Immune Clonal Feature Selection and Under-Sampling Ensemble for Web Spam Detection

Xiaoyong Lu[1], Musheng Chen[2,*], Jhenglong Wu[3] ,Peichan Chang[3]

1 Software School
Nanchang University
Nanchang, P. R. China
2 School of Information Engineering
Nanchang University
Nanchang, P. R. China
3 Information Management
Yuan Ze University
Taoyuan, Taiwan

***Abstract* — Web spamming is nowadays a serious problem for search engines. It not only degrades the quality of search results by intentionally boosting undesirable web pages to users, but also causes the search engine to waste a significant amount of computational and storage resources in manipulating useless information. In this paper, we present an ensemble classifier for web spam detection combining immune clonal feature selection algorithm and under-sampling technique. At first, we resample the dataset with under-sampling technique to convert unbalanced dataset into balanced one. Second, we select several optimal feature subsets using a customized immune clonal feature selection algorithm. Third, we train an ensemble C4.5 decision tree classifier based on the selected feature subset and balance datasets. At last, we classify the samples in the test dataset with the ensemble C4.5 decision tree classifier. Experiments on WEBSPAM-UK2006 dataset show that our proposed approach contributes to more accurate classification compared with several state-of-the-art baselines classification models.**

Keywords - Web spam detection; Ensemble learning; Immune clonal algorithm; Feature selection; Decision trees

I. INTRODUCTION

Web spam can be defined as websites or web pages which have good search engine rankings not consistent with their true values. The phenomenon of web spam takes place mainly due to the fact that the web users tend to browse only the top ranked search engine results. Therefore, many website owners attempt to obtain a higher search engine ranking by unethical ways. Web spam weakens trust of the search engine users, wastes a significant amount of computational and storage resources, deprives legitimate websites of revenue, and deteriorates the quality of search results [1]. Web spam detection has been one of the top important tasks for web search engines.

Generally, there are three kinds of web spam: link spam, content spam, and usage spam [2]. Similarly, researchers extract amount of features based on the web pages' content, links and usage log. In order to identify junk pages, search engines can decrease the spamming page's ranking score calculated based on its features. However, Web spam takes various forms and lacks a consistent definition. Once the search engines obtain an effective algorithm based on specific features to detect a specific web spam, a new spamming method may appear soon. In order to avoid this problem and take full advantage of the extracted features, researchers use machine learning techniques, e.g. decision tree classification methods, to create models based on all of these features to detect spam pages.

Nowadays, numerous features are extracted for web spam detection. When training traditional classifiers such as decision trees, it can be found that they are efficient in few of features, but fail to provide meaningful results when amount of features are adopted. This occurrence is known as the curse of dimensionality or the Hughes phenomenon [3]. How to combine the features to train a more meaningful classification model and improve its performance has been a challenging task. One of the feasible methods is to obtain optimal feature subsets by feature selection and each feature subset can be used to train a classification model. Filter method [4], wrapper method [5] are used to search optimal feature subsets. However, numerous problems related to feature selection have been shown to be NP-hard. Many heuristic algorithms, including simulated annealing [6], ant colony optimization [7], genetic algorithms [8] etc. are proposed to find suboptimal solutions. Artificial immune clonal selection algorithms, addressed in this paper, also have been used as heuristic algorithms for feature selection and achieved better performance than other methods [9-10].

The number of spam pages is huge and growing on the Internet. However, it is still the minority as compared to the number of normal pages. It means that datasets for web spam detection are usually imbalance. Most of the classification methods, including decision tree methods, do not work well for class imbalance problem. In order to tackle this problem, lots of solutions have been presented, such as resampling [11], cost sensitive analysis [12], Kernel-based classifier [13] and so on. As a resampling

technique, under-sampling is an effective method to convert an unbalanced data set into a balanced one and is adopted in this paper.

In this paper, we propose an ensemble C4.5 decision tree classifier based on under-sampling technique and immune clonal features selection algorithm in web spam detection. Under-sampling technique is adopted to sample the imbalance web spam dataset into several balance datasets to solve the class imbalance problem. Immune clonal algorithm for feature selection extracts several optimal feature subsets.

