Fuzzy Inference for Intrusion Detection of Web Robots in Computer Networks

Mahdieh Zabihi, Majid Vafaei Jahan, Javad Hamidzadeh

M.Sc. Student of Computer Engineering (Software), Imam Reza International University, Mashhad, Iran.

Department of Computer Engineering, Azad University, Mashhad, Iran.

Department of Computer Engineering, Sadjad University, Mashhad, Iran.

Abstract

Distinction between humans and web robots, in terms of computer network security, has led to the robot detection problem. An exact solution for this issue can preserve web sites from the intrusion of malicious robots and increase the performance of web servers by prioritizing human users. In this article, we propose a novel method called FID (Fuzzy Intrusion Detection), for accurate intrusion detection of web robots, in computer networks. The proposed method relies on fuzzy inference systems and uses a decision tree to fuzzify the features which are used to describe web clients. The aim of FID is to overcome the curse of dimensionality, reduce the number of fuzzy rules and subsequently, facilitate the designing of fuzzy inference system. Our experiments show that the proposed algorithm has 0.13 false alarm ratios, in distinguishing humans from robots. Experimental results, on a real data set, demonstrate the effectiveness of our approach when compared to some of the state-of-the-art algorithms.

Keywords: web robots, fuzzy inference system, decision tree, web server access log file, web usage mining.

1. Introduction

Internet, one of the most important technologies in these days, is a massive information repository and a new medium for communication and collaboration. Undoubtedly, to manage and update these repositories and gain some knowledge from this information, appropriate solutions are needed. Web robots, as one of these solutions, send requests to web servers and analyze the results received to fulfill their specific purposes. Well-behaved or non-malicious robots, with useful purposes such as collecting the statistics about the web structure, site maintenance and checking for broken hyperlinks, are active researchers in the World Wide Web. Unlike these autonomous systems, some robots with malicious purposes, such as click fraud, harvesting Email addresses, collecting business intelligence and DDoS[1] attacks, threaten the security of web servers. A similarity between malicious and non-malicious web robots is occupying the network bandwidth and reducing the performance of web servers. Indifference, Failure to follow instructions on how to design a robot and changing its characteristics in order to imitate the human's behavior, are some obstacles for web robot detection.

Our proposed algorithm, namely FID (Fuzzy Intrusion Detection), is a novel fuzzy approach towards features used to describe web visitors. This algorithm utilizes a decision tree to fuzzify the attributes (features), reduce the number of fuzzy rules and facilitate the designing of the fuzzy inference system. Given the importance of proper and relative attributes/features, correlation analysis is applied on each attribute, in order to choose more appropriate

[1] Distributed Denial of services

features. Our previous study (Rajabnia et al. 2013) is also based on the fuzzy inference system; but now, we use a more comprehensive data set in our experiments.

The remainder of the paper is organized as follows: In Section 2, a survey of web robot detection is presented. In section 3, the proposed method for web robot detection is introduced. In Section 4, the experimental results of the proposed method are shown, and in Section 5, a comparison between FID and some of the state-of-the-art algorithms is considered. Eventually, Section 6 finalizes our conclusions and remarks.

2. Related works

Previous studies considered in this paper, are some of the analytical learning techniques which are more accurate than other offline techniques used for web robot detection (Doran et al. 2010). In one of the first such studies (Tan et al. 2002), the authors use a new approach to extract web sessions and introduce 25 new features to distinguish humans from robots and utilize C4.5 decision tree to classify them. They apply their method on an academic access log collected over a period of month in year 2001 and achieve very accurate web robot detection.

In 2005, Bomhardt et al. develop a web log pre-processing tool called RDT and use neural network and decision tree to detect web robots. They use two log files to evaluate their method, one from an educational web site and the other from an online shop (Boomhardt et al. 2005).

In (Stassopoulou et al. 2009), the authors present a Bayesian approach to crawler detection and declare a dynamic threshold for session identification. Also, they compare their results to the results obtained with the decision tree and achieve very high recall and precision values in web robot detection.

