BULGARIAN ACADEMY OF SCIENCES

CYBERNETICS AND INFORMATION TECHNOLOGIES • Volume **18**, No 4
Sofia • 2018 Print ISSN: 1311-9702; Online ISSN: 1314-4081
DOI: 10.2478/cait-2018-0046

Spam Review Classification Using Ensemble of Global and Local Feature Selectors

Gunjan Ansari[1], *Tanvir Ahmad*[2], *Mohammad Najmud Doja*[2]

[1]*JSS Academy of Technical Education, C-20/1, NOIDA-201301, India*
[2]*Jamia Millia Islamia,Jamia Nagar, New Delhi-110025, India*
E-mails: gunjanansari@jssaten.ac.in tahmad2@jmi.ac.in ndoja@yahoo.com

***Abstract*:** *In our work, we propose an ensemble of local and global filter-based feature selection method to reduce the high dimensionality of feature space and increase accuracy of spam review classification. These selected features are then used for training various classifiers for spam detection. Experimental results with four classifiers on two available datasets of hotel reviews show that the proposed feature selector improves the performance of spam classification in terms of well-known performance metrics such as AUC score.*

***Keywords*:** *Feature selection, improved global feature selector, odds ratio, Spam classification.*

1. Introduction

E-commerce and online opinion sharing websites allow people to express their opinion regarding any product or service launched in the market. In these reviews, people share their real life experiences, which play a major role in decision making process of users while buying any product or booking any service. This widespread sharing and effect of customer's reviews has increased the chances of spam attacks. Spam reviews are written to deviate the customer's opinion thus increasing the sales of product or service. It is very difficult for the customer to analyze the difference between fake and genuine reviews, thus researchers are working to find out the linguistic difference between both for automatic spam classification.

The extraction of meaningful review and reviewer centric features from text to improve accuracy of supervised approaches is a major challenge in this area. In addition, use of big data technique is needed to address the issue of increasing reviews and opinions shared by customers at various sites [3, 8, 23]. Spam reviews are very small as compared to non-spam reviews leading to data imbalance problem in supervised learning approaches [20]. According to investigation in [7], the use of

ensemble learning methods with global feature rankers for spam detection lead to better accuracy. Recent study on text classification showed that integration of one-sided local feature selection method with filter-based global feature selection method could further increase the model accuracy [26].

In our proposed work, an ensemble of local and global feature selectors to reduce feature space is used to improve the accuracy of spam review classification. The approach is benefitted from the negative features that are produced by one-sided local feature selector. One-sided local feature selector or Odds ratio assigns positive or negative score to every feature with respect to a particular class. Negative or Positive score indicates the non-membership or membership of that feature to any class. Each feature is sorted on the basis of global feature ranker and while selecting the feature subset, local score with respect to a particular class is also taken into consideration so that selected feature subset represents all classes almost equally. The selected features are then used for training various classifiers. The evaluation on real and synthetic dataset of hotel reviews showed that the ensemble of local and global feature selection method outperforms global feature rankers in terms of well-known Area Under the Curve score (AUC) performance metrics.

The remainder of the paper is organized as follows: Section 2 contains background study in the area of spam detection, Section 3 provides details of the dataset, feature selectors, classifiers, proposed architecture and algorithm used in our study, Section 4 presents experimental results and performance evaluation of proposed work and Section 5 gives conclusion and future scope of our research.

2. Related work

The problem in review spam detection using supervised learning is lack of gold standard datasets. Thus, many of the researchers use manually annotated datasets for spam detection. An artificial dataset of fake hotel review was created for spam detection and supervised learning approach was applied on the created dataset. An automatic approach using features such as Genre identification, psycholinguistic feature using Linguistic Inquiry and Word Count (LICW), and text categorization considering unigram, bigram and trigram features was applied on the artificial dataset [21]. The combination of bigrams with LICW as features with Support vector machine when applied on this artificial dataset gives an accuracy of 89.8%.