II. LITERATURE REVIEW

A. Web Spam Detection Problem.

In general, web spam can be classified into three categories: content spam, link spam and usage spam. According to these three categories, three kinds of features, content-based, link-based and usage-based features, can be extracted from web pages and web server logs to identify spam pages [2]. In order to take full advantage all kinks of the features form promoting the performance of web spam detection, machine learning techniques, including supervised, unsupervised and semi-supervised learning methods, tend to be used. Scarselli et al. [14] presented a cascade architecture containing a probabilistic mapping graph self-organizing map and a graph neural network to detect web spam. The experiments on WEBSPAM-UK2006 showed that the results reached the state of the art when compared with some of the best results obtained by other quite different approaches. An efficient fuzzy clustering method was presented by Jegadeesh [15] to detect spam web pages. Wang et al. [16] proposed two new semi-supervised learning algorithms integrating the traditional co-training with the topological dependency based hyperlink learning to boost the performance of web spam classifiers.

B. Heuristic Feature Selection.

Feature selection refers to the identification of the appropriate features that should be introduced in the analysis in order to maximize the expected performance of the resulting model. The feature selection problem is an optimization problem to search throughout the space of feature subsets to identify the optimal or near-optimal subset with respect to the performance measure of its ability to classify the samples [9]. It is a well-studied problem since the selection of the right set of features for classification is one of the most important problems in designing a good classifier.

There are two approaches that can be used to address the feature selection problem: the filter method, the wrapper method. The filter method relies on general data characteristics to evaluate and select feature subsets without the use of a mining algorithm. Bonev et al. [4] proposed a novel feature selection filter for supervised learning, which relies on the efficient estimation of the mutual information between a high-dimensional set of features and the classes. The wrapper method requires one predetermined mining algorithm and uses its subsequent performance as the evaluation criterion. Moustakidis et al. [5] proposed a wrapper feature selection method, in the context of support vector machines (SVMs), named Wr-SVM-FuzCoC.

Numerous problems related to feature selection have been shown to be NP-hard. Therefore, many heuristic algorithms are used for feature selection to obtain optimal or near-optimal feature subsets. Lin et al. [6] proposed SVM classifier based on optimal parameters search and feature selection by simulated annealing. Ahmed [7] presented a novel feature subset search procedure that utilizes the Ant Colony Optimization (ACO) for two different classification problems. Ahmad et al. [8] proposed a genetic algorithm (GA) for simultaneous feature selection and parameter optimization of an artificial neural network (ANN) with application to an automatic breast cancer diagnosis.

C. Artificial Immune Clonal Algorithm for Feature Selection.

The artificial immune clonal algorithm is inspired by the clonal selection and affinity maturation process of B cells . B cells are a type of lymphocytes that are responsible for identifying and killing pathogens. Pathogens are foreign bodies including viruses, bacteria, multi-cellular parasites, and fungi. The clonal selection algorithm has been used for many real world problems. One of the main algorithms based on clonal selection theory is the CLONALG algorithm [17]. The CLONALG algorithm is used for a binary character recognition task, for a multi-modal optimization task and for solving a 30 cities instance of the travelling salesman problem. Dudek et al. [18] proposed a point symmetry-based clonal selection clustering algorithm (PSCSCA) to cluster data set with the character of symmetry.

Clonal Selection Algorithm is a fast optimization algorithm and is used for feature selection usually. In the immune clonal algorithm for feature selection, an antibody corresponds to a solution while an antigen represents the optimization problem. The degree of binding between the antibody and the antigen represents the objective function to be optimized. The objective is to start from an initial population of solutions (antibodies) and by using the algorithm iteratively to improve the quality of the solutions in the population. Samadzadegan et al. [10] used immune clonal selection optimization algorithm for feature selection and one of the fastest artificial immune classification algorithms to compute fitness function of the feature selection. The comparison of the feature selection results with genetic algorithm shows that the clonal selection has higher performance to solve selection of features. Marinaki and Marinakis [9] proposed a hybridized version of the clonal selection algorithm, the clonal selection algorithm–iterated local search–variable neighborhood search (CSA–ILS–VNS), for the solution of the feature selection problem to test the algorithm using various benchmark data sets from the UCI machine learning repository. The algorithm is compared with a particle swarm optimization algorithm, an ant colony optimization algorithm and a genetic algorithm and the experimental results show that it has a better performance.