As previously mentioned; in (Rajabnia et al. 2013), we used the fuzzy inference system to classify web users; but the data set was approximately small in size. So, the study presented in this paper utilizes a broader data set collected over a period of a month in year 2013, came from the web server at Imam Reza International University[2].

3. The Proposed method

In this section, we present the proposed method for intrusion detection of web robots, FID, and describe each step of the flowchart of this algorithm shown in Figure 1.

In step 1, session identification, two consecutive HTTP requests that have same IP addresses or same user agent strings, will be in a same session; if the time-lapse between them is within a pre-defined threshold (30 minute in the majority of web-related literatures).

After session identification, all features listed in table 1, are specified for each extracted session to identify and distinguish web robots from human users (step 2).

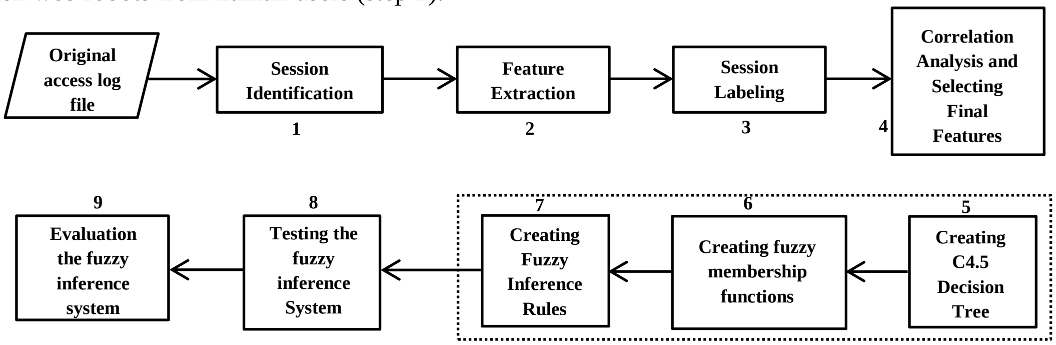

Original access log file
Session Identification
1
Feature Extraction
2
Session Labeling
3
4
Correlation Analysis and Selecting Final Features
5
Creating C4.5 Decision Tree
6
Creating fuzzy membership functions
7
Creating Fuzzy Inference Rules
8
Testing the fuzzy inference System
9
Evaluation the fuzzy inference system

Figure 1. Flowchart of the proposed method

[2] www.imamreza.ac.ir

In the third step, each session is labeled as belonging to one of the following 2 categories: human visitors, web robots. This phase is performed by observing whether the respective user has attempted to access the robot.txt file. Moreover, any session whose user agent string (or IP address) matches a known web robot, is labeled as web robot. All other web sessions are labeled as human visitors.

Table 1. Features extracted for each session in second phase

Remark	Feature name	Remark	Feature name
The volume of data transmitted in a session.	Volume	Session length.	Length
% of consecutive sequential HTTP requests.	% CSRequests	Total number of requests.	Total Requests
% of requests with status=404.	Error 404	Whether this file is requested or not.	robots.txt
Maximum depth of all requests.	Max Depth	Ratio of switching the type of files requested.	Switching factor of file type
% of PDF/PSS files requested.	%PDF/PSS	% of requests with unassigned referrer.	Null Referrer
Number of HTML page requests over the number of image files requests.	HTML-to-Image ratio	% of CSS files requested.	%CSS
Standard deviation of requested page’s depth.	stdevDepth	% of requested made between 12 am to 7 am.	Night

In the next step, all specified features are filtered according to a correlation analysis, in order to eliminate all attributes that correlate rarely with robots or humans. Undoubtedly, having more features for each session is equivalent to have more information about it. But, increasing the number of features can complicate the designing of fuzzy inference system; because, the depth of the decision tree and subsequently, the number of inference rules extracted from this tree are increased.

