Soft Comput (2018) 22:2175–2188
https://doi.org/10.1007/s00500-016-2476-4

METHODOLOGIES AND APPLICATION

Detection of Web site visitors based on fuzzy rough sets

Javad Hamidzadeh[1] · **Mahdieh Zabihimayvan**[2] · **Reza Sadeghi**[2]

Published online: 2 January 2017

Abstract Despite emerging of Web 2.0 applications and increasing requirements to well-behaved Web robots, malicious ones can reveal irreparable risks for Web sites. Regardless of behavior of Web robots, they may occupy bandwidth and reduce performance of Web servers. In spite of many prestigious researches trying to characterize Web visitors and classify them, there is a lack of concentration on feature selection to dynamically choose attributes used to describe Web sessions. On the other hand, depending on an accurate clustering technique, which can deal with huge number of samples in a reasonable amount of time, is practically important. Therefore, in this paper, a new algorithm, fuzzy rough set–Web robot detection (FRS-WRD), is proposed based on fuzzy rough set theory to better characterize and cluster Web visitors of three real Web sites. External evaluations show that in contrast to state-of-the-art algorithms, FRS-WRD achieves better results in terms of *G*-mean 95%, Jaccard 88%, entropy 0.36, and finally, purity 96%. Moreover, according to confusion matrixes, it can better detect malicious Web visitors.

Keywords Web robot detection · Fuzzy rough set · Clustering

Communicated by V. Loia.

✉ Javad Hamidzadeh
j_hamidzadeh@sadjad.ac.ir

Mahdieh Zabihimayvan
m.zabihi@imamreza.ac.ir

Reza Sadeghi
reza.sadeghi@imamreza.ac.ir

[1] Faculty of Computer Engineering and Information Technology, Sadjad University of Technology, Mashhad, Iran

[2] Department of Computer Engineering, Imam Reza International University, Mashhad, Iran

1 Introduction

Nowadays, increasing growth of Internet-based applications and services reveal requirement to some techniques in order to manage and update massive Web repositories. Web robots known as crawlers are practical appearance of such techniques trying to collect and analyze statistics of dynamic content generated by visitors. Crawlers are autonomous agents that start from a list of Web pages and recursively visit all resources accessible from them. Besides crawling Web sites, communication with users, maintaining mirror sites, and testing pages for valid syntax and structure are some helpful tasks of these agents. However, click fraud, collecting business intelligence, DDos attacks, harvesting Email addresses and grabbing the content of Web sites without permission are some purposes of malicious Web crawlers. In addition to these security threats, occupying bandwidth and reducing performance of Web servers lead us to an accurate characterization and detection of Web robots.

A great deal of research has been conducted seeking to characterize Web visitors and detect crawlers based on different data mining concepts. Studies like Sisodia et al. (2015), Suchacka and Sobkow (2015), Tan and Kumar (2002), Lee et al. (2009), Zhao et al. (2013), Bomhardt et al. (2005) using classification techniques and Lu and Yu (2006) utilizing hidden Markov model are some samples of several studies in this field. Such studies, which rely on session labels for detection, should use an accurate session labeling to be adequately reliable.

According to Stevanovic et al. (2013), since a precise session labeling needs a deep insight into dynamic and mutable world of Web visitors, clustering which does not use labels for categorizing sessions can be a better and more accurate alternative. Accordingly, in Stevanovic et al. (2013), two unsupervised neural network learning algorithms, self-

organizing map (SOM) and modified adaptive resonance theory 2 (modified ART2), are used to classify visitors into four groups: Humans, well-behaved clawers, malicious clawers, and unknown visitors. Since Web usage data usually suffer from noises and incomplete vague information, neural networks not only have the potential to extract embedded knowledge in the form of Web session clusters from these huge data, but also provide tolerance against imperfect and noisy data (Ansari et al. 2015). Moreover, the authors use ten features/attributes two of which, standard deviation of requested page's depth and percentage of consecutive sequential HTTP requests, are the new proposed features which are also regarded in this paper.

As another similar research, Zabihi et al. (2014) utilizes DBSCAN (density-based spatial clustering of applications with noises) to distinguish robot traffics for two real Web sites. Moreover, two new features, maximum rate of browser file request and penalty, are proposed based on the behavioral patterns of Web visitors. Besides these features, 12 common attributes are extracted for each session and finally, a significance of the difference test (T test) is used to eliminate irrelevant features and overcome curse of dimensionality.

Human involvement, various types and tasks for Web crawlers, and their dynamic behavior can have influence on selecting the features to describe sessions. In addition, the content of Web sites is the other factor, which has effects on feature selection. Indeed, for a Web site with few numbers of images rather than HTML pages, the HTML/image ratio cannot be a proper attribute. Also, for a Web server with a notable number of requests having response code 304, considering the percentage of such responses can be very useful in contrast to the percentage of 4xx response code. Therefore, relying on an approach to select more proper features and eliminate the irrelevant ones for arbitrary data sets is necessary.

As far as our recent search shows except for Zabihi et al. (2014) which uses a T test to dynamically select more relevant features to describe sessions, other related studies do not focus on this matter. It is worth mentioning that although Zabihi et al. (2014) utilizes a T test for choosing relevant features, this test is just based on comparing a summary statistic over the attributes of robot and human sessions. Hence, using a more long-sighted algorithm for feature selection may have a noticeable effect on the efficiency which is the main novelty used in the proposed algorithm. Furthermore, this methodology can reduce complexity of computations by conserving the variety of samples in contrast to sample reduction methods (Hamidzadeh et al. 2014; Hamidzadeh 2015; Hamidzadeh et al. 2015).

One of the popular algorithms for feature selection is based on fuzzy rough set (FRS) theory (Zabihi et al. 2014) which have a wide range of applications in several fields such as support vector data description improvement (Sadeghi and Hamidzadeh 2016) which is a kernel based classifier (Moghaddam and Hamidzadeh 2016), k-NN (k-nearest neighborhood) improvement (Verbiest et al. 2013; Nowicki et al. 2016), and fuzzy decision tree expansion (Wang et al. 2008; Zhai 2011). In fact, FRS has a potential to cope with vagueness in data by calculating the membership degree of each sample. Hence, it can be useful in feature selection for Web robot detection where the attribute values suffer from vagueness, incompleteness, and noises.

As a result, in this paper, a novel algorithm called FRS-WRD (fuzzy rough set–Web robot detection) is proposed to characterize visitors by possibly the most relevant features and finally cluster sessions. Shortly, the main purposes of FRS-WRD are:

1. To eliminate irrelevant features by fuzzy rough set (FRS) theory and consequently overcome the curse of dimensionality as the main novelty used in FRS-WRD.
2. To use a sufficiently potent clustering algorithm to accurately categorize massive data sets of Web sessions in an acceptable time.
3. To rely on an algorithm which can well distinguish between robots and humans and furthermore correctly detect malicious Web visitors.

Briefly, some features proposed in other related studies and introduced to be useful in Web robot detection are extracted for each session. After that, FRS feature selection Pawlak (1982) is employed to possibly select the more proper features from the extracted attributes. Eventually, with inspiration from Stevanovic et al. (2013), all sessions are clustered into human and Web robot classes by the SOM neural network algorithm. At last, interpretation of final clusters specifies sessions of malicious Web visitors. Conducted experiments follow two specific goals: first, speed and ability of FRS-WRD in clustering Web visitors are compared with those of state-of-the-art algorithms. Second, the proposed algorithm and the state-of-the-art ones are compared with each other to reveal the superior algorithm for detecting malicious visitors. All experimental results are reported in terms of some external evaluation metrics and confusion matrices.

The content of this paper is organized as follows: In Sect. 2, other related works on Web robot detection are presented. Brief background information on fuzzy rough set theory and the SOM neural network is demonstrated in Sect. 3. The proposed algorithm is considered in Sect. 4. Section 5 describes the experiments used in this paper, while Sect. 6 summarizes the conclusions and future work.

2 Related works

One of the main factors which has noticeable influence on the performance of methods used for Web robot detection is the relevance of each attribute for describing Web sessions.

