Detecting Impolite Crawler by using Time Series Analysis

Zhiqian Chen
Dept. of Software Engineering
Peking University
Beijing, China
imczq@pku.edu.cn

Wenya Feng
Dept. of Software Engineering
Peking University
Beijing, China
pkuwenyafeng@gmail.com

***Abstract*—Numerous web crawlers especially impolite crawlers visit websites to get contents every day, which yields higher access frequency than the websites can hold. The big traffic of impolite crawlers causes a strong hazard on analysis of normal users and advertisement income. In this paper, we present a method to detect impolite crawlers by using time series analysis. This method is applied to real data of web server logs. Compared with the old methods only using common log attributes as features, the method using time series features improves detection accuracy by at least 20%.**

Keywords-impolite crawlers; time series; web server log; data mining; web analysis

I. INTRODUCTION

A new method is introduced to detect general impolite crawlers, which helps significantly improve accuracy. We use machine learning algorithms with features extracted by using time series analysis.

The visitors are divided into three groups: normal users, impolite crawlers and polite crawlers. Polite crawlers, such as Google robots, gently get contents from websites and are easy to detect, so they will not be discussed in this paper. Impolite crawlers who quickly download resources from the target websites bring at least two types of harm. On one hand, they disturb normal web analysis. If a huge amount of traffic from impolite crawlers cannot be identified, they could be considered as normal users. Finally it will lead to a bad decision. On the other hand, impolite crawlers access websites without reading advertisements, and they will not purchase anything including the advertisement, but the advertisers have paid for it.

In this paper, a general, simple and feasible method based on common server logs that meet the criterion of W3C log format is proposed. These servers are common, such as Apache, Lighttpd, and Nginx. Crawler access and normal user access are different. Generally, the crawlers' behaviors are regular, while the normal users' behaviors are irregular. A number of features that can deliver the degree of regularity are needed. From every single piece of log, its common attributes are extracted, such as access time, access URL, referred URL, and user agent. In addition, some features are extracted by time series analysis that will show more information of a crawler and help us detect impolite crawlers more accurately.

The rest of this paper is organized as follows: the related work is reviewed in section II. The proposed method is introduced in section III. The findings by applying the proposed method to real web server logs are discussed in section IV. Finally, the conclusions and future work are presented in section V.

II. RELATED WORK

Machine learning algorithms are widely applied in detection of crawlers [1][2] by extracting many features from a single log. It's important that some priori conditions are needed in such research. Another interesting method describes all access traces in three-dimensional model [3], so crawlers can be easily detected by eyes. All these methods use common attributes of a single log as features, but there is a drawback that not all real crawlers may have these features. For example, not all crawlers get all pages of a target website [3], so a more general method is needed.

III. OUR APPROACH

A. Why Time Series Features

A single piece of log only reveals information about the characters of website visitors, while time series features gather behavioral information in a continuous time and reveal more comprehensive information about the visitors. Crawlers' behaviors show obvious regularity, while other visitors show less regularity.

In the target website, each page is labeled in a numeric format (the number only has categorical meaning), we call this number page-code for short, and it can be easy to present in time series diagram. After arranging every page-code of single user in chronological order, we get what Figure 1 shows. The x-axis represents time, while y-axis indicates the page-code. The figure shows a typical access time series of a normal user, it looks totally random, like white noise in signal processing field.

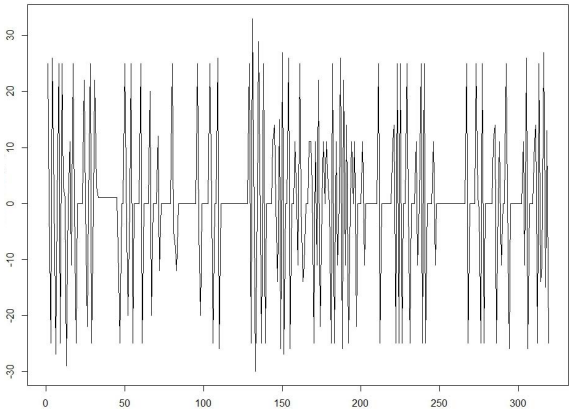

Figure 1. Real User Access Page Time Series

1082-3409/13 $31.00 © 2013 IEEE
DOI 10.1109/ICTAI.2013.28

Time series of impolite crawlers are labeled by the same method, and several typical ones is presented in Figure 2. Obviously, these diagrams looks regular, although the 3 pieces of time series have different curve regularities.