Table 2 shows the values of correlation analysis for each attribute. The negative value shows that the considered attribute is often more for humans than robots and vice versa. It is obvious that a small correlation value (zero or near zero) shows the Insufficiency of a feature to distinguish robots and humans from each other (Tan et al. 2002).

Table 2. Correlation analysis for each extracted feature

Feature name	Correlation coefficient
Volume	-0.011
% CSRequests	-0.12
stdevDepth	-0.28
Error 404	0.14
Total Requests	-0.02
%PDF/PSS	0.07
Night	0.33
robots.txt	0.46
Length	0.11
%CSS	-0.24
Switching factor of file type	-0.12
Null Referrer	0.46
Max Depth	-0.29
HTML-to-Image ratio	0.04

In FID algorithm, all features whose analysis values are in $(-\infty,-1] \cup [1,+\infty)$ are chosen as the final attributes. Indeed, step 4 helps to solve the curse of dimensionality problem. Although, step 5, explained in the following, also helps this problem to be solved. As explained, 10 attributes are chosen and used for creating a C4.5 decision tree to convert each feature to a fuzzy variable in step 5.

Creating a decision tree is based on choosing a feature with maximum information gain in each level of the tree (Hand et al. 2001; Han et al. 2011). So, the depth of the tree and subsequently, the number of features and inference rules extracted from this tree are reduced. Figure 3 shows the C4.5 decision tree made, according to features selected in step 4. As it is clear; only four attributes are selected and used for making the decisions.

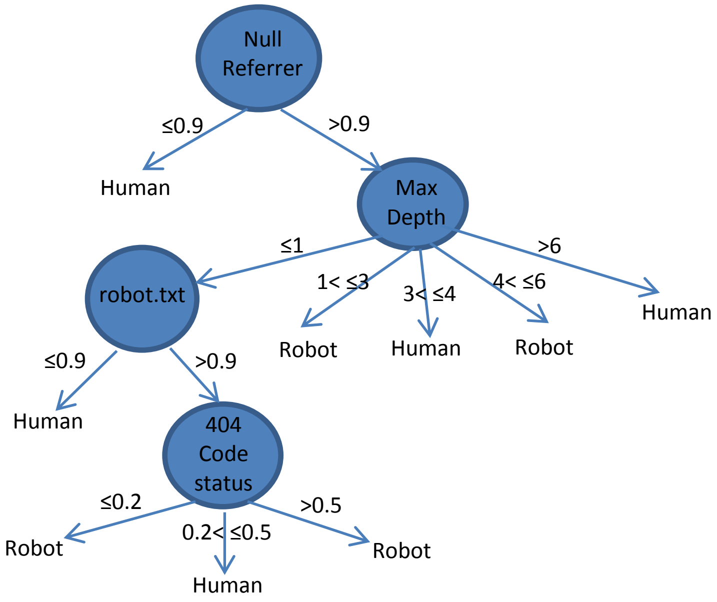
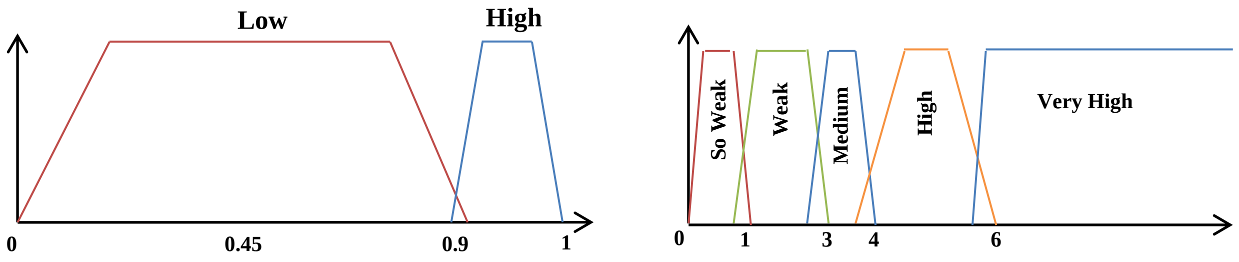