III. A ENSEMBLE C4.5 DECISION TREE BASED ON IMMUNE CLONAL FEATURE SELECTION AND UNDER-SAMPLING

The framework of our web spam detection method is shown in Fig.(1). The training phase includes three procedures. The first one is to select several optimal features subsets with immune clonal algorithms. The second one is to convert the unbalanced dataset into balanced one with under-sampling technique. The Third one is to train an ensemble C4.5 decision trees classifier with the optimal feature subsets and under-sampling data subsets. We name the ensemble classifier as immune clonal feature selection and under-sampling ensemble C4.5 classifier (ICFSUS-EC4.5). In testing phase, the ICFSUS-EC4.5 classifier is used to evaluate testing data as well as unknown samples. The immune clonal feature selection algorithm is a wrapper method and the evaluation function also uses the under-sampling technique to balance dataset and build ensemble C4.5 decision trees classifiers. So the under-sampling technique is introduced at first as following.

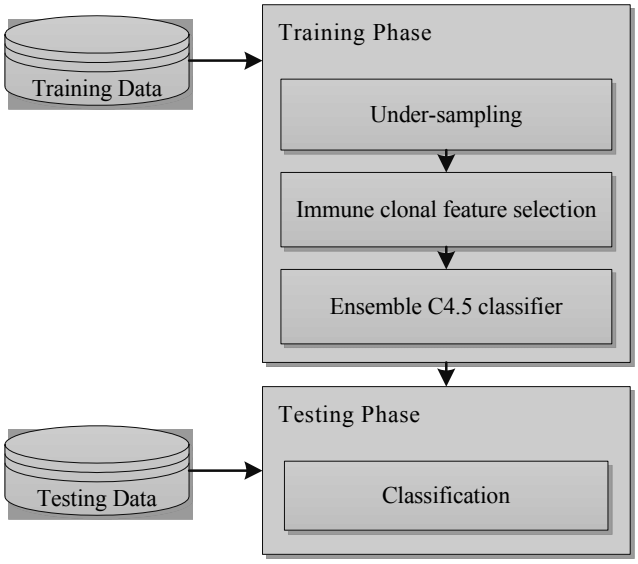

Figure 1. The framework of ICFSUS-EC4.5 classifier.

A. Under-Sampling.

In this paper, we want to solve imbalance problem of dataset by using an under-sampling method. Given the minor sample set S and the major sample set N, under-sampling randomly samples a subset N' from N, where n'<n and n represents the sample number in set N while n' represents the sample number in set N'. Since under-sampling uses only a subset of the major class samples with minor class samples to train the classifier, the training process is very efficient for learning algorithms that do not consider class-imbalance. We set the number of N' approximately equal to the number of S (the number of S is s), and divide all of the major sample set N into several sample subsets which contain about s samples randomly. As a result, the sampling ratio r≈s/n and k=round(n/s) subsets are obtained. Suppose that sample subsets N1', N2',…, Nk' are acquired from N. A classifier C is trained using sample subset Di={S, Ni}, i=1, 2, …, k. The k sub-classifiers are generated by C4.5 algorithm using individual sample subset Di. The under-sampling algorithm is shown below as Figure 2.

```
Input: Imbalance sample set with two classes: minor class samples set S and major class samples set N.
Output: several balanced sample subsets Di, i =1, 2, …, k
Begin
        s = the number of minor class samples
        n = the number of major class samples
        k = round(n/s)
        Divide the major samples into k sample subsets N1', N2', …, Nk', the number of each samples subset Ni' is approximately equal to s.
        Combine each sample subset Ni' and minor class samples S into a new sample subset Di.
        Return all the sample subset Di = {D1, D2,…,Dk}
End
```

Figure 2. Pseudo code of under-sampling algorithm.

B. Immune Clonal Feature Selection.

In the immune clonal feature selection algorithm, B cell and antibody correspond to the optimal feature subsets

while the antigen represents the optimization problem to search optimal feature subsets. Discovering an optimal feature subset by using immune clonal algorithm is similar to the process of searching an optimal antibody to recognize and kill an antigen. The objective function to compute the affinity between an antibody and the antigen is also an ensemble classification algorithm based under-sampling technique which returns the AUC value of the classification.

The process of the algorithm is shown in Fig.3. At first, the candidate population of the antibodies will be prepared in an initialization process before an iterative operation for clonal selection starts. In each of the iterative process, the first operation is to calculate the affinity between each antibody (a candidate solution representing a selected feature subset) and antigen (the problem to select several optimal feature subsets), then sort the antibodies by the affinity value descending and delete other antibodies from the candidate population in contrast preserving the prior antibodies. The prior antibodies will be cloned and mutated to get inserted into the candidate population of the antibodies. In order to diversify the antibodies, a suppression operator will be executed to insert some new diverse antibodies into the population to overcome early constringency.