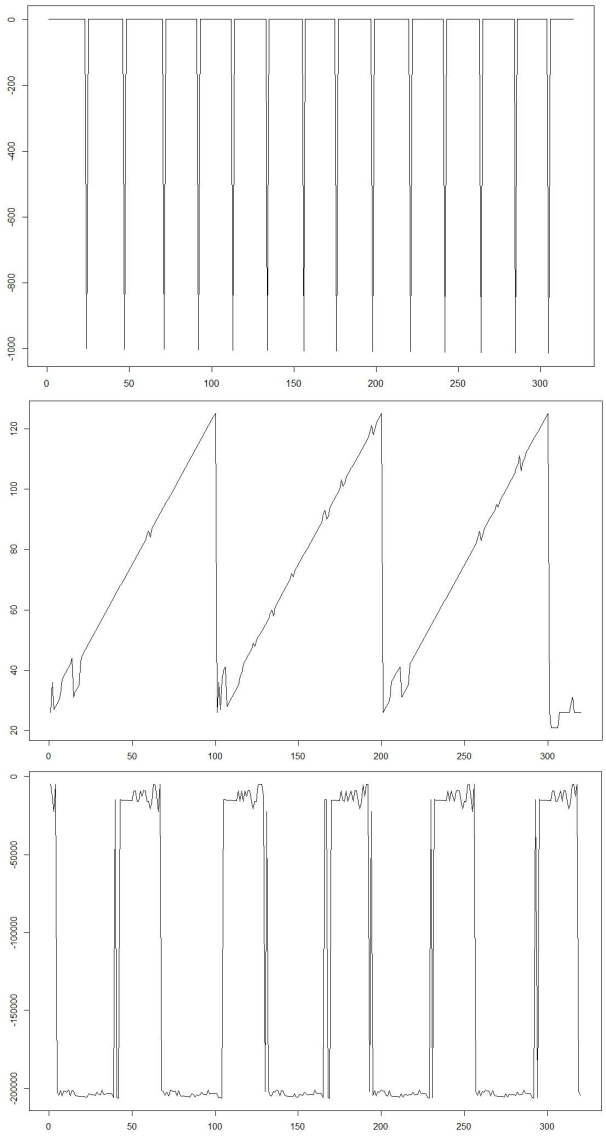

Figure 2. Impolite Crawler Access Page Time Series

Figure 1 and Figure 2 show the regularity difference between real users and impolite crawlers. A model that can describe this regularity difference is needed. In time series analysis, some effective models are used for prediction. If these models are applied to predict all above time series, the prediction accuracy of real user is supposed to be not high, because it is like white noise. That is to say, it has little regularity information and is hard to predict. On the contrary, the prediction accuracy of impolite crawlers should be much higher than that of real users, for it has more regularity information. The prediction model relies on periodicity, so it will be easier to predict.

As shown in Figure 3, the top time series shows a trace of an impolite crawler, which is also the same as the third one in Figure 2. In the bottom time series, there's a prediction of two periods (the time series surrounded by shadow) based on the data of the first three periods. Obviously, it is easy for time series model to predict crawler trace.

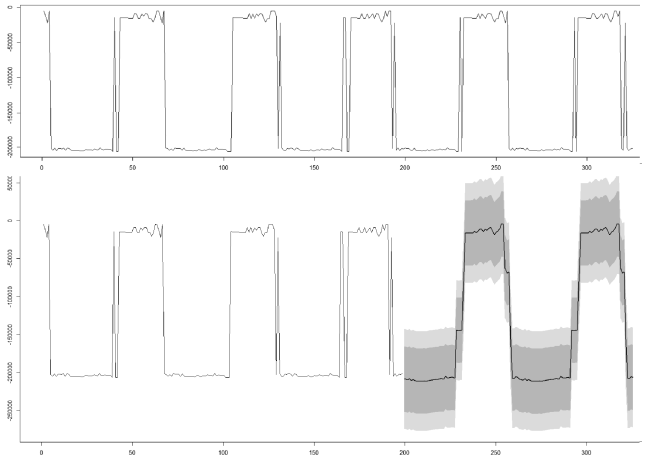

Figure 3. Prediction of Impolite Crawlers Access

There is another feature that also can distinguish the impolite crawlers and real users: first-order difference of timestamp time series.

First-order difference of timestamp time series shows the same effect as the access time series. The time series of impolite crawlers is more regular than that of real users, and is easier to predict.

When employing first-order difference to time series access data, and then making a distribution statistics, we find the two types of visitors are very different. As shown in Figure 4, the first-order difference of impolite crawler data concentrates on few kinds of numbers, while first-order difference of real user data distribute on a lot of different numbers.

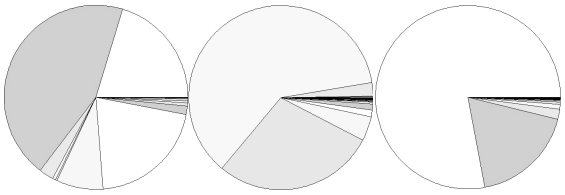

Figure 4. First-Order Difference of Impolite Crawlers Access Data

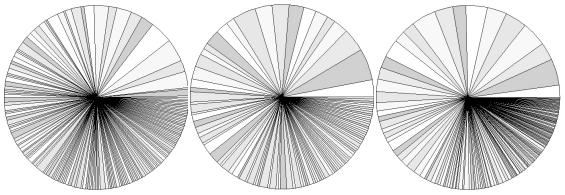

Figure 5. First-Order Difference of Real Users Access Data

B. The Two Kinds of Time Series in this paper

Two types of time series are used: access time series and first-order difference of access timestamp access. The two types of time series will generate features by using models.

