8.4	7.7

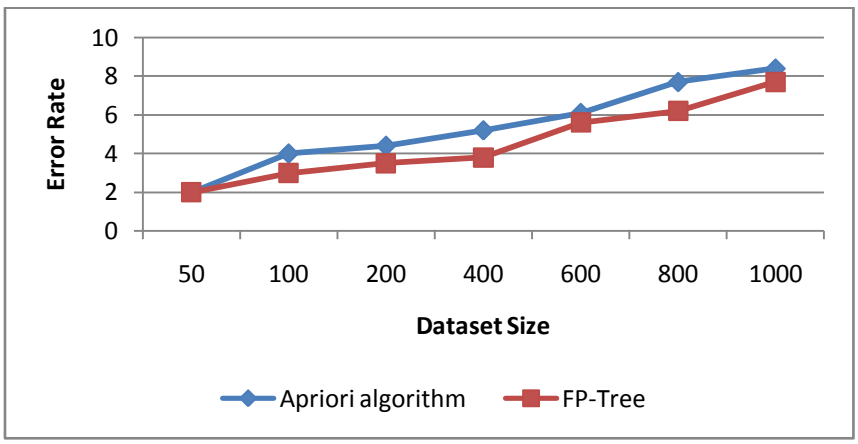

Figure 5 error rate

The error rate for both the algorithms, namely Apriori algorithm and FP-growth algorithm for classifying the phishing URLs are given using figure 5 and table 9. The table includes the error rate values and the graph includes the lines for

representing the performance values. The X axis of data contains the amount of data instances used for experimentation and the Y axis shows the corresponding obtained an error rate produced by algorithms. According to the experimental results the FP-Tree algorithm produces the least amount of error rate as compared to the Apriori algorithm.

IV. CONCLUSION

The key aim of the proposed work is to find the most efficient and accurate technique for phishing URL classification is achieved successfully. This chapter draws the conclusion of the entire work performed; additionally the future extension of the work is also included.

A. Conclusion

As the internet growing the significant amount of traffic is increasing on the internet. Not all the users of the internet know about the forgery or phishing over the internet, some new users easily trapped in the phishing issues. In order to prevent the phishing a number of techniques available among some of them are developed with the black and the white list concept, some of them are usages the certificate and cryptographic approaches. But all these algorithms have their own limitations. The machine learning based technique can be helped to analyze the phishing URLs more efficiently and accurately by estimating the features from data URLs. These features are created by the observations of the phishing URL patterns. The proposed work includes the implementation of two different algorithms for classifying the phishing URLs. The phishing URLs are extracted from the phish tank database. This database contains the reported phishing URLs by different organization and institutions. Not all the data is required for learning and detection purpose. Therefore, only the URLs are extracted from the phish tank dataset. This process is termed as the preprocessing of data. To next process the URLs available are used with the 14 given heuristics. Using these heuristics and pre-approximated thresholds the data are encoded in the form of binary strings. In next process the Apriori algorithm and FP-Tree algorithm is used for processing the data. This phase returns the association rules from the available data. These rules are used as “if then else” rules for classifying the datasets, thus it is used by the testing set for evaluation of the performance of the techniques.

The implementation of the proposed study of phishing URL detection using FP-Tree and Apriori algorithm is performed using JAVA technology. After implementation the evaluation of both the algorithms is performed for finding their efficiency and accuracy. The table 10 includes the evaluated parameters and obtained experimental values.

Table 10 performance summary

S. No.	Parameters	FP-Tree	Apriori
1	Time complexity	27-457 MS	31-499 MS
2	Space complexity	27200-30500 KB	28300-31800 KB
3	Accuracy	92-98%	91-98%
4	Error rate	2-8%	2-9%

The given table demonstrates the obtained experimental outcomes of both the algorithm. According to the results the FP-Tree implementation is much acceptable as compared to the Apriori algorithm due to less time and memory resource consumption additionally produces higher accurate results.

B. Future work

The experimental results demonstrate the effectiveness of the both models for classification of phishing URLs. In the near future the following extension of the proposed technique is feasible.

1. Implement the given concept using some kinds of web browser toolbar for detection of real time phishing URLs
2. Need to enhance the system for obtaining more accurate results
3. Apply the performance improvement techniques for improving the current performance of URL detection or classification such as bagging and boosting.

REFERENCES

[1] Geer, David. "Security technologies go Phishing." Computer 38.6 (2005): PP. 18-21.
[2] Anti-Phishing Working Group. Phishing Activity Trends Report, Third Quarter 2013. URL: http://antiphishing.org/resources/apwg-reports/ last Accessed: April 2016G.
[3] Jeeva, S. Carolin, and Elijah Blessing Rajsingh. "Intelligent phishing url detection using association rule mining." Human-centric Computing and Information Sciences 6.1 (2016): PP. 1-19.
[4] Abdullah Saad Almalaise Alghamdi, “Efficient Implementation of FP Growth Algorithm-Data Mining on Medical Data”, IJCSNS International Journal of Computer Science and Network Security, VOL.11 No.12, December 2011
[5] SONALI SONKUSARE, JAYESH SURANA, “Implementation And Comparative Study Of Improvedapriori Algorithm For Association Pattern Mining ”, International Journal of Advanced Computational Engineering and Networking, Volume-4, Issue-8, Aug.-2016.