Figure 3. C4.5 decision tree made in step 5

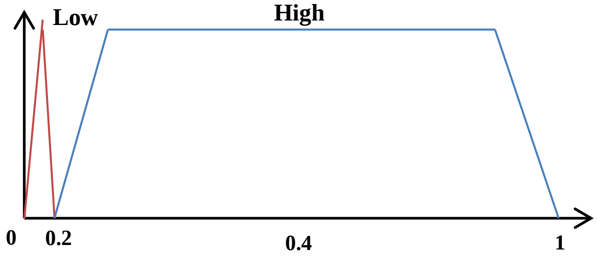
In step 6, according to C4.5 decision tree made in previous step, we draw all membership functions of the features used in this tree.

Table 3. Membership functions for all features used for making C4.5 decision tree

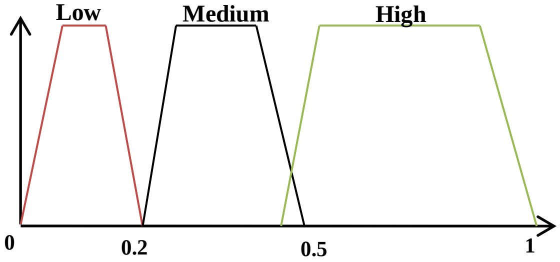


a. Membership function of Null Referrer

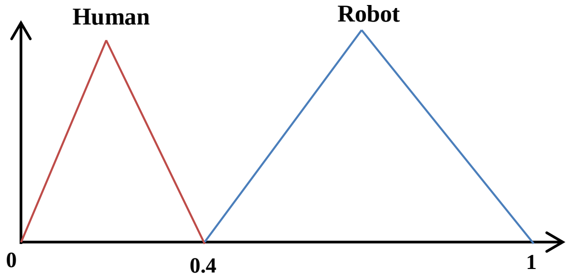
b. Membership function of Max Depth



c. Membership function of robot.txt



d. Membership function of 404 code status



e. Membership function of output

After creating a C4.5 decision tree in step 5, we extract all inference rules, by navigating from the root to the leaves of the tree, in step 6. The purpose of these consecutive steps is training the fuzzy inference model. Figure 4 illustrate all rules extracted from the C4.5 decision tree (made in step 5).

Table 3. Fuzzy rules extracted from C4.5 decision tree

If (**null Referrer is High**) & (**Max Depth is So Weak**) & (**robot.txt is High**) & (**E404 Status is Low**) then (**output is Robot)**

If (**null Referrer is High**) & (**Max Depth is So Weak**) & (**robot.txt is High**) & (**E404 Status is Medium**) then (**output is Human)**

If (**null Referrer is High**) & (**Max Depth is So Weak**) & (**robot.txt is High**) & (**E404 Status is High**) then (**output is Robot)**

If (**null Referrer is High**) & (**Max Depth is So Weak**) & (**robot.txt is Low**) then (**output is Human)**

If (**null Referrer is High**) & (**Max Depth is Weak**) then (**output is Robot)**

If (**null Referrer is High**) & (**Max Depth is Medium**) then (**output is Human)**

If (**null Referrer is High**) & (**Max Depth is High**) then (**output is Robot)**

If (**null Referrer is High**) & (**Max Depth is Very High**) then (**output is Human)**

If (**null Referrer is Low**) then (**output is Human)**

By the end of sixth step, we can test the model and evaluate its performance by some evaluation metrics used for two-class problems (Hand et al. 2001; Han et al. 2011). In this paper, k-cross validation is used to select the training and testing data sets (k=10).

4. Experiments

In the experimental stage, we use an access log file generated from the web server of Imam Reza International University, over a month-long period, from 26 October through 26 November 2013. Table 3 shows the number of sessions and class label distributions in this data set.

