DOI: http://dx.doi.org/10.26483/ijarcs.v8i7.4141

ISSN No. 0976-5697

Volume 8, No. 7, July – August 2017

International Journal of Advanced Research in Computer Science

REVIEW PAPER

Available Online at www.ijarcs.info

CLASSIFICATION OF MALICIOUS URLS FOR WEB USING RIPPER ALGORITHM

Er. Gugneet Kaur
Research Scholar
Dept. of Computer Engineering,Punjabi University
Patiala,Punjab

Dr. Jaswinder Singh
Assistant Professor
Dept. of Computer Engineering,Punjabi University
Patiala,Punjab

Abstract: Now a days Web data is the most discussed topic. In various fields related to internet produces data of thousands of gigabytes every minute. Various applications uses multimedia data sharing procedure. So data will automatically be of bulk amount. This bulk amount of data is hard to process, takes longer time of search this much large data. RIPPER (Repeated Incremental Pruning to Produce Error Reduction) is one of the Classification rule algorithm.

Keywords: Ripper, Web Data

1. INTRODUCTION

1.1 Web Data

Web related data is the application of specialized tools through which large amount of data will be processed. This data other wise will be very difficult to process without the automated tool.

The amount of data generated in the different mediums is enormous. Various social media sites which are producing the data of large nature. This type of data requires large amount of data processing abilities. So that after analysis the data can be represented in graphical way. This graphically represented data will helps in having better and fast data point of view. So that system understanding regarding the system will be better.

As we know the data produced will be enormous. This data belongs unstructured category. Because data produced in different mediums like audio, videos, text etc. this type of data is produced in billions of bytes every hour. Once this whole data will be produced and stored at the server. Now requires various levels of processing. So that system of understanding regarding the data can be developed. This data requires various levels of processing. Structuring etc[1].

1.2 Malicious Urls

URLs are the main culprit for any web attacks. Such that any malicious intention user can steal the identity of the legal person by sending the malicious URL. There requires authenticity system which can authentify the URL. So that only legal URLs are allowed to enter in. BLADE is system which is to authentify the URLs[4]. It simply downloads the contents and checks the authenticity of the contents. Analyze how much time it has taken to download the what is the download time. But contents based detection is not the base for identifying the attack. As new URLs are being produced every hour. It proposes the content based description to identify the malicious nodes. So that list of malicious and legitimate URL can be identified. Those URLs which fails the conditions will be put into the malicious list. And those which pass the contents description will be put into the legitimate list of URLs[2].

1.3 Web data URLs Challenges

i. Large scale: several million URLs are being produced every hour.
ii. Extremely imbalanced data set: The list of Malicious URLs are in very small amount compared to the total no. of URLs. It is 0.01% of total URLs list.

1.4 Ripper

RIPPER is one of the classification rule algorithm. It basically extracts the rules directly from the data. this algorithm progresses through the given four phases: Growth phase, pruning, optimization, selection. In first phase that is growth phase first rule is generated and various attributes are added incrementally till certain stopping criteria arises. Each rule is incrementally pruned for any final sequence of the attributes. this procedure will goes on till the final step is achieved. finally those attributes are selected which are best suitable for the situation. Ripper is a rule based learner that build a set of rules that identify the classes while minimizing the amount of error[3].

The RIPPER algorithm builds a single rule in the following steps:

- Split the dataset with growing and pruning set.
- In growth phase, starts the things with empty set.
- Add the new rule and also provide gain criteria.
- Repeat step 3 till negative example or dataset is not found.
- Prune the new rule(attribute) based on new prune rules.

In a multi-class situation, the rules generated from the RIPPER algorithm are ranked in ascending order based on the number of examples in the class.

The RIPPER algorithm for multi-class classification is described in the following steps:

i. Ripper arrange the class based on ascending or descending order.

ii. It identifies the short class as positive class and long class as negative class.
iii. Only positive class for rules is to identified.
iv. Repeat the steps 2 and 3 until short class finding stops.

1.4.1 Problem in ripper algorithm

i. As there will be growth in the knowledge of the attributes. This over knowledge will generates the over fitting of the rules , which may leads to the misclassification.
ii. The major disadvantage is the noisy data. This noisy data can leads to mis classification.
iii. The major drawback of RIPPER is the over fitting of the rules. Such that wrong justification is performed at.

In RIPPER algorithm the normalization and balancing follows the common procedure. The rules developed are based on training dataset. the ruleset covers the rules based on various attributes[5].

i. The algorithm is designed to be fast and accurate. so that the improved proficiency is shown such that detecting malicious URLs can be identified.
ii. If the rule set length is more and attributes are less then activity is performed using loop. The RIPPER algorithm with normalization is fast and effective way of doing the activity..
iii. Each rule's attributes are checked against the initial seven rules. then aggregation of the rules is taken place. Only those rules are selected which are based on high rank value[12].