To date, there are many prestigious papers proposing some innovative features to characterize robots and human users. In a general view, we can categorize these features into two classes of the attributes, which are good indicators for Web robots, and the features, which have distinguishing values for human visitors.

Regarding to the first class, accessing to *robots.txt* file can simply reveal a session belonging to a Web robot. Because such a file is one of the resources with no link on Web pages for human accessing and typically most humans are unaware of their existence (Tan and Kumar 2002; Zabihi et al. 2014a). Also, the percentage of requests sent to a Web server during night hours and the percentage of requests with 4xx response code are the other features which can indicate robot sessions (Stevanovic et al. 2012, 2013).

Session time, the number of seconds between the first and last requests in a session, is the other feature with higher value for robot sessions than human ones (Grzinic et al. 2015; Stassopoulou and Dikaiakos 2009). In fact, it can be explained by the fact that in contrast to humans with a specific goal in their searching, Web robots try to protract their surfing on a Web site to access to all required resources. Consequently, click number, the number of HTTP requests sent in a session, is the other numerical metric for detecting the presence of Web crawlers Stevanovic et al. (2012), Stevanovic et al. (2013) which have higher values for this attribute. In addition, standard deviation of requested page's depth introduced in Stevanovic et al. (2012) is a numerical attribute calculated as the standard deviation of page depths across all requests in a session. Since robots possess simple navigational patterns and do not have to follow the link structures of Web pages, this feature has large values for Web robot sessions. Finally, the percentage of unassigned referrer field is one of the other Web robot characteristics used in Lee et al. (2009), Lourenco and Belo (2006), Tan and Kumar (2002).

Following this, there are some useful indicators for human users such as switching factor on number of bytes sent from clients to Web servers (Kwon et al. 2012). Also, the authors suggest that switching factor of requested files types have unanimously and notably lower values for Web crawlers. In addition, regarding to the navigational differences between humans and Web robots, Zabihi et al. introduce two new features called penalty and the maximum rate of browser file request as useful indicators for human visitors (Zabihi et al. 2014, a). Similarly, according to (Bomhardt et al. 2005; Stassopoulou and Dikaiakos 2009), the percentage of image requests in a session is usually high for human users.

Despite all the above attributes introduced by the precious studies, the mutable nature of Web visitors, various tasks for Web robots, and especially the different contents of Web sites reveal the need for an approach to dynamically choose the features for any arbitrary Web site. As previously mentioned, for a Web site which does not exhibit many image files, the percentage of image requests cannot be an apparent metric for human users. Also, in spite of the effectiveness of standard deviation of requested page's depth (Stevanovic et al. 2012), it may not be useful for Web sites with few number of pages. As a result, relying on an approach to cope with this vagueness intrinsically present in Web data and eliminate the irrelevant features not only has remarkable influence on detection performance, but also removes the curse of dimensionality.

According to our recent search, it seems that Zabihi et al. (2014) is the only related research which utilizes T test to select more relevant features among fourteen valid attributes. Although their proposed method has effective performance in distinguishing robots from human users, T test is only based on comparing a summary statistic over the attributes and, thus, uses a limited view on the data.

In this paper, our proposed algorithm, FRS-WRD, exerts the SOM neural network to cluster Web sessions similar to Stevanovic et al. (2013). On the other hand, it selects the probably more relevant features based on fuzzy rough set (FRS) to not only deal effectively with the data vagueness but also eliminate the curse of dimensionality. Moreover, on the contrary to the T test used by Zabihi et al. (2014), FRS considers the similarity of all attributes simultaneously to gain a more expanding view on data.

3 Background

In this section, background information on fuzzy rough set theory and the SOM clustering algorithm is presented, respectively.

3.1 Fuzzy rough set

Fuzzy rough set (Dubois and Prade 1990), FRS, was made by combining both fuzzy (Zadeh 1974) and rough set (Pawlak 1982) theories to categorize not only vague but also incomplete data. The main goal of rough sets is to categorize data according to an initial similarity and then classify them into three specific groups.

First group contains all samples which certainly belong to the objective data, while second group comprises all samples which may be objective, and the last one includes data which are not confidently objective ones. In this categorization, data should be characterized by discrete features like "Trap File Requests," "Penalty," "Width," and "Depth." Although, there are some continuous attributes predominantly used for describing Web sessions. "PPI," "session time," and "RES" are some examples of such features (for more information about mentioned attributes, refer to Appendix 1). Therefore, FRS which can apply this categorization for both discrete and continuous attributes is essential in practice.

FRS uses two main concepts, fuzzy rough lower and upper approximation memberships, to define the certainty and probability degree for each sample. Moreover, the basic relations are defined by two fuzzy operators, triangular (T-norm) and implicator I, which are generalized in Radzikowska and Kerre (2002) and are in the form of Lukasiewicz in this paper.

Assume a collection of samples described by a set of conditional attributes, c and categorized by a set of decision features, d, in data set (DS). The lower and the upper memberships of sample s_i toward the others when DS is segregated by function F are formulated in Eqs. (1) and (2), respectively:

$$\mu_{\underline{F_{R_c,d}}}(s_i) = \inf_{s_j \in \mathrm{DS}, s_i \neq s_j} \mathrm{I}\left(R_c(s_i, s_j), R_d(s_i, s_j)\right) \tag{1}$$

$$\mu_{\overline{F_{R_c,d}}}(s_i) = \sup_{s_j \in \mathrm{DS}, s_i \neq s_j} \tau\left(R_c(s_i, s_j), R_d(s_i, s_j)\right) \tag{2}$$

where $R_c(S_i, S_j)$ shows the similarity of two samples, S_i and S_j, in the form of Eq. (3) and $R_d(S_i, S_j)$ described in Eq. (4) indicates their membership rate to be in the same class based on an objective function:

$$R_c(s_i, s_j) = \underset{a \in c}{\tau}\left(\max(0, 1 - |s_i(a) - s_j(a)|^2)\right) \tag{3}$$

$$R_d(s_i, s_j) = \underset{a \in d}{\tau}\left(\max(0, 1 - |s_i(a) - s_j(a)|^2)\right) \tag{4}$$

Obviously for $R_c(S_i, S_j)$ and $R_d(S_i, S_j)$, the results become maximum, namely 1, which cause lower and upper approximation memberships to be equal to 1. Therefore, such situations are practically avoided in the calculation of membership functions through Eqs. (1) and (2).

Finally, to reduce the sensitivity of membership degrees to noises (Verbiest et al. 2013b), FRS is improved by the ordered weight average (OWA) concept while $|DS|$ shows total number of samples:

$$\mu_{\underline{F_{R_c,d}}}(s_i) = \underset{s_j \in \mathrm{DS}, s_i \neq s_j}{\mathrm{OWA}_{\min}} \mathrm{I}(R_c(s_i, s_j), R_d(s_i, s_j)) \tag{5}$$

$$\mu_{\overline{F_{R_c,d}}}(s_i) = \underset{s_j \in \mathrm{DS}, s_i \neq s_j}{\mathrm{OWA}_{\max}} \tau(R_c(s_i, s_j), R_d(s_i, s_j)) \tag{6}$$

In the above equations, inputs of the OWA operators are Is and τs sorted descendingly. $\mathrm{OWA}_{\min}$ multiplies $\frac{2}{((|\mathrm{DS}|-1).|\mathrm{DS}|)}, \frac{3}{((|\mathrm{DS}|-1).|\mathrm{DS}|)}, \ldots, \frac{2.|\mathrm{DS}|}{((|\mathrm{DS}|-1).|\mathrm{DS}|)}$ by the inputs while $\mathrm{OWA}_{\max}$ uses $\frac{2.|\mathrm{DS}|}{((|\mathrm{DS}|-1).|\mathrm{DS}|)}, \ldots, \frac{3}{((|\mathrm{DS}|-1).|\mathrm{DS}|)}, \frac{2}{((|\mathrm{DS}|-1).|\mathrm{DS}|)}$ as their coefficients. Therefore, OWA involves all samples to reduce the effects of noisy and outlier samples in computing membership functions.