In the clonal selection algorithm for feature selection, we use binary digits for features representation to apply clonal selection algorithm to the feature selection problem. The bits consisted of digit 0 and digit 1, which correspond to non-selected and selected features, respectively. Each cell was coded as binary alphabetical string. For example, a cell [1 0 1 0 0] contains five features where only the first and the third features were selected.

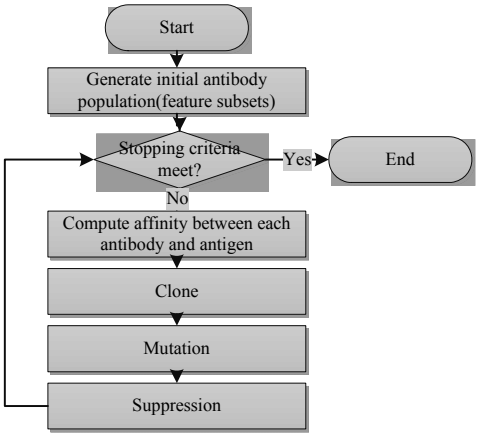

Figure 3. flow chart of immune clonal feature selection.

Step 1: Generate initial antibodies population

Initially, we have to choose the population of the antibodies. Each antibody is randomly placed in the d-dimensional space as a candidate solution (in the feature selection problem d corresponds to the number of activated features). The initialization formulation is shown as in Eq. (1):

$$P = round(rand(p, f)) \quad (1)$$

where P is the population of antibodies; p represents the number of the antibodies in the population, f represents the number of the features in each antibody. The rand(p,f) will return a double array of p by f, each element in the double array is in the range of (0,1). Round operation will force the double value to an integer. Because each element in the double array is in the range of [0,1], each element in antibody population will be assigned 0 or 1.

Step 2: Affinity computation

The fitness function in immune clonal algorithm is a measure of the fitness of a solution to the objective function. According to the antibody, which represents the selected features, the fitness value was obtained from AUC, which indicate the classification accuracy in Web spam detection. The classification method used for web spam detection is an ensemble C4.5 classifier based on under-sampling technique. In the process of selecting optimal feature subsets, for each feature subset, each sample in the training dataset is assigned into one of the k samples fold and classified by the ensemble C4.5 classifier. The spamicity value of each sample can be returned from the ensemble C4.5 classifier. Based on the spamicity value of each sample, the ROC AUC value represented as the affinity between the antibody and antigen value can be calculated. The pseudo code of the fitness function to compute the affinity between the antibody and antigen is shown in Fig.4.

Step 3: Antibody clonal selection

From the antibodies population P, the best solutions L are selected based on the sorting of the antibodies by the affinity value. The selected antibodies are preserved and other candidate antibodies in the antibodies population P are deleted. The preserved antibodies will be cloned and mutated to construct new candidate population of antibodies. Each antibody in the best solutions L generates lc cloned antibodies proportional to their affinities. The fittest antibodies create a larger number of cloned antibodies. The number of cloned antibodies is given by the following:

$$l_c = \sum_{i=1}^{l} round(\frac{\beta l}{i}) \quad (2)$$

where β is a multiplying factor; l is the number of best solutions; lc is the number of total cloned antibodies from all of the best solutions. The antibodies cloned from each best solution are numbered in decreasing order of its fitness. For example, if l is equal to 7 and β is equal to 2, the first (i = 1) antibody will produce lc1 equal to 14 cloned antibodies, the second antibody will produce lc2 equal to 7 cloned antibodies, and so on.

Step 4: Antibody mutation

Each cloned antibody will be changed by a mutation operator. Because the antibodies are represented in the form of bit strings, the mutation of the antibodies can be implemented by changing the bits in the bit string. The bit string of each antibody is changed from two directions on adding or deleting bits. The heuristic rules of the mutation operator are originated from following two hypotheses:

(1) Adding a few features into an optimal feature subset to obtain a new feature subset, the new feature subset will be more accurate for classification than the original feature subset.

(2) Deleting a few features from an optimal feature subset to obtain a new feature subset, the new feature subset will be more accurate for classification than the original feature subset.

In order to confirm the uniqueness of the antibody, it is required to validate whether the new antibody is included in the candidate antibodies population. If it is, the mutation operation is executed again to create a new antibody, otherwise, the antibody will append into the candidate population of the antibodies.