1.5.1 Malicious URL

URLs have now a days become a way to hack the resources belings to other. Attacker using malicious URLs distributes the malicous programs all around. **Kaspersky La. b** Author has reported that the browser bsed attacks has grown substantially. URLs are the main culprit for any web attacks. Such that any malicious intention user can steal the identity of the legal person by sending the malicious URL. There requires authenticity system which can authentify the URL. So that only legal URLs are allowed to enter in. BLADE is system which is to authentify the URLs[15]. It simply downloads the contents and checks the authenticity of the contents. Analyze how much time it has taken to download the what is the download time. But contents based detection is not the base for identifying the attack. As new URLs are being produced every hour. It proposes the content based description to identify the malicious nodes. So that list of malicious and legitimate URL can be identified[7]. Those URLs which fails the conditions will be put into the malicious list. And those which pass the contents description will be put into the legitimate list of URLs.

Ripper Algorithm

RIPPER algorithm builds a single rule in the following steps:

- Split the dataset with growing and pruning set.
- In growth phase, starts the things with empty set.
- Add the new rule and also provide gain criteria.
- Repeat step 3 till negative example or dataset is not found.
- Prune the new rule(attribute) based on new prune rules.

In a multi-class situation, the rules generated from the RIPPER algorithm are ranked in ascending order based on the number of examples in the class.

The RIPPER algorithm for multi-class classification is described in the following steps. Ripper arrange the class based on ascending or descending order. It identifies the short class as positive class and long class as negative class. Only positive class for rules is to identified. Repeat the steps 2 and 3 until short class finding stops. (Pan & Ding, 2006).

Analysis:RIPPER (JRip) is a direct method i.e. is often used to extract rules directly from data. In WEKA tool RIPPER is implemented as JRip, generates rules set after theevaluation over the Training dataset. This rules set is the classifier model for JRip algorithm which can further be used to predicting the unknown URLs. Here, the output rules set of used to predict the data of the testing set after which all the parameters listed in table 5.2 is calculated[13]. Figure 5.1 shows the ruleset generated by the RIPPER algorithm. There are a total of 25 attributes and RIPPER algorithm make rulesets using these attributes. The rules are: -

Table 1: Rulesets of RIPPER Algorithm

Rule 1: (Favicon=yes)^(SSL_final_state=yes) → Legitimate
Rule 2: (Favicon=yes)^(having_host_name=yes) Legitimate
Rule 3: (Page_Rank=2)^(Favicon=yes) ^ (URL_Length=56)→Legitimate
Rule 4: (double_slash_redirecting=yes)^(folder_name=no) →Legitimate
Rule 5: (URL_Length=55)^(Favicon=yes) →Legitimate
Rule 6: (Favicon=yes)^(URL_L ength=54)→Legitimate
Rule 7: Otherwise→Malicious

A rule-based is a technique for classifying record using a collection of "if…then…"rules. Table 4.3 ensure that every records is covered by exactly one rule.

i. The first rule is interpreted asif a URL have the value yes for both favicon and SSL final state then the result shows that it is a legitimate URL.
ii. The second rule is interpreted as if a URL have the value yes for favicon and en for having_host_name then the result shows that it is a legitimate URL.
iii. The third rule is interpreted as if a URL have the value yes for favicon and have the value 2 for Page_Rank and also have the value 56 for URL_Length then the result shows that it is legitimate.
iv. The fourth rule interpreted as if a URL have the value yes for double_slash_redirecting and have the value no for folder_name then the result shows that it is legitimate URL.
v. The fifth rule interpreted as if URL have the value yes for favicon and have the value 55 for URL_Length then the result shows that it is legitimate URL.
vi. The sixth rule interpreted as if URL have the value yes for favicon and have the value 54 for URL_Length then the result shows that it is legitimate URL[6].
vii. If all the previous rules are not satisfied by the URL of dataset then it will go to seventh rule which interpret that URL is malicious.

According to RIPPER algorithm, it is clear from the confusion matrix of the true positive rate of this algorithms proportion of examples which were classified the last rule, among all examples which truly have rules, i.e., how much of the rules was captured correctly (the number of malicious executable examples classified as malicious executables). True Negatives rate is proportion of examples which were classified above mentioned six rules was capture correctly the number of legitimate URLs classified as legitimate[8]. False positive are those URLs which are actually legitimate but predicted malicious. False Negatives are those URLs which are actually malicious but predicted legitimate. So after each and every URLs data is checked with these rule sets the total number of True positive, true negative, false positive, false negative are calculated. After that accuracy of URLs is calculated from number of true positive and true negative by total number of URLs data. Error rate of URLs is calculated from number of false positive and false negative by total number of URLs data. Precision of the URLs is calculated from the number of exactly classified instance of a target URL, i.e., positive URL, over the number of instance classified as view to that URLs. It is also known as positive predicted value. Recall of the URLs is calculated from the number of exactly classified instance of a URL, i.e., positive URL, over the number of instance of that URL. The F-measure of URLs is calculated from the compromise between recall and precision[9].