Table 3. The data set used in this paper

Total number of requests	Total number of sessions	Total number of robot sessions	Total number of human sessions
311633	17969	1170	16799

Table 4 lists all evaluation metrics used after testing the proposed model. $Recall^{x}$ shows the percentage of examples belonging to class x that are correctly identified; while $Precision^{x}$ is the percentage of times the classification rule is correct. It is worth pointing out that the class of robots and humans are respectively shown by '+' and '-'.

In computer networks, TPR[3] and FPR[4] are the standard metrics used to evaluate the performance of a classification model for intrusion detection. TPR shows the percentage of web robots that are correctly classified and FPR is the percentage of humans incorrectly classified which is sometimes called the false alarm ratio too. In intrusion detection problems, FPR is more important than TPR and all other metrics listed in table 2; because a small value of FPR can cause a huge false alarm rate and subsequently, reduce the performance of web servers, in most computer networks (Han et al. 2011).

According to table 4, the large difference between the number of robots and humans, cause the small value for $Precision^{+}$ and subsequently $F - measure^{+}$.

Table 4. Metrics used for evaluating the proposed model

Num	Evaluation metric	value
1	$Recall^{+} = \frac{TP}{FN + TP}$	0.97
2	$Precision^{+} = \frac{TP}{TP + FP}$	0.34
3	$F - Measure^{+} = \frac{2 \times Recall^{+} \times Precision^{+}}{Recall^{+} + Precision^{+}}$	0.50
4	$Recall^{-} = \frac{TN}{FP + TN}$	0.87
5	$Precision^{-} = \frac{TN}{TN + FN}$	0.98
6	$F - Measure^{-} = \frac{2 \times Recall^{-} \times Precision^{-}}{Recall^{-} + Precision^{-}}$	0.92
7	$TPR = \frac{TP}{FN + TP}$	0.97
8	$FPR = \frac{FP}{TN + FP}$	0.13

[3] True positive Rate
[4] False Positive Rate

5. Comparisons

In this stage, we compare the FID algorithm with some classifiers used in previous related studies. Figure 2 shows the TPR and FPR for all these classifiers. According to the results, the false alarm ratio of proposed method is lower than the others (0.13) and its accuracy in detecting the web robots is the most (TPR=0.97).

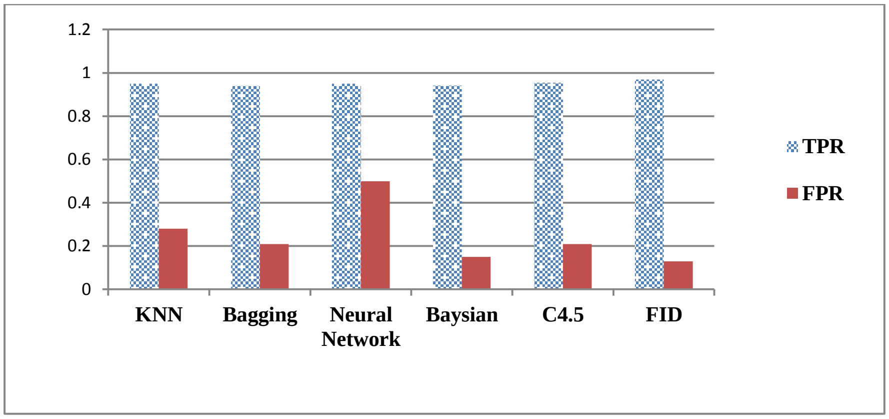


Figure 2. TPR & FPR of proposed algorithm and other classifiers

6. Conclusion and remarks

Web administrators should pay special attention and closely inspect web sessions that correspond to web robots; because the traffic of these autonomous systems occupies the bandwidth, reduces the performance of web servers and in some cases, threaten the security of human users. In this paper, we propose a novel fuzzy algorithm based on the decision trees. In order to overcome the curse of dimensionality issue and facilitate the designing of the fuzzy inference system, we use a correlation analysis to eliminate some features. For converting each filtered attribute to a fuzzy variable, a C4.5 decision tree is used. It is worth pointing out that making a decision tree is based on choosing the best feature with the most information gain metric in each level of the tree. So, we can reduce the number of attributes again. Finally, the fuzzy rules are extracted from the C4.5 decision tree and the fuzzy inference model is made.