Step 5: Antibody suppression

The aim of the suppression operator is to diversify the population of the antibodies. The suppression operator is similar to the initialization operator, which is to generate p new antibodies randomly and append them into the candidate antibodies population. Similar to the antibodies generated by clonal and mutation operators, it is required to validate their uniqueness.

```
Input: training set D, binary alphabetical string S indicating a selected feature subset, a integer n indicating n fold cross validation
Output: the ROC AUC value
Begin
    Extract the training set D based on the binary alphabetical string S into a new training subset D'.
    Split the training subset into two sample subsets: minor sample subset DS and major sample subset DN.
    Divide the minor sample subset into n parts uniformly DSi {DS1, DS2, …, DSn}, the same operation as the major sample subset DNi {DN1, DN2, …,
DNn}.
    Merge each DSi and DNi serially into datasets DMi {DM1, DM2, …, DMn}.
    For each DMi
        Consider DMi as testing set DMte, other samples in DM as training set DMtr.
        Achieve several balance sample subsets DBi{DB1,DB2,…,DBk} by sampling the training set DMtr using the under-sampling technique.
        Initialize the spamicity of each sample in the testing set DMte: spamicity=0;
        For each DBi
            Train a C4.5 decision tree model with the balance sample subset DBi
            Classify the samples in the testing set DMte, the classification result value (result)of each sample is 1 (spam) or -1 (normal).
            accumulate the spamicity of each sample in the testing set DMte: spamicity=spamicity+result
        End for
    Calculate the average of the spamicity: spamicity = spamicity / k
    End for
    Compute the ROC AUC value based on the spamicity of all samples in the testing set.
End
```

Figure 4. pseudo code of the fitness function in immune clonal feature selection algorithm.

C. Ensemble C4.5 Decision Trees Classifier

After obtaining n optimal feature subsets with immune clonal feature selection algorithm and k balanced datasets, n*k balanced datasets can be achieved with features projection based on the optimal feature subsets and resampling based on under-sampling technique. Based on each of the n*k balanced dataset, we can train a C4.5 decision tree classifier. Integrating all of the C4.5 decision tree classifiers, we can obtain an ensemble classifier.

D. Testing Pphase

With the ensemble classifier, we can test all of the samples in the testing dataset. When testing, each of the C4.5 decision tree in the ensemble classifier classifies the testing samples and achieves classification results. The classification results can be calculated by the following:

$$Score(x,C) = \begin{cases} 1, testing\ \ host\ is\ sp\ am \\ -1, testing\ ho\ st\ is\ norm\ al \end{cases} \quad (3)$$

where x is a sample in the testing dataset; C is a C4.5 decision tree classifier; Score(x,C) is the classification result that sample x is classified by classifier C, we name it as score of the classification result. Summarizing all of the testing sample's classification results viz. its scores obtained by C4.5 decision trees in the ensemble classifier and averaging them, we can achieve a classification result for each sample in a range of [-1,1]. The classification result for each website can be called spamicity. The spamicity value for each testing sample can be used to compute the ROC AUC value of the ensemble classifier. At the same time, the final classification result of each sample can be calculated by the following:

$$Classification\,\mathrm{Re}sult = \begin{cases} 1, spamicity > 0 \\ -1, spamicity \le 0 \end{cases} \quad (4)$$

The final classification result of the sample x is 1, it is classified as spam, and otherwise, it is classified as normal.

IV. EXPERIMENTAL RESULTS

In this section, we have designed a series of testing for our proposed ICFSUS-EC4.5 ensemble classifier to evaluate web spam detection problem

A. Dataset and Evaluation Metrics

All of the experiments are conducted on WEBSPAM-UK2006 dataset [19], which is a publicly available collection introduced by the Web Spam Challenge and the Adversarial Information Retrieval on the Web workshop (AIRWeb) 2007. Such a collection of pages was particularly suitable for our purposes, since the dataset is large in size and it has been used by other research teams. Moreover, the dataset comes with a predefined splitting of the dataset into a training dataset and a testing dataset and it includes a set of precomputed features, which simplifies the pre-processing procedures. In order to compare with other state-of-the-art algorithms, we adopt four kinds of the features: content-based features which has 96 variables, link-based features which has 41 variables, link-transformed features which has 135 variable, and neighborhood-based features which has 2 variables. The numbers of spam and non-spam hosts on WEBSPAM-UK2006 are displayed in table 1. We can find that the ratio between non-Spam hosts and spam hosts in the training dataset is approximately 7:1. It means that the training dataset is imbalance and it is in line with the actual situation.