It is notable that FRS has close relation with belief function theory (Shafer 1976) which has noticeable performance in many applications (Antoine et al. 2014; Liu et al. 2016, 2015). In fact, it endeavors to demonstrate the universe of discourse according to occurrence probability of a subset in the universe of discourse.

Similar to FRS, the belief function theory is an extension of belief function which introduced two basic concepts. They are belief and plausibility functions. The belief one for a certain set let assume $A \subset U$aggregates the occurrence probability of all subset of A while the other function sums up the occurrence probability of all subsets of the universe of discourse which have at least one common item with A. This description of the universe of discourse is exactly equivalent to rough set constructed based on lower and upper approximation sets (Yao and Lingras 1998). This resemblance can be generalized to fuzzy one as it is discussed in Dubois and Prade (1990), Wu et al. (2002), Chen et al. (2008), Inuiguchi et al. (2015), Sadeghi and Hamidzadeh (2016).

3.2 The SOM clustering algorithm

SOM is a type of artificial neural networks trained by competitive unsupervised learning. Its aim is to make different parts of the network similarly respond to same input patterns. SOM is used to map multidimensional input data to a lower-dimensional subspace. The geometric relationships between points indicate their similarity usually defined by Euclidean distance (Kohonen 2013).

According to Fig. 1, SOM neural network contains an input layer, vectors in the form of $(X_1, X_2, \ldots, X_k)$, and an output layer which is a two-dimensional structure of neurons or clusters. In this paper, the input vectors are the sessions corresponding to the features indicated by $X_i (i = 1, \ldots, k)$.

Each neuron/cluster associates with k-weighted synapsis, while k is the number of variables in each input vector. These weights specify how much the input is important for neuron activation function. Indeed, W_{ij} shows the synapsis between ith variable of the input and the jth neuron of the map. If l shows the number of all clusters and O_i exhibits output of ith neuron, the input variables are multiplied by the associated weights and then, sum of all produces the input of activation function (f). This function determines final output of neuron under influence of the α parameter which defines learning rate and is reduced by time.

Since each input vector, Web session, is randomly adjusted to the network, learning algorithm of SOM (in this research, Batch Unsupervised Weight/Bias Training) aims to find and move the closest neuron to each input sample. Such a neuron is called winner node.

After finding the winner neuron, its weights are updated according to the learning algorithm. If s shows the sth step of the algorithm, W_m^s defines the weight vector of the cluster m in step s, and Xpresents the input sample. Accordingly, index q for this winner neuron is defined as Eq. (7):

$$q = \arg_m \min \left\| W_m^s - X \right\| \tag{7}$$

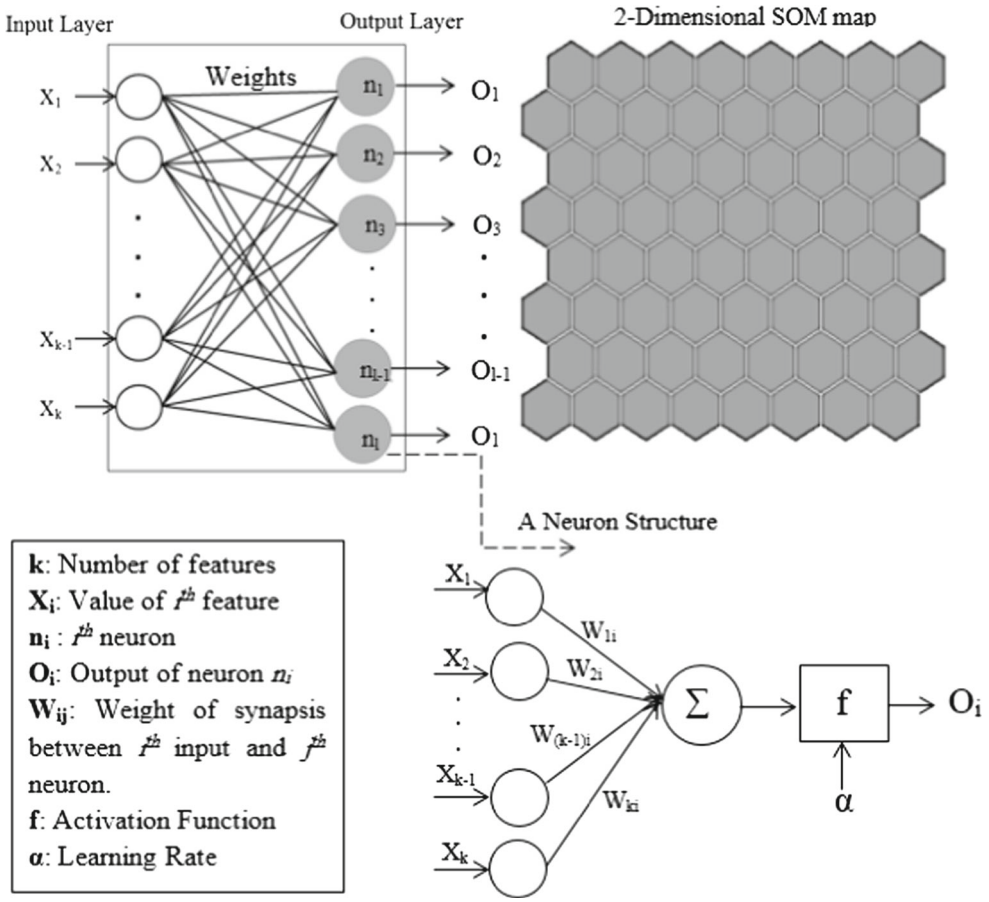

Fig. 1 The SOM map and the structure of a neuron

Moreover, update rule is calculated by Eq. (8) where α^s shows the learning rate of sth step:

$$W_q^{s+1} = W_q^s\left(1-\alpha^s\right) + X_i\alpha^s + \alpha^s\left(X_i - W_q^s\right) \tag{8}$$

4 The proposed algorithm

The FRS-WRD algorithm proposed in this paper uses the SOM neural network to categorize Web visitors into human and Web robot classes. Furthermore, it utilizes the FRS feature selection to eliminate irrelevant attributes extracted for Web sessions and overcome the curse of dimensionality.

According to Fig. 2, input of the proposed algorithm is an original server access log file in combined log format. In such a file, each received request is shown in a separate line and a user's session is defined as a set of HTTP requests received by Web server. To identify a session, two consecutive requests having similar IP and user agent strings will belong to a same session if time difference between them is less than 30 minutes (a standard threshold in majority of related literatures).

After session identification, some features introduced in other related works (Bomhardt et al. 2005; Kwon et al. 2012; Lee et al. 2009; Stevanovic et al. 2013; Tan and Kumar 2002; Zabihi et al. 2014) and shown to be useful in distinguishing between robots and humans are extracted for each session. These attributes are the primary ones which should be filtered by the FRS feature selection in the next step to specify the more proper features. Appendix 1 briefly shows all the primary attributes used in this paper. A C#-based program has been designed to analyze the original file, identify sessions, and extract the primary features for every session.

As mentioned previously, the values of the most attributes used to describe sessions suffer from ambiguity and especially noise data. Hence, in the next step, we utilize the FRS feature selection to select effective group of attributes and eliminate the inappropriate features.