Experimental results on a real data set demonstrate the effectiveness of our approach when compared to some of the state-of-the-art algorithms.

References

[1] J. Rajabnia, M. Zabihi, and M. Vafaei Jahan, "Web Robot Detection With Fuzzy Inference System Based on decision trees", The Seventh Iran Data Mining Conference, Tehran, 2013.

[2] D. Doran and S. S. Gokhale, "Web robot detection techniques: overview and limitations", Data Mining and Knowledge Discovery, vol. 22, pp. 183-210, 2010.

[3] P.-N. Tan and V. Kumar, "Discovery of web robot sessions based on their navigational patterns", Data Mining and Knowledge Discovery, vol. 6, pp. 9-35, 2002.

[4] C. Bomhardt, W. Gaul, and L. Schmidt-Thieme, "Web Robot detection pre-processing web logfiles for Robot Detection", New Developments in Classifiation and Data Analysis, pp. 113-124, 2005.

[5] W. Siler and J. J. Buckley, Fuzzy Expert Systems and Fuzzy Reasoning. Hoboken, New Jersey: John Wiley & Sons, 2005.

[6] P. Hayati, V. Potdar, K. Chai, and A. Talevski, "Web Spambot Detection Based on Web Navigation Behaviour", 24th IEEE International Conference on Advanced Information Networking and Applications (AINA), Perth, Western Australia,

pp. 797-803, 2010.

[7] J. Han and M. Kamber , Data Mining: Concepts and Techniques. Morgan Kaufmann, 2011.

[8] D. Doran and S. S. Gokhale, "A classification framework for web robots", Journal of the American Society for Information Science and Technology, vol. 63, no. 12, pp. 2549-2554, 2012.

[9] G. K. Kanji, 100 Statistical Tests, 3rd ed. SAGE Publication, 2006.

[10] D. Petrilis and C. Halatsis, "Two-level Clustering of Web Sites Using Self-Organizing Maps", Neural Processing Letters, vol. 27, no. 1, pp. 85-95, 2008.

[11] S. M. Ross, Introduction to Probability and Statistics for Engineers and Scientists. California: Academic Press, 2009.

[12] A. Stassopoulou and M. D. Dikaiakos, "web robot detection:a probabilistic reasoning approach", Computer Network, vol. 53, pp. 265-278, 2009.

[13] D. Stevanovic, A. An, and N. Vlajic, "Feature evaluation for web crawler detection with data mining techniques", Expert System with Application, vol. 39, pp. 8707-8717, 2012.

[14] B. S. Everitt, S. Landau, M. Leese, and D. Stahl, Cluster Analysis. Whiley Press, 2009.

[15] D. J. Hand, H. Mannila, and P. Smyth, Principles of data mining. MIT Press, 2001.

[16] S. Known, M. Oh, D. Kim, J. Lee, Y. Kim, and S. Cha, "Web Robot Detection Based on Monotonous Behaviour", Proceedings of the Information Science and Industrial Applications, 4, 2012.

[17] Losawar, M. Joshi, "Data Pre-processing in Web Usage Mining", International Conference on Artificial Intelligence and Embedded Systems (ICAIES), Singapore, 2012.

[18] X. Lin, L. Quan, H. Wu, "An Automatic Scheme to Categorize User Sessions in Modern HTTP Traffic", Global Telecommunications Conference, pp.1485-1490, 2008.

[20] V. Sumalatha, K. Ramani, K. Lakshmi, "Fuzzy Inference System to Control PC Power Failures", International Journal of Computer Applications, vol. 28, no. 4, pp. 10-17, 2011.

[21] (2014) user-agent-string.info. [Online]. http://user-agent-string.info.

[22] (2014) Bots vs Browsers. [Online]. http://www.botsvsbrowsers.com.