TABLE I THE EXPERIMENTAL DATA OF WEBSPAM-UK 2006

Dataset	Hosts number	Spam hosts number	Non-spam hosts number
Training set	5,622	674	4,948
Testing set	1,851	1,250	601
Total	7,473	1,924	5,549

We use three metrics to evaluate our model: Accuracy, F1-Measure, and ROC AUC. Each of them is commonly used in information retrieval and in data mining. In this paper, we consider web spam detection as a binary-classification problem. For a binary-classification problem, if a testing instance belongs to the target class, it is classified as a positive instance, otherwise, it is classified as a negative instance. After all the testing instances are classified, the confusion matrix of the classification can be obtained. In the confusion matrix, TP is the number of true positive instances returned in response to a classification. Similarly, TN, FP and FN are the number of true negatives, of false positives and of false negatives, respectively. Then the Accuracy, F–Measure are defined as shown as in Eq. (5) and Eq. (6).

$$Accuracy = \frac{TP + FN}{TP + FP + FN + TN} \tag{5}$$

$$F1 - Measure = \frac{2TP}{2TP + FN + FP} \tag{6}$$

Theoretically, the ROC AUC value is equal to the probability that a classifier will rank a randomly chosen positive instance higher than a randomly chosen negative one (assuming positive ranks higher than negative) [20]. Therefore, it is closely related to the Mann–Whitney U [21]. It is also equivalent to the Wilcoxon test of ranks [22]. The ROC AUC has the advantage over the F-measure as being independent of the threshold that must be applied on the score in order to decide whether an instance has been classified as positive or negative.

B. Arameter Settings

When applying the ICFSUS-EC4.5 Classifier in web spam detection, several parameters need to setup. The first parameter is the number of initial antibody population. It is also the number of antibodies generated by suppression operator. Because the new antibodies will be generated in each generation and the number of them is not so critical to search optimal feature subsets, we empirically set it to 20. The second parameter is the number of folds for cross-validation when computing the affinity for each antibody in the training phase. It will change the number of the training set and then change the classification results. In order to ensure the number of samples in the training set, it cannot be too low. On the other hand, if it is too high, it is required to train more classifiers and the training time greatly extend. Empirically, we set it to 5. The third parameter is the number of generation. We found that it will be convergent before 100th generation. Therefore, we set the number of generations to 100. The last and most important parameter is the number of optimal feature subsets. It is also the number of optimal antibodies to be cloned. Empirically, we set it to an odd number (i.e. 3, 5, 7, 9, 11, or 13). For each of the odd numbers, we execute the experiment 30 times. From the experimental results, we found that the optimal number of antibodies is 7. Therefore, we set the value of the parameter denoting the number of optimal antibodies as 7.

C. Comparisons of Different Approaches.

We take the output of the ICFSUS-EC4.5 Classifier and compare the average result with the output from traditional classification model deprived of C4.5. The traditional classification models include C4.5, Bagging with C4.5 (C4.5+bagging), Adaboost with C4.5 (C4.5+AdaBoost), and Under-sampling based ensemble classifier with C4.5 (C4.5+US). All of the experimental results are shown in table 2. Comparing and analysing the Accuracy, F1-Measure and ROC AUC results in Table 2, we can find that: the ICFSUS-EC4.5 classifier outperforms the other classification models deprive from C4.5. In the traditional classification models deprived of C4.5, the Under-sampling based ensemble classifier with C4.5 (C4.5+US) has obtained better results than others: Its Accuracy is 0.8358, its F1-Measure is 0.8717, and its ROC AUC value is 0.9135. However, compared with the ICFSUS-EC4.5 classifier, its metric values of its classification results are much less.

Table 3 compares the result achieved by our approach with the teams that participated in the web spam challenge

on 2007 . The table shows that our approach obtains the best F1-Measure and the second best ROC AUC. The F1-Measure value of the ICFSUS-EC4.5 ensemble classifier is 0.93 (rounded from 0.9254) is higher than Benczur et al.'s F1-Measure value which is the best value in web spam challenge 2007. The ROC AUC value of the ICFSUS-EC4.5 classifier is 0.95 (rounded from 0.9487) and it is only slightly less than the best value achieved by Cormack. However, Cormack's F1-Measure value is quite low (only 0.67) and much less than others. On the other hand, some of the competitors optimized the features selectively and added some features by including new ad hoc information, e.g., the value of Google AdSense, statistics about HTML, and data provided by email filters. Even so, the ICFSUS-EC4.5 classifier has achieved comparable classification results.