In this paper, the FRS feature selection is utilized by implementing the forward approximation algorithm (FAA) proposed in Qian et al. (2015) because of its ability to accelerate the selection of samples by merging sample and dimensionality reduction, simultaneously. FAA is an iterative algorithm that tries to gain a decision boundary (DB) on

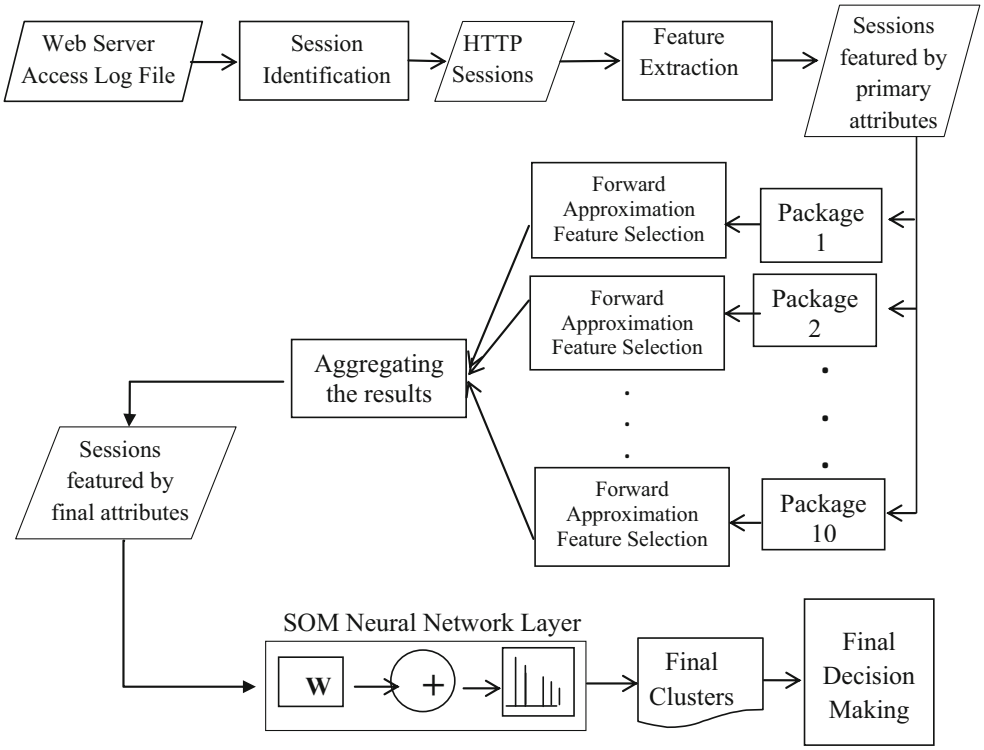

Fig. 2 Flowchart of the proposed algorithm

the universe of discourse to segregate humans from robots by choosing a minimal group of the features describing each session. In FAA, the ability of a feature group, $\emptyset \in$ features, to show the correct labels of all samples according to DB is explained by $\gamma_{\varphi \subseteq \text{Features}}$ which is called dependency function. As described in Eq. (9), $\gamma_{\varphi \subseteq \text{Features}}$ shows the percentages of samples certainly located in DB where |DS| indicates the total number of samples:

$$\gamma_{\varphi \subseteq \text{Attributes}} = \frac{\left| \sum_{s_i \in \text{Data Set}} \mu_{\underline{DB_{IR_{\varphi,\text{Label}}}}}(s_i) \right|}{|\text{DS}|} \tag{9}$$

In each iteration, FAA assigns one attribute to the final set. This set containing the result of the FAA algorithm will involve all finally selected attributes. At first, FAA checks all attributes and calculates the dependency function for all of them. Accordingly, the attribute with the maximum value of $\gamma_{\varphi \subseteq \text{Features}}$ is added to the final set. In the next iteration, the other attributes will be separately considered in a pair containing the feature having been added to the final set. $\gamma_{\varphi \subseteq \text{Features}}$ should be calculated for each pair of attributes, and the one having more dependency function than that gained in the previous iteration forms the new value of the final set. This process continues and gradually increases the size of the final set. The algorithm will be stopped when adding more new features to the final set cannot improve the value of the dependency function.

Although FAA decreases the computational time for big data sets, it suffers from high computational complexity made by high-dimensional samples. Therefore, in this paper, ten packages each of which contains ten thousand sessions randomly selected are used in practice to execute FAA separately for each package. Finally, ten sets are gained to demonstrate the features selected for each of them. Afterward, based on a predefined threshold (shown by *FRS_thre*), the final attributes are chosen according to the percentage of time (*P_T*) they emerged in the ten sets. For example, if an attribute exists in six of the ten sets, its *P_T* is equal to 60%. Accordingly, if *FRS_thre* is equal to 60, all the features having *P_T* $\geq$60% are chosen as the final ones.

After eliminating the irrelevant features, sessions with finally chosen attributes are ready to be clustered by the SOM neural network.

Eventually, the resulted clusters are considered as human and robot classes and evaluated by some external metrics. In addition, sessions of malicious Web visitors are traced and the ability of FRS-WRD in detecting them is examined. Figure 3 demonstrates pseudocode of the FRS feature selection and proposed algorithm.

5 Experimental results

To validate the proposed FRS-WRD algorithm, some experiments have been conducted over three real data sets. For all

Fig. 3 Pseudocode of the proposed algorithm

```
FRS_FeatureSelection (FeaturedSessions) // FRS feature selection procedure

    PackageList: array of packages // each package is a set of 10000 sessions.
    FAFS: array containing results of Forward Approximation feature selection executed over
    each element of PackageList.
    FinalAttrs: array of attributes selected as the final ones (the final set).
    thre: integer // the predefined threshold value (FRS_thre).

    for i=1 to 10 do
        PackageList [i] = 10000 sessions randomly selected from FeaturedSessions;
        FAFS [i] = the results of Forward Approximation feature selection algorithm executed
        on PackageList [i];
    end;
    for i=1 to 10 do
        prec: integer // the p_t for each attribute.

        for each attributes of FAFS[i]
              prec = compute the p_t for the attribute;
             if prec >= thre then
                 Add this attribute to FinalAttrs;
        end;
    end;
    return FinalAttrs;
end;
```

```
main procedure (OriginalFile)

    OriginalFile: the original access log file which contains all request records.
    HTTPSessions: set of all HTTP sessions found in the original access log file.
    FeaturedSessions: set of sessions featured by primary features (shown in Appendix 1).
    FinalllyFeaturedSessions: set of sessions featured by final attributes.
    FinalClusters: set of final clusters made by SOM.

    HTTPSessions = SessionIdentification (OriginalFile);
    FeaturedSessions = FeatureExtraction (HTTPSessions);
    FinallyFeaturedSessions = FRS_FeatureSelection (FeaturedSessions);
    FinalClusters = SOM (FinallyFeaturedSessions) // execute SOM to cluster all sessions
    featured by final attributes.
end;
```

Table 1 Description of data sets employed in this paper

Web site name	Data set name	No. of requests	No. of sessions	No. of all Web robots	No. of well-behaved Web robots	No. of malicious Web robots	No. of humans
Imam Reza International University (Imam Reza International University, 2015)	DS-IR	311633	17969	1170	1118	52	16799
Torghe Energy (Torghe Energy, 2015)	DS-T	500781	21533	6093	5984	109	15460
ArticleBaz (ArticleBaz, 2014)	DS-AB	400556	18884	5623	5552	71	13261

experiments, tenfold cross-validation is used. Accordingly, each data set is divided into ten mutually exclusive blocks. Training set is built with nine of the ten blocks, while the remaining one is used for test. Finally, average of the ten test sets is reported.

The selected data sets and their related parameters are listed in Table 1. This table shows the description in the form of Web site name, data set name, and number of all requests, all sessions, all robots, well-behaved Web robots, malicious ones, and finally human users.

In order to disclose the performance of FRS-WRD, we compare it with state-of-the-art algorithms. As mentioned earlier, Stevanovic et al. (2013), we call in this paper NN-WRD (neural network for Web robot detection), is one of the leading-edge algorithms employing the SOM and modified ART2 clustering methods to categorize Web sessions into humans, well-behaved robots, malicious crawlers, and unknown visitors. On the other hand, Zabihi et al. (2014), DBC-WRD, is the other related research using DBSCAN to cluster sessions into humans and Web robots. What is more, a T test is applied to select the more proper attributes among 14 features used to characterize Web sessions.

For externally evaluation, all initial sessions are labeled as humans or Web robots. According to Stevanovic et al. (2013), all sessions that have the user agent of a known Web browser and do not access to *robots.txt* file are labeled as humans. A user agent string matches a known Web robot reveals a session corresponding to Web crawlers.