C. The Time Series Model Used in this paper

Some widely used prediction models are adopted in time series analysis stage in this experiment, such as autoregressive integrated moving average (ARIMA), locally weighted scatterplot smoothing (LOESS), and Holt Winters (exponential smoothing). These models with the two types of time series are used to generate features. We will try to use the prediction results and model parameters as features.

D. The Algorithms and Tools Used in this paper

Some popular and state-of-the-art tools will be employed in this experiment. Common algorithms [4] in machine learning and data mining are used, such as decision tree, random forest, support vector machine (SVM), linear regression, and neural network. Hadoop softwares is applied to extract data from raw web server logs, because it is powerful for huge data processing, and then Linux Shell and Hadoop are used to do some Extract, Transform and Load (ETL) work. Finally, the R software [10-13] and Rattle [16-17] package which are also popular and powerful tools will be used to build the model and result chart.

IV. EXPERIMENT WITH REAL DATA

Our method is applied to a real big website with more than 200 million page views per day. This site has run into trouble with impolite crawlers, because numerous impolite crawlers access it every day. When report data fluctuates without no clear reason, it is very time-consuming and manpower-consuming to find out the specific reason from their big data including logs and databases. Detecting these impolite crawlers will help the administrator get rid of this problem.

First, we find that users grouped by access frequency are in a long tail distribution. In consideration of this, we group users by access frequency, such as 0-100 times per day, and 101-200 times per day, and extract data from every group for training, validating and testing. This sampling method will help cover all types of visitors.

This experiment will use both common log features and time series features. The same algorithms are applied in the two types of features and then their effects are compared.

In this experiment, we find it not suitable for neural network model to work in our application scenario, which is full of categorical attribute but without numerical attribute. The data of neural network model will be ignored later in this experiment.

We sample the training and testing data, and divide them into a training set and a testing set, with a specific random seed each time. After training and testing, we obtain the results as shown in Table 1. The accuracy using time series attributes (average 95.44%) is much higher than that using common attributes of logs (average 72.91%).

TABLE 1. ACCURACY COMPARSION RESULT

Random Seed	Algorithm	Common attributes Method Precision (%)		Time Series Method Precision (%)	
		Validate Set	Test Set	Validate Set	Test Set
10	Decision Tree	69.66	71.86	98.24	91.37
	Random Forest	74.55	72.12	96.49	87.93
	SVM	73.52	73.14	98.24	93.10
	Lineal Regression	72.23	72.12	100	91.37
	Neural Network	73.00	75.70	68.42	51.72
42	Decision Tree	72.49	72.89	94.73	93.10
	Random Forest	69.92	73.91	96.49	94.82
	SVM	71.46	75.44	98.24	100
	Lineal Regression	73.26	73.40	98.24	94.82
	Neural Network	73.26	75.19	64.91	65.51
600	Decision Tree	76.09	71.86	94.73	96.55
	Random Forest	74.80	68.03	94.73	93.10
	SVM	78.66	72.89	96.49	96.55
	Lineal Regression	76.09	69.56	96.49	94.82
	Neural Network	79.43	73.40	57.89	63.79

Receiver operating characteristic (ROC) curve is used for the evaluation of machine learning algorithms. The area under the ROC curve (AUC) is a measure of classifier performance, which is the area surrounded by the curve and the diagonal line, and its area size indicates the model performance, a big area size means a good performance. In comparison of AUCs between the two groups, the time series group (average 0.99) is also obviously larger than that of the common attributes group (average 0.79).