Scarselli et al. [14] presented a cascade architecture containing a probabilistic mapping graph self-organizing map (PM-G) and a graph neural network (GNN) to detect web spam. Table 4 shows the evaluation metrics of the experimental results obtained by four optimal cascade models presented. The results of FNN, PM-G+GNN(3)+GNN(1) reached the state of the art at that time. Therefore, we compared their results that the ICFSUS-EC4.5 classifier outperforms the FNN, PM-G+GNN(3) +GNN(1) approaches. Although the Accuracy of our approach (0.8973) is still lower than the FNN,PM-G+GNN(3) +GNN(1)'s Accuracy value (0.9294), the F1-Measure value is 0.9266 and the ROC AUC value is 0.9507 of our approach are much higher than FNN,PM-G+GNN(3)'s F1-Measure value is 0.6324 and ROC AUC value is 0.9362.

TABLE II COMPARISONS ON DIFFERENT MODELS

Approach	Accuracy	F1-Mesaure	ROC AUC
C4.5	0.7277	0.7609	0.7742
C4.5+Bagging	0.7763	0.8097	0.7849
C4.5+Adaboost	0.7509	0.7849	0.7931
C4.5+US	0.8358	0.8712	0.9135
ICFSUS-EC4.5	0.8973	0.9266	0.9507

TABLE III COMPARISONS WITH WEB SPAM CHALLENGE 2007

Team	F1-Measure	ROC AUC
Benczur et al., Hungarian Academy of Sciences	0.91	0.93
Filoche et al., France Telecom	0.88	0.93
Geng et al., Chinese Academy of Science	0.87	0.93
Abou-Assaleh et al., Genie Knows	0.81	0.80
Fetterly et al., Microsoft search Labs	0.79	-
Cormack, University of Waterloo	0.67	0.96

T TABLE IV COMPARISONS WITH SCARSELLI ET AL.'S APPROACH

Approach	Accuracy	F1-Measure	ROC AUC
FNN, PM-G+GNN(3)	0.9124	0.5890	0.9236
GNN+GNN(1)	0.9070	0.4400	0.8103
Autoassociator+GNN(1)	0.9104	0.4173	0.8070
FNN,PM-G+GNN(3)+GNN(1)	0.9294	0.6324	0.9362

D. Discussion

Although the experimental results do not allow us to conclude which method is the best one, these results suggest that the performance of the ICFSUS-EC4.5 classifier is comparable with those obtained by other state-of-the-art techniques. We think our approach can get such a performance mainly due to the following reasons. The first one is that the under-sampling technique is used to sample the normal websites and the number of spam and normal websites in each sub training dataset is almost equivalent. Training in a balance dataset promotes the classification accuracy of the C4.5 decision tree classifier. Second, Ensemble classifier based under-sampling take full advantage of all of the samples in the training dataset. Finally, ensemble classifier based on immune clonal feature selection algorithm takes full advantage of all of the optimal feature subsets.

V. CONCLUSIONS

Web spam detection is an important topic in the field of information retrieval. Because web spam datasets are serious imbalance datasets, we propose an ensemble classifier based on under-sampling and immune clonal feature selection techniques. Experiments on WEBSPAM-UK2006 show that our method outperforms other approaches. Although the algorithm for web spam detection has got good performance in evaluation indices AUC and F1-Measure，its accuracy is still not so good enough and needs to be improved. Web spam detection is still one of the hard tasks in information retrieval on the Internet, although some state-of-the-art algorithms for web spam detection are designed and tested. Besides exploring new machine learning algorithms, it is also a traditional and normal way to extract new features from the web content, links and usage information. Apart from that, we also plan to further extend our research by performing

more experiments on other datasets and applications to test whether the ICFSUS-EC4.5 can be used generally.

ACKNOWLEDGMENT

This work is supported by the Science and technology support program of Jiangxi Province, China (No.20121102040073 and No.20131102040039).

REFERENCES

[1] Spirin N., Han J, "Survey on web spam detection: principles and algorithms",ACM SIGKDD Explorations Newsletter, vol. 13, pp. 50-64, 2012.