All the tests have been performed on an Intel processor at 2.53 GHz with 4 GB of RAM. To have fair comparisons, related parameters are adjusted based on the values set in the algorithms. Accordingly, to implement SOM, the MATLAB's Neural Network Toolbox is adjusted with a network made of 100 neurons in a 10-by-10 hexagonal arrangement with epoch = 200. To execute the modified ART2 clustering, a MATLAB code based on an algorithm introduced in Vlajic and Card (2001) has been implemented and the $p_{\max}$, Δ_p, and $n_{\max}$ parameters have been set to 1.2, 0.4, and 3, respectively. To perform the DBSCAN algorithm, WEKA 3.6.9 has been used while *epsilon* and *minPoints* have been set to the default values. Finally, The *FRS_thre* used in the FRS feature selection is set to 20 for all three data sets.

The results are reported in the form of G-mean, Jaccard, Entropy, and Purity rates while TP and TN are the numbers of robots and humans correctly classified. Likewise, FN shows the number of robots which are incorrectly specified as humans and FP is the number of humans clustered as Web crawlers. Since the above data sets are notably imbalanced and thus, Rand Index or generally, the accuracy Amigo et al. (2013) is biased toward the majority classes, it cannot be a good criterion characterizing final performance. Therefore, G-mean as the performance measure monitoring both accuracy of the majority and minority classes is used as Eq. (10). According to the definitions of TPR (true positive rate) and TNR (true negative rate), which indicate the rate of true detection for robot and human classes respectively, the G-mean metric shows the accuracy of the detection algorithm.

$$G\text{-mean} = \sqrt{\text{TPR} \times \text{TNR}}, \quad \text{TPR} = \frac{\text{TP}}{\text{TP} + \text{FN}}$$
$$\text{TNR} = \frac{\text{TN}}{\text{TN} + \text{FP}} \tag{10}$$

Jaccard coefficient, as the other metric used for problems with asymmetric binary classes, is derived from the Rand index equation TN of which is deleted to accentuate the importance of TP (Amigo et al. 2013). Regarding to the TP definition, P refers to the robot class; therefore, Jaccard is an important metrics which can show how the detection algorithm can cluster crawlers well. Equation (11) indicates this metric as follows:

$$\text{Jaccard} = \frac{\text{TP}}{\text{TP} + \text{FP} + \text{FN}} \tag{11}$$

The entropy metric reveals how all samples are distributed within each cluster. In fact, a good clustering technique has a small value for entropy (Amigo et al. 2013). This metric is computed as Eq. (12) where n shows total number of samples

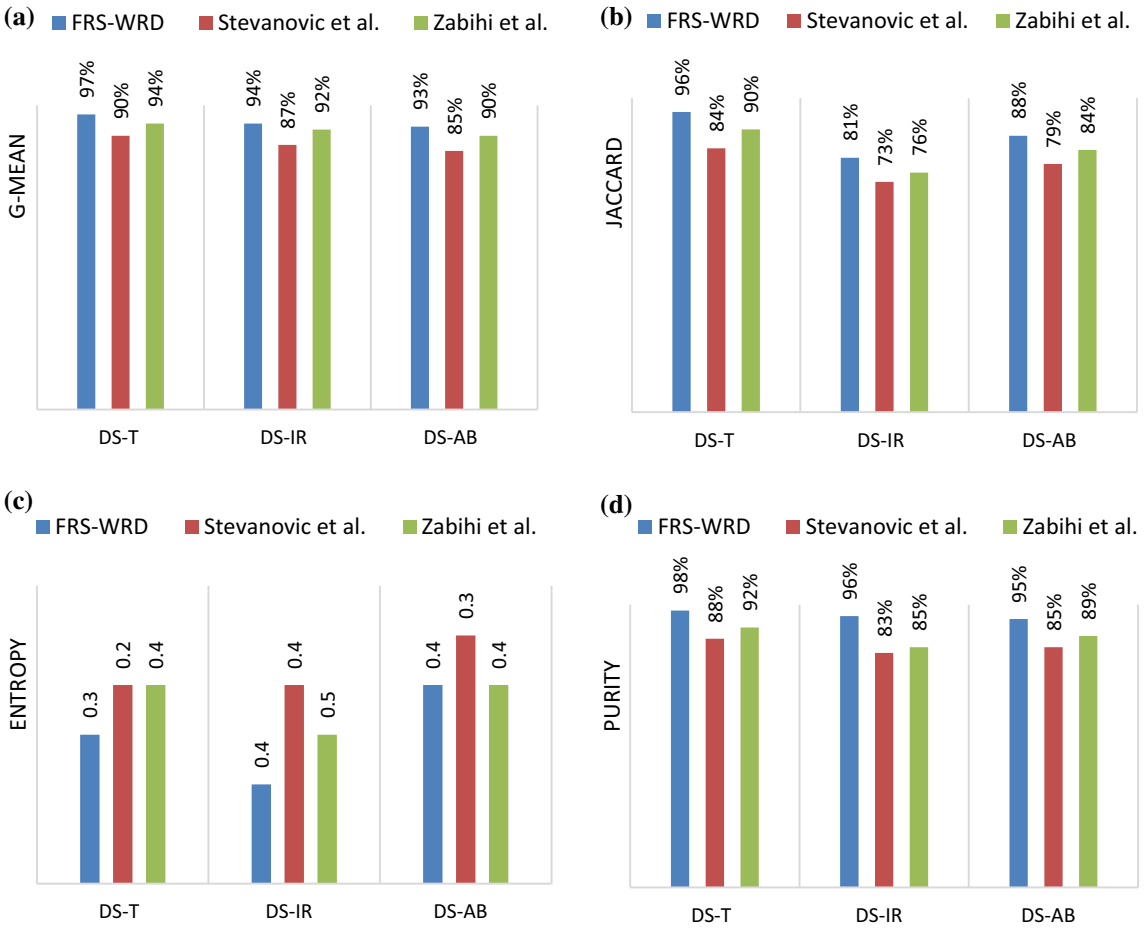

Fig. 4 The external evaluation metrics reported for FRS-WRD and state-of-the-art algorithms **a** *G*-mean, **b** Jaccard, **c** Entropy, **d** Purity

while n_c indicates number of existing categories. Moreover, n_{ij} is number of elements from category i in cluster j having n_j elements.

$$\text{Entropy} = \sum_{j=1}^{n_c} \frac{n_j}{n} e_j, \quad e_j = -\sum_{i=1}^{n_L} \frac{n_{ij}}{n_j} \log_2 \frac{n_{ij}}{n_j} \tag{12}$$

Ultimately, purity is one of the most popular measures for cluster evaluation which concentrates on frequency of the most common category in each cluster. Indeed, it is calculated by considering weighted average of maximal precision values (Amigo et al. 2013). Equation (13) shows how to compute this metric:

$$\text{Purity} = \sum_{j=1}^{n_c} \frac{n_j}{n} p_j, \quad p_j = \max_i \frac{n_{ij}}{n_j} \tag{13}$$

Figure 4 demonstrates the values of the above evaluation metrics for the three comparative algorithms on the data sets.

According to Fig. 4a, FRS-WRD has better *G*-mean results than the state-of-the-art algorithms. Hence, the ability of the proposed algorithm in detecting both robots and humans classes is higher than the compared ones. In addition, DBC-WRD gains better *G*-mean results in contrast to NN-WRD. It can be attributed to the feature selection approach used by the authors. However, the better results for FRS-WRD than DBC-WRD may reveal the superiority of the FRS feature selection to the T test used by DBC-WRD due to its more expansive view on all attributes.

According to Fig. 4b, FRS-WRD gains higher results for the Jaccard metric rather than the state-of-the-art ones. It indicates the more ability of the proposed algorithm in clustering Web robots and separating them from human users. Higher Jaccard values gained by FRS-WRD than those resulted by DBC-WRD probably indicates the higher capability of FRS feature selection in comparison with the T test. Besides, the higher Jaccard values of DBC-WRD than those of NN-WRD seemingly offers that how focusing on dynamic feature selection to choose more proper attributes can enhance the detection performance.