A generative model of deception that jointly models the classifier's uncertainty as well as ground truth deceptiveness of each review was also proposed [22]. Using this method, they explored the prevalence of deception among positive reviews in six popular online review communities. The artificial data was further analyzed in the study [1] and it was found that writing style and readability of review are effective parameters for spam detection. The results obtained with logistic regression using these features showed an accuracy of 71.25% and misclassification rate of 28.49%. This dataset was further used in study [17] and using the ontological features they categorized reviews into non-review, brand-review, off-topic review and spam reviews. They tested their approach on three popular products and collected their

reviews from e-Commerce websites. Their results on this dataset showed a precision of 75%.

Positive Unlabeled (PU) learning method to detect fake reviews from Chinese review hosting site Dianping outperformed significantly and detects hidden fake reviews in the unlabeled set that Dianping could not find [14]. The lacking of Dianping was that it considers only review's side information, IP address etc. to detect spams but not text content. They considered two classes one as positive and other as unknown which can still be fake for training their classifier. Temporal and spatial features for supervised opinion spam detection [13] was analyzed on large-scale real-life dataset and results showed an accuracy of around 85%. The investigation in [7] showed that Select-Boost, Multinomial Naïve Bayes with Chi-Square test or Signal to noise ratio as feature selectors outperformed all methods except Random forest using 500 trees.

Apart from content based spam detection in hotel reviews much work has been done in the domain of product reviews collected from amazon dataset. The researchers investigated their work on 5.8 million reviews crawled from amazon [9]. The research included parameters as review rating and writing style of textual reviews for detecting spam. In [16], the research relied on review content and rating to define four different spamming behavioral models – targeting products, targeting group, general rating deviation, and early rating deviation. The other work in the area of spam identification [12] used semi supervised, co-training method of machine learning to identify spams.

A lightweight effective method using binomial test [25] to find the difference between spammer rating and majority opinion of reviewers was analyzed on downloaded data from amazon. A novel and principle method to exploit observational behavior footprints for spammer detection using unsupervised Bayesian framework was proposed in the work [19]. To detect review burstiness, Kernel density estimation was used in the study [5]. Graph based method in [28, 29] used reviewer's trustiness, honesty of review and reliability of store as parameters to detect spammers. Frequent item set data mining approach was used to find candidate sets for detecting group spammers [18].

Our approach of spam detection is based on the content of the reviews. However, many approaches of content based spam detection using classification has been proposed in the past, our approach is novel as it reduces features using integration of local feature selection with global feature selection as discussed in the Section 3.

3. Proposed approach for Spam classification

The architecture of the proposed work is shown in Fig.1. The steps are explained in the following subsections.

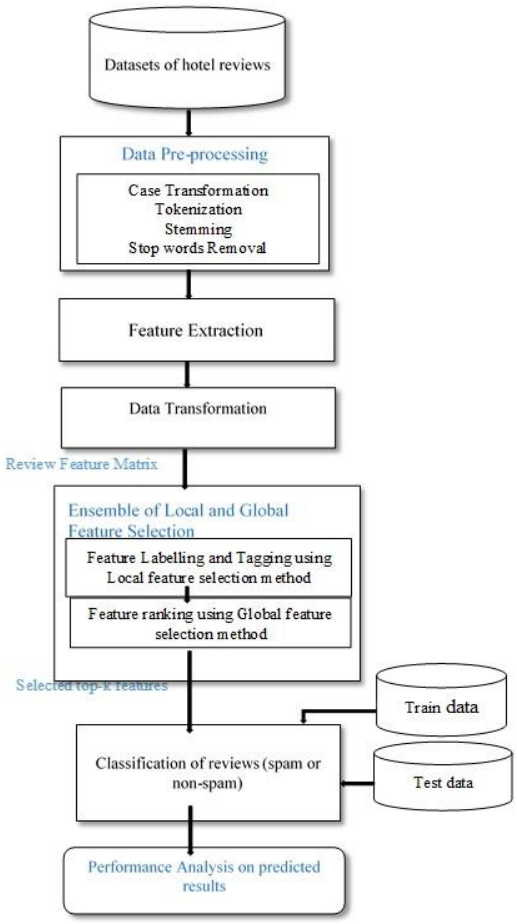

Fig. 1. System architecture of proposed framework

3.1. Datasets

In our work, the following two datasets of hotel reviews are used.