[2] Chandra A., Suaib M, "A Survey on Web Spam and Spam 2.0", International Journal of Advanced Computer Research , vol. 4, pp. 634-644, 2014.

[3] Tahir M.A., Bouridane A., Kurugollu F, "Simultaneous feature selection and feature weighting using Hybrid Tabu Search/K-nearest neighbor classifier",Pattern Recognition Letters , vol. 28, pp. 438-446, 2007.

[4] Bonev B., Escolano F., Cazorla M, "Feature selection, mutual information, and the classification of high-dimensional patterns", Pattern Analysis and Applications , vol. 11, pp. 309-319, 2008.

[5] Moustakidis S.P., Theocharis J.B, "A fast SVM-based wrapper feature selection method driven by a fuzzy complementary criterion",Pattern Analysis and Applications , vol. 15, pp. 379-397, 2012.

[6] Lin S., Lee Z., Chen S., Tseng T, " Parameter determination of support vector machine and feature selection using simulated annealing approach", Applied Soft Computing , vol. 8, pp. 1505-1512, 2008.

[7] Ahmed A. Feature subset selection using ant colony optimization", International Journal of Computational Intelligence and Applications , vol. 2, pp. 53-58, 20058.

[8] Ahmad F., Isa N.A.M., Hussain Z., Osman M.K., Sulaiman S.N, "A GA-based feature selection and parameter optimization of an ANN in diagnosing breast cancer",Pattern Analysis and Applications , vol. 5, pp. 1-10, 2014.

[9] Marinaki M., Marinakis Y, "A hybridization of clonal selection algorithm with iterated local search and variable neighborhood search for the feature selection problem",Memetic Computing ,2015;

[10] Samadzadegan F., Namin S.R., Rajabi M.A, "Evaluating the potential of clonal selection optimization algorithm to hyperspectral image feature selection", Key Engineering Materials , vol. 500, pp. 199-805, 2012.

[11] Yen S., Lee Y, " Cluster-based under-sampling approaches for imbalanced data distributions", Expert Systems with Applications , vol. 36, pp. 5718-5727, 2009.

[12] Sun Y., Kamel M.S., Wong A.K., Wang Y, "Cost-sensitive boosting for classification of imbalanced data. Pattern Recognition, vol. 40, pp. 3358-3378, 2007.

[13] Hong X., Chen S., Harris C.J, " A kernel-based two-class classifier for imbalanced data sets", IEEE Transactions on Neural Networks , vol. 18, pp. 28-41, 2007.

[14] Scarselli F., Tsoi A.C., Hagenbuchner M., Di Noi L, "Solving graph data issues using a layered architecture approach with applications to web spam detection", Neural Networks , vol. 48, pp. 78-90, 2013.

[15] Jegadeesh J.S., Jacob P.L, "Web spam detection using fuzzy clustering",International Journal on Recent and Innovation Trends in Computing and Communication , vol. 1, pp. 928-938, 2013.

[16] Wang W., Lee X., Hu A., Geng G, "Co-training based semi-supervised Web spam detection", 10th International Conference on Fuzzy Systems and Knowledge Discovery (FSKD), pp. 789-793, 2013.

[17] De Castro L.N., Von Zuben F.J, " Learning and optimization using the clonal selection principle", Evolutionary Computation, IEEE Transactions on , vol. 6, pp. 239-251, 2002.

[18] Dudek G, " An artificial immune system for classification with local feature selection", Evolutionary Computation, IEEE Transactions on , vol. 16, pp. 847-860, 2012.

[19] Castillo C., Donato D., Becchetti L., Boldi P., Leonardi S., Santini M., Vigna S, "A reference collection for web spam. ACM Sigir Forum , vol. 40, pp. 11-24, 2006.

[20] Fawcett T, "An introduction to ROC analysis", Pattern Recognition Letters , vol. 27, pp. 861-874, 2006.

[21] Hanley J.A., McNeil B.J, "The meaning and use of the area under a receiver operating characteristic (ROC) curve", Radiology, vol. 143, pp. 29-36, 1982.

[22] Mason S.J., Graham N.E., "Areas beneath the relative operating characteristics (ROC) and relative operating levels (ROL) curves: Statistical significance and interpretation", Quarterly Journal of the Royal Meteorological Society , vol. 128, pp. 2145-2166, 2002.