Figure 4c, d shows the entropy and purity values for the comparative algorithms, respectively. As indicated, the lower

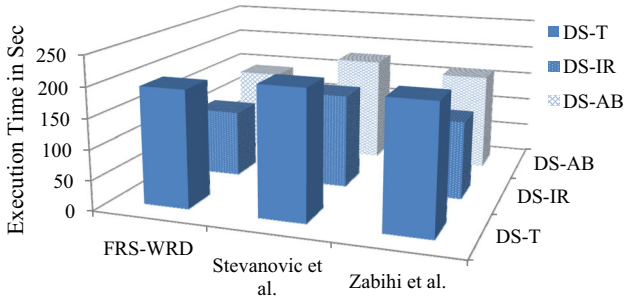

Fig. 5 Total time taken (in sec) to execute FRS-WRD and state-of-the-art algorithms

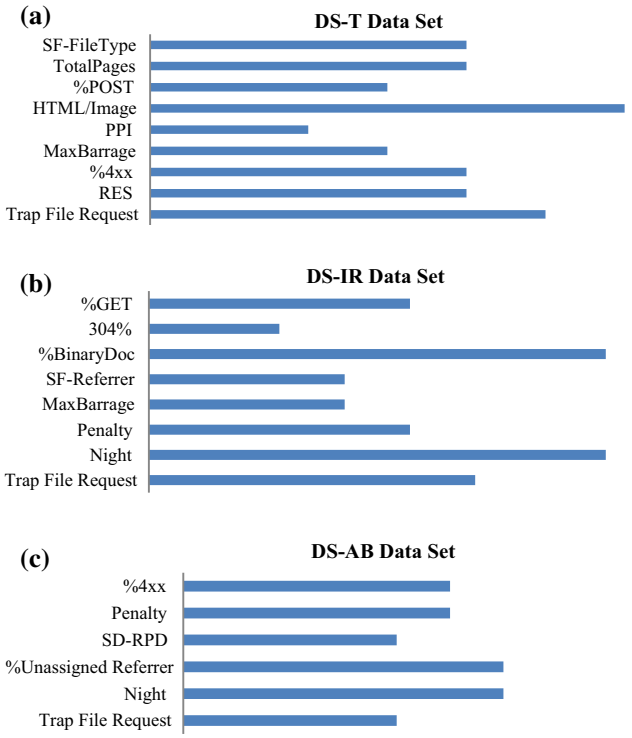

Fig. 6 The attribute selected as final after conducting FRS feature selection over each data set

entropy and higher purity values for FRS-WRD reveal that this algorithm can better separate both robot and human classes from each other and form clusters with lower disordering. Similarly, it seems that not only such an ability is made by the FRS feature selection used in FRS-WRD to eliminate the irrelevant features, but also it is superior to the *T* test in characterizing the Web sessions.

Since the main goal of Web robot detection is to protect the Web servers against unfavorable Web agents, examining such sessions and limiting their accessing to Web sites should be done fast. Therefore, in addition to the evaluation metrics reported for the three algorithms, their total execution times are compared to indicate how quickly they can distinguish Web robots from human users. Moreover, as Web robot detection algorithms use a preprocessing stage to make original data sets useable, the execution times reported in Fig. 5 are just for the clustering phase not the preprocessing step.

According to Fig. 5, FRS-WRD has lower execution time rather than the others. Although both the FRS-WRD and NN-WRD employ SOM for clustering, the FRS feature selection used in the proposed algorithm diminishes the number of final attributes used for session characterization and, thus, the total execution time of the clustering.

Figure 6 demonstrates results of the FRS feature selection as the final attributes chosen for each data set.

Accordingly, the finally chosen attributes gained for a data set can be significantly different with those of other data sets. For examples, the attributes chosen for data set D–T show that for such a Web site which has few numbers of visitors interested in images rather than HTML files, the *HTML-to-Image ratio* can be a good separating feature. Similarly, while *%4xx* is a promising feature for DS-AB, *%304* is chosen as a proper attribute for DS-IR. Likewise, although *Penalty* and *Max Barrage* are introduced as good indicators for humans, they are not selected as final features for DS-T and DS-AB, respectively. *%Night* is the other attribute introduced as an apparent metric for robots sessions, while it is selected as the final feature for just both data sets (DS-IR and DS-AB). One of the interesting attributes selected for the whole data sets is *Trap file request*. But it does not mean that it can be a promising feature for any arbitrary data set. On the other hand, despite the popularity of *%Unassigned Referrer* and *SD-PRD* in several valuable studies, in this paper, they are chosen just for one data set (DS-AB).

Briefly, it shows that how the differences between contents of Web sites and navigational patterns of Web visitors can effect on the attributes chosen for describing Web sessions. Hence, merely relying on a number of features which have been even confirmed in many literatures is not sufficient enough to expect a high performance for every data set.

At last, the ability of the detection algorithms in distinguishing malicious visitors are compared with each other. To shorten the comparisons, the following reports are related to DS-AB data set. Figure 7 shows the map result of neuron hits in FRS-WRD conducted over this data set. According to test data, the major and minor class found in each neuron is specified as the size and type of classes indicated in the form of 'R' for robots and 'H' for humans. For instance, the first neuron in the top left corner of the map shows a neuron with 659 robots (the major class) and 68 humans (the minor class).

According to Fig. 7, there are some neurons/clusters the major classes of which are human users. Among the crawlers clustered in such neurons (the minor class), there are some Web robots having the user agent strings or IP addresses of known malicious Web robots [List of User Agents (Spiders, Robots, Browser) 2015; Staeding 2015]. Hence, it seems

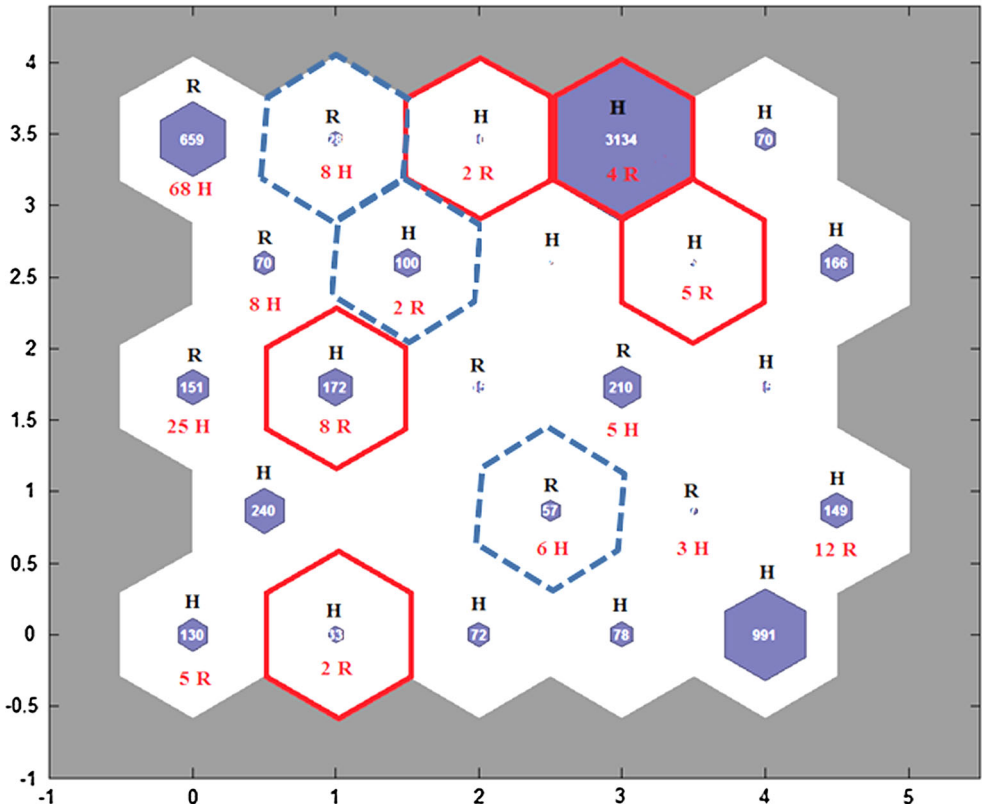

Fig. 7 Map result of neuron hits for FRS-WRD based on the SOM clustering. *Bolded neurons* are the clusters containing malicious Web robots imitating human behaviors, while *dashed ones* are neurons including malicious human users