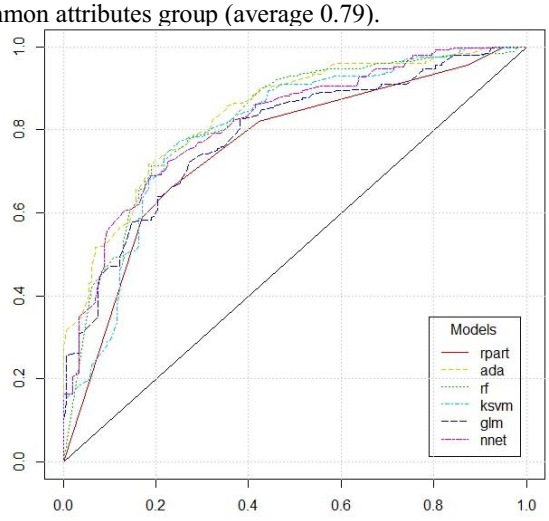

Figure 6. The AOC chart of Common Attributes Group

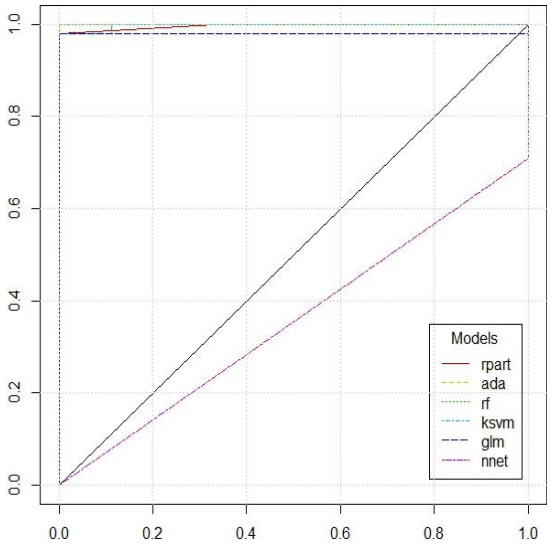

Figure 7. The AOC chart of Time Series Group

TABLE 2. THE AUC COMPARSION RESULT

Algorithms	Common attributes	Time Series
Decision Tree	0.7551	0.9965
Random Forest	0.8205	1.0000
SVM	0.7987	0.9977
Lineal Regression	0.7856	0.9792

V. CONCLUSION

This paper shows a new method using time series analysis that could detect impolite crawlers. Compared with old detecting methods, which only use common attributes extracted from every single log, the time series method could get much information. After analyzing some typical access records of real users and impolite crawlers, we find a reason for using time series features. The results also demonstrate our idea.

Two types of time series features are extracted: access time series and first-order difference of access timestamp time series. Then some popular time series models are used to generate time series features. The former detecting method is used to get common attributes, and the same machine learning algorithms are applied to the two groups of features. The time series method shows an obvious advantage.

Therefore, time series features can help detect impolite crawlers. All knowledge in time series analysis can be applied in this topic for more attempts. It is also an interesting direction for future work.

REFERENCES

[1] Marios Dikaiakos, Athena Stassopoulou, Loizos Papageorgiou, Characterizing Crawler Behavior from Web Server Access Log, 2003

[2] Marios Dikaiakos, Athena Stassopoulou, Crawler Detection : A Bayesian Approach, 2005.

[3] Jawaheer, Gawesh Kostkova, Patty.Web crawlers on a health related portal: detection, characterisation and implications.Developments in E-systems Engineering (DeSE), 2011

[4] S. B. Kotsiantis.Supervised Machine Learning: A Review of Classification Techniques. Informatica 31:249−268 (2007)

[5] Trevor Hastie, Robert Tibshirani, Jerome Friedman.The Elements of Statistical Learning. Springer, 2008.

[6] Roman Timofeev .Classification and Regression Trees (CART) Theory and Applications, 2004

[7] John Ross Quinlan.C4.5: Programs for Machine Learning. Morgan Kaufmann, 1992.

[8] Kotsiantis, S. B. Supervised Machine Learning: A Review of Classification Techniques. Informatica, 2007, 31, 249-268

[9] Robert B Cleveland, William S. Cleveland, Jean E. McRae, Irma Terpenning.STL :A Seasonal-Trend Decomposition Procedure Based on Loess.Journal Of Official Statistics,1990.,6(1) : 3-73

[10] G.P. Nason,Introduction to R for Times Series Analysis. http://www.metu.edu.tr/~ceylan/r-ts.pdf, 2005-09-28/2013-02-26.

[11] Bivand, Roger S., Pebesma, Edzer J., Gómez-Rubio, VirgilioXIV.Applied Spatial Data Analysis with R.Springer, 2008

[12] Paul S. P. Cowpertwait,Andrew V. Metcalfe.Introductory Time Series with R.Springer, 2009

[13] Robert H. Shumway,David S. Stoffer.Time Series Analysis and Its Applications -with R examples.Springer, 2011

[14] C. C. Holt.Forecasting seasonals and trends by exponentially weighted moving averages, ONR Research Memorandum. International Journal of Forecasting, 2004 ,Volume 20, Issue 1,: 5-10.

[15] P. R. Winters.Forecasting sales by exponentially weighted moving averages, Management Science Volume 6, 324–342, 1960.

[16] Graham Williams .Data Mining with Rattle and R.Springer, 2011

[17] Graham J Williams.Rattle A Data Mining GUI for R. The R Journal, 2009, Vol. 1/2.

[18] George E. P. Box, Gwilym M. Jenkins, Gregory C. Reinsel. Time series analysis: forecasting and control. Prentice Hall, 1999