1.6 Conclusion

Web data have various challenges related to security like-computation in distributed programming, security of data storage. For tackling with such security challenges we used different security methods like Type Based keyword search for security of Web data, use of hybrid cloud to provide privacy in Web data. Various techniques have been implemented in order to control the malicious attacks[14]. Different tools and software are there to determine such sites. Most of the browsers are built with phishing alert functionality for these cases. Another functionality of Blacklisting has come out to be a promising approach in past but with its dynamic nature of malicious URLs demanding more and more efficient methods. Different systems such as Phish Tank and Wiktionary are provided in order to determine URLs that are malicious and pose threat to the users in real time. Data mining techniques are utilized in order to detect such malicious URLs on a regular basis. Data mining methods use algorithms that First extract the features of the suspected site and check it with the provided classifier. Classifiers are the rules generated using data mining algorithms for determining the legitimate from the illegitimate malicious ones. In this research work there is use of JRip i.e. Ripper algorithm[10].

REFERENCES

[1] Due, B., Kristiansen, M., Colomo-Palacios, R., & Hien, D. H. T. Introducing Web Data Topics: A Multicourse Experience Report from Norway. Proceedings of the 3rd International Conference on Technological Ecosystems for Enhancing Multiculturality, 2015, 8(2), 565–569.

[2] Shirudkar, K., &Motwani, D. Web-Data Security, 2015, 5(3), 1100–1109.

[3] J. Ma, L. Saul, S. Savage and G. Voelker, “Learning to Detect Malicious URLs”, ACM Transactions on Intelligent Systems and Technology, (2011), 1(1), 30:1-30:24.

[4] H. S. Choi, B. B. Zhu and H. J. Lee, “Detecting Malicious Web Links and Identifying Their Attack Types”, Proceedings of the 2nd USENIX Conference on Web application development (WebApps), USENIX Association Berkeley, (2011), 1(3), 1-12.

[5] B. Eshete, A. Villafiorita and K. Weldemariam, “BINSPECT: Holistic Analysis and Detection of Malicious Web Pages”, Proceedings of the 8th International ICST Conference, SecureComm,(2012), 3(6), 1544-1562.

[6] W. Tao, S. Z. Yu and B. L. Xie, “A Novel Framework for Learning to Detect Malicious Web Pages”, Proceedings of the International Forum on Information Technology and Applications (IFITA), (2010), 1(9), 212-220.

[7] W. Zhang, Y. X. Ding, Y. Tang and B. Zhao, “Malicious web page detection based on online Learning algorithm”, Proceedings of the International Conference on Machine Learning, (2011), 17(6), 1914-1919.

[8] V. L. Le, I. Welch, X. Y. Gao and P. Komisarczuk, “Two-Stage Classification Model to Detect Malicious Web Pages”, Proceedings of the International Conference on Advanced Information Networking and Application (AINA), (2011), 15(11), 113-120.

[9] M. Cova, C. Kruegel and G. Vigna, “Detection and Analysis of Drive-by-Download Attacks and Malicious JavaScript Code", Proceedings of the International World Wide Web Conference Committee (IW3C2), WWW, (2010), 44(1), 48-58.

[10] Y. H. Choi, T. G. Kim and S. J. Choi, “Automatic Detection for JavaScript Obfuscation Attacks in Web Pages through String Pattern Analysis”, International Journal of Security and Its Applications, (2010), 22(7), 13-26.

[11] R. B. Basnet and A. H. Sung, “Classifying Phishing Emails Using Confidence-Weighted Linear

[12] Classifiers”, Proceedings of the International Conference on Information Security and Artificial Intelligence (ISAI), (2010), 4(3), 108-112.

[13] R. B. Basnet and A. H. Sung, “Learning to Detect Phishing Webpages”, Journal of Internet Services and Information Security (JISIS), (2014), 4(1), 21-39.

[14] K. Rieck, T. Krueger and A. Dewald, “Cujo: Efficient Detection and Prevention of Drive-by-

[15] DownloadAttacks”, Proceedings of the 26th Annual Computer Security Applications Conference (ACSAC), (2010), 3(5), 31-39.