Table 2 Confusion matrix of malicious visitors over DS-T data set

Detected actual	Web robots		Human users	
	FRS-WRD	NN-WRD	FRS-WRD	NN-WRD
Malicious robots	37	26	15	26
Human users	23	37	15437	15423

Table 3 Confusion matrix of malicious visitors over DS-IR data set

Detected actual	Web robots		Human users	
	FRS-WRD	NN-WRD	FRS-WRD	NN-WRD
Malicious robots	37	26	15	26
Human users	12	29	16787	16770

that such malicious crawlers have attempted to imitate the human behavior and, thus, have been clustered in a neuron with dominant population of humans. Bolded neurons are the human clusters which contain these malicious Web robots. Undoubtedly, other robots clustered in these neurons show weakness of the detection algorithm in correctly identifying them. Similarly, there are some neurons with the major class of robots. More scrutinizing on the minor classes (humans) of such neurons reveal that theses minorities contain some human users who have accessed the *robots.txt* file which is predominantly visited by Web crawlers. Thus, such humans have seemingly the potential to be malicious. Dashed neurons indicate the neuron with major robot classes which include these malicious humans. Evidently, other humans located in these neurons show the false detection of the algorithm.

To gain a better view on the performance of FRS-WRD in detecting malicious Web visitors, the above explanations are summarized in Tables 2, 3 and 4 as the confusion matrices of such sessions. In addition, since DBC-WRD does not focus on malicious visitors detection, the performance of FRS-WRD in detecting such users are compared with the efficiency of NN-WRD over all three data sets.

Table 4 Confusion matrix of malicious visitors over DS-AB data set

Detected actual	Web robots		Human users	
	FRS-WRD	NN-WRD	FRS-WRD	NN-WRD
Malicious robots	48	32	21	39
Human users	16	29	13245	13232

Table 5 Primary features extracted for each session

Id	Attribute name	Description	Index
1	Trap file request	Shows that if a session includes requests for trap files which are the resources human users are unaware of their existence since there are no links to them in a Web site page (Zabihi et al. 2014)	R
2	Session time	The approximated time difference between the first and the last requests in a session (Tan and Kumar 2002)	H
3	% Night	The percentage of requests demanded between 12 am and 7 am (Tan and Kumar 2002)	R
4	% Unassigned referrer	The percentage of requests with unassigned referrer (Tan and Kumar 2002)	R
5	SD_RPD	Or standard deviation of requested page's depth which states the standard deviation of page's depth across all requests sent in an individual session (Stevanovic et al. 2013)	R
6	% CSR	Or percentage of consecutive sequential HTTP requests of a session for all pages belonging to the similar directory (Stevanovic et al. 2013)	H
7	SCB-SCB	Number of bytes sent from the server to a client and vice versa (Lee et al. 2009)	H
8	RES	Time taken to serve a request (Lee et al. 2009)	H
9	% Head	Percentage of HTTP requests of type HEAD sent in a single session (Tan and Kumar 2002)	R
10	% 4XX	Percentage of erroneous HTTP requests sent in a single session (Stevanovic et al. 2013)	R
11	Penalty	A penalty value for every back-and-forward navigation or loop (Zabihi et al. 2014)	H
12	Max Barrage	Or Maximum rate of browser file requests sent in a single session (Zabihi et al. 2014)	H
13	SF-FileType	Or switching factor of file types for each individual session (Kwon et al. 2012)	R
14	SF-csbytes	Or switching factor on number of bytes from clients to the server (Kwon et al. 2012)	
15	SF-referrer	Or switching factor on unassigned referrer field (Kwon et al. 2012)	R
16	Width	The number of leaf nodes generated in the graph showing all the HTTP requests sent in a session (Tan and Kumar 2002)	R
17	depth	The maximum depth of the tree within the graph showing all the HTTP requests sent in a session (Tan and Kumar 2002)	H
18	TotalPages	Total number pages requested in a single session (Tan and Kumar 2002)	H
19	PPI	Or Page popularity index showing the average value of page popularity index for all pages retrieved in a single session (Lee et al. 2009)	H
20	HTML-to-Image ratio	The number of HTML page requests over the number of images requested in an individual session (Stevanovic et al. 2013)	R
21	% Zip	Percentage of zip/gz files requested in a session (Tan and Kumar 2002)	R
22	% Binary Doc	Percentage of PDF/PS files requested in a session (Tan and Kumar 2002)	R
23	% Binary Exec	Percentage of cgi/exe files requested in a session (Tan and Kumar 2002)	H
24	MultiIP	Shows that if a session contains multiple IP addresses (Tan and Kumar 2002)	R
25	MultiAgent	Shows that if a session contains multiple user agent strings (Tan and Kumar 2002)	R
26	% 304	Percentage of HTTP requests sent in a single session with status code 304 (Bomhardt et al. 2005)	H
27	% Multimedia	Percentage of multimedia files requested in a session (Tan and Kumar 2002)	R
28	% Other	Percentage of other type of resources requested in a session (Tan and Kumar 2002)	R
29	% POST	Percentage of HTTP requests of type POST sent in a single session (Tan and Kumar 2002)	H
30	% GET	Percentage of HTTP requests of type GET sent in a single session (Tan and Kumar 2002)	H

As shown in the above tables, the number of actual malicious Web crawlers which identified as Web robots show the number of malicious crawlers correctly detected. In contrast, the number of actual malicious robots which identified as humans indicate the number of malicious crawlers imitating the human's behaviors. Likewise, the number of actual

humans detected as human users indicate the true detection of the algorithm, while the number of actual humans identified as Web robots show the human users which may have the malicious potential or indicate the false detection of the algorithm.

According to the results, the numbers of malicious crawlers which have been correctly identified are higher for FRS-WRD than those reported for NN-WRD. On the other hand, although the numbers of actual malicious crawlers detected as humans are larger for NN-WRD, it does not mean the superior ability of this algorithm in detecting malicious robots. Because according to the evaluation results reported in Fig. 4b, the performance of FRS-WRD in distinguishing robots from human users is better than that of NN-WRD. Hence, it seems that larger numbers of actual malicious robots detected as humans reveal the larger false detection rate of NN-WRD in separating malicious crawlers from human visitors.

Following this, the numbers of actual humans correctly detected as human users are higher in the results of FRS-WRD than those of NN-WRD. Moreover, the numbers of actual humans which have been clustered as Web robot are larger for NN-WRD. But more scrutinizing of such human sessions shows that all the human sessions identified as robots by FRS-WRD have accessed to *robots.txt* file, while among the actual humans detected as Web crawlers by NN-WRD, there are some actual human users which did not have access to this file. Therefore, they may show the larger false detection rate of NN-WRD.

It shows that dynamic feature selection implemented in FRS-WRD can cause a better characterization of Web sessions and increase the final efficiency of detection.

6 Conclusion and future work

To date, in the field of Web robot detection, many prestigious studies have been devoted to introducing new features for describing Web sessions. However, different nature of Web sites and mutable behavior of Web visitors can have notable influences on the effectiveness of features selected for describing the sessions of a Web site. To cope with this inherent vagueness present in Web data, we propose a novel algorithm called FRS-WRD which is based on fuzzy rough set theory to handle the noisy and vague data and select more relevant features for Web sessions. In fact, the main novelty of FRS-WRD is to dynamically select the features used to describe Web visitors to not only enhance the detection performance, but also eliminate the curse of dimensionality.

Experimental results reveal that despite many reliable features ever being introduced in related literatures, dynamically choosing the attributes for each data set is so effective in final performance. The results are reported in the form of *G*-mean, Jaccard, entropy, and purity rates, which are superior for the proposed algorithm to those of the state-of-the-art ones. Moreover, according to the confusion matrixes, FRS-WRD can better detect malicious Web visitors in practice.