TripAdvisor Hotel Review dataset

This dataset is publically available and contains 800 genuine and 800 deceptive reviews. Due to lack of annotated dataset for spam detection this synthetic dataset has been created by researchers [21] for classification. The dataset consist of 400 positive and 400 negative truthful reviews collected from 20 popular hotels of TripAdvisor and 400 positive fake and 400 negative fake reviews generated from Amazon Mechanical Turk.

Yelp Filtered review dataset

As the first dataset used is synthetic and does not represent real world dataset, the other dataset used for our research is Yelp Filtered review dataset. In this dataset, reviews are collected from Yelp.com and the dataset is previously used by the researchers [24]. This dataset contains reviews from 5044 restaurants by 260,277 reviewers. Yelp has a filtering algorithm that identifies fake/suspicious reviews and separates them into a filtered list. The recommended reviews by yelp filter are

considered as genuine and filtered reviews are considered fake in the study. We only collected review text and label of Chicago hotels for the purpose of our research. There are total 5854 hotel reviews, out of which only 778 reviews are spam. For our work, we created a dataset consisting of 778 spam reviews and 800 non-spams reviews. These non-spam reviews are randomly sampled out of available 5076 non-spam reviews. The sampling is done to create a balanced dataset of both types of reviews.

3.2. Data Pre-processing

Preprocessing steps are applied on the content of reviews extracted from the datasets. This first phase consist of lowercase conversion, tokenization, stemming and stop words removal.

3.3. Feature extraction

All unigram and bigram features are extracted from pre-processed textual data. Unigram are single token and bigrams are sequence of two tokens extracted from the text of the review.

3.4. Data transformation

Data transformation phase is required to transform the data into the suitable form required for model construction. It transforms each review into a review-feature matrix using term frequency or term frequency – inverse term frequency of every extracted unigram or bigram feature in the corresponding review document.

3.5. Improved Global Feature Selection for spam classification (IGFS)

In the selection process each unigram and bigram features are assigned a score using score-computing function of global or local feature selection method. In our work of spam classification we integrate a local feature selection method with three different global feature selection methods. The local feature selection method used is Odds ratio and global feature selection methods used are Gini index, Information gain, and Distinguished feature selection. The mathematical definitions of the score-computing functions are discussed in the following subsections:

3.5.1. Odds Ratio (OR)

It is one-sided metric and is used to rank a feature with reference to a specific class. It reflects the odds of the term occurring in the positive class normalized by that of the negative class. The score of OR can be negative or positive. Negative score indicates non-membership of a feature to a class and termed as negative feature for that particular class [6]. The mathematical formula for OR of feature with respect to k-th class is as follows:

$$\text{OR}(f\text{eature}|C_k) = \log\frac{P(\text{feature}|C_k)[1-P(\text{feature}|\overline{C_k})]}{[1-P(\text{feature}|C_k)]P(\text{feature}|\overline{C_k})}, \quad (1)$$

where $1 \le k \le m$ and m is the number of classes; $P(\text{feature}|C_k)$ is Probability of a feature in presence of class C_k; $P(\text{feature}|\overline{C_k})$ is Probability of a feature in absence of class C_k.

3.5.2. Information Gain (IG)

It is two-sided global feature selection metric. The score of IG is obtained by the presence or absence of a term in a document for predicting the correct class of the document [15]. IG score is calculated using the following formula:

(2) $$\begin{aligned}\mathrm{IG}(\mathrm{feature}) = &-\sum_{k=1}^{m} P(C_k)\ \log P(C_k) + \\ &+P(\mathrm{feature})\sum_{k=1}^{m} P(C_k|\mathrm{feature})\ \log P(C_k|\mathrm{feature}) + \\ &+P(\overline{\mathrm{feature}})\sum_{k=1}^{m} P\left(C_k\middle|\overline{\mathrm{feature}}\right)\ \log P\left(C_k\middle|\overline{\mathrm{feature}}\right),\end{aligned}$$

where $1 \le k \le m$ and m is the number of classes; $P(C_k)$ is Probability of a class C_k and $P(\mathrm{feature})$