In future studies, we are interested in examining other clustering algorithms which can handle huge, vague, and noisy data efficiently. Furthermore, although FRS has notable performance in dynamic feature selection for Web robot detection, more studies to define other novel attributes based on navigational patterns and Web server workloads can be promising.

Compliance with ethical standards

Conflict of interest The authors declare that they have no conflict of interest

Ethical approval This article does not contain any studies with animals performed by any of the authors.

Appendix

In this section, a summary of all primary features used in this paper is presented. These attributes have been proposed in other related works and indicated to be helpful in separating humans from Web robots. The index column of Table 5 demonstrates whether the related attribute has higher value for Web robots (R) or human users (H).

References

Amigó E, Gonzalo J, Verdejo F (2013) A general evaluation measure for document organization tasks. In: Proceedings of the 36th international ACM SIGIR conference on Research and development in information retrieval, ACM, pp 643–652

Ansari ZA, Sattar SA, Babu AV (2015) A fuzzy neural network based framework to discover user access patterns from web log data. Adv Data Anal Classif. doi:10.1007/s11634-015-0228-4

Antoine V, Quost B, Masson M-H, Denoeux T (2014) CEVCLUS: evidential clustering with instance-level constraints for relational data. Soft Comput 18(7):1321–1335

Bomhardt C, Gaul W, Schmidt-Thieme L (2005) Web robot detection-preprocessing web logfiles for robot detection. In: Bock HH et al (eds) New developments in classification and data analysis. Springer, Berlin, pp 113–124

Chen D, Yang W, Li F (2008) Measures of general fuzzy rough sets on a probabilistic space. Inf Sci 178(16):3177–3187

Dubois D, Prade H (1990) Rough fuzzy sets and fuzzy rough sets*. Int J Gen Syst 17(2–3):191–209

Gržinić T, Mršić L, Šaban J (2015) Lino-an intelligent system for detecting malicious web-robots. In: Asian Conference on Intelligent Information and Database Systems, Springer International Publishing, pp 559–568

Hamidzadeh J (2015) IRDDS: instance reduction based on distance-based decision surface. J AI Data Min 3(2):121–130

Hamidzadeh J, Monsefi R, Yazdi HS (2014) LMIRA: large margin instance reduction algorithm. Neurocomputing 145:477–487
Hamidzadeh J, Monsefi R, Yazdi HS (2015) IRAHC: instance reduction algorithm using hyperrectangle clustering. Pattern Recogn 48(5):1878–1889
Inuiguchi M, Wu W-Z, Cornelis C, Verbiest N (2015) Fuzzy-rough hybridization. Springer Handbook of Computational Intelligence. Springer, Berlin
Kohonen T (2013) Essentials of the self-organizing map. Neural Netw 37:52–65
Kwon S, Oh M, Kim D, Lee J, Kim Y-G, Cha S (2012) Web robot detection based on monotonous behavior. In: Proceedings of the Information Science and Industrial Applications, vol 4. Springer-Verlag, pp 43–48
Lee J, Cha S, Lee D, Lee H (2009) Classification of web robots: an empirical study based on over one billion requests. Comput Secur 28(8):795–802
List of User-Agents (Spiders, Robots, Browser) (2015) Retrieved from http://www.user-agents.org/
Liu Z, Pan Q, Dezert J, Martin A (2016) Adaptive imputation of missing values for incomplete pattern classification. Pattern Recogn 52:85–95
Liu Z, Pan Q, Dezert J, Mercier G (2015) Credal c-means clustering method based on belief functions. Knowl Based Syst 74:119–132
Lourenço AG, Belo OO (2006) Catching web crawlers in the act. In: Proceedings of the 6th international Conference on Web Engineering, vol 263, ACM, pp 265–272
Lu W-Z, Yu S (2006) Web robot detection based on hidden Markov model. In: 2006 International Conference on Communications, Circuits and Systems
Moghaddam VH, Hamidzadeh J (2016) New Hermite orthogonal polynomial kernel and combined kernels in support vector machine classifier. Pattern Recogn 60:921–935
Nowicki RK, Nowak BA, Woźniak M (2016) Application of rough sets in k nearest neighbours algorithm for classification of incomplete samples. In: Knowledge, Information and Creativity Support Systems. Springer International Publishing, pp 243–257
Pawlak Z (1982) Rough sets. Int J Comput Inf Sci 11(5):341–356
Qian Y, Wang Q, Cheng H, Liang J, Dang C (2015) Fuzzy-rough feature selection accelerator. Fuzzy Sets Syst 258:61–78
Radzikowska AM, Kerre EE (2002) A comparative study of fuzzy rough sets. Fuzzy Sets Syst 126(2):137–155
Sadeghi R, Hamidzadeh J (2016) Automatic support vector data description. Soft Comput. doi:10.1007/s00500-016-2317-5
Shafer G (1976) A mathematical theory of evidence, vol 1. Princeton University Press, Princeton
Sisodia DS, Verma S, Vyas OP (2015) Agglomerative approach for identification and elimination of web robots from web server logs to extract knowledge about actual visitors. J Data Anal Inform Process 3(2):1–10
Staeding A (2015) Bots versus browsers—public bots and user agents database and commentary. Retrieved from http://www.botsvsbrowsers.com/
Stassopoulou A, Dikaiakos MD (2009) Web robot detection: a probabilistic reasoning approach. Comput Netw 53(3):265–278
Stevanovic D, An A, Vlajic N (2012) Feature evaluation for web crawler detection with data mining techniques. Expert Syst Appl 39(10):8707–8717
Stevanovic D, Vlajic N, An A (2013) Detection of malicious and non-malicious website visitors using unsupervised neural network learning. Appl Soft Comput 13(1):698–708
Suchacka G, Sobkow M (2015) Detection of internet robots using a Bayesian approach. In: Cybernetics (CYBCONF), 2015 IEEE 2nd International Conference on, IEEE, pp 365–370
Tan P-N, Kumar V (2002) Discovery of web robot sessions based on their navigational patterns. Data Min Knowl Disc 6(1):9–35
Verbiest N, Cornelis C, Herrera F (2013a) FRPS: a fuzzy rough prototype selection method. Pattern Recogn 46(10):2770–2782
Verbiest N, Cornelis C, Herrera F (2013b) OWA-FRPS: a prototype selection method based on ordered weighted average fuzzy rough set theory. In: International Workshop on Rough Sets, Fuzzy Sets, Data Mining, and Granular Computing, vol 8170. Springer, Berlin, pp 180–190
Vlajic N, Card HC (2001) Vector quantization of images using modified adaptive resonance algorithm for hierarchical clustering. IEEE Trans Neural Netw 12(5):1147–1162
Wang Xi-Zhao, Zhai Jun-Hai, Shu-Xia Lu (2008) Induction of multiple fuzzy decision trees based on rough set technique. Inf Sci 178(16):3188–3202
Wu W-Z, Leung Y, Zhang W-X (2002) Connections between rough set theory and Dempster–Shafer theory of evidence. Int J Gen Syst 31(4):405–430
Yao YY, Lingras PJ (1998) Interpretations of belief functions in the theory of rough sets. Inf Sci 104(1):81–106
Zabihi M, Jahan MV, Hamidzadeh J (2014a) A density based clustering approach for web robot detection. In: Computer and Knowledge Engineering (ICCKE), 2014 4th International eConference on, IEEE, pp 23–28
Zabihi M, Jahan MV, Hamidzadeh J (2014b) A density based clustering approach to distinguish between web robot and human requests to a web server. ISC Int J Inf Secur 6(1):77–89
Zadeh LA (1974) The concept of a linguistic variable and its application to approximate reasoning. Springer, Berlin
Zhai J (2011) Fuzzy decision tree based on fuzzy-rough technique. Soft Comput 15(6):1087–1096
Zhao D, Traore I, Sayed B, Lu W, Saad S, Ghorbani A, Garant D (2013) Botnet detection based on traffic behavior analysis and flow intervals. Comput Secur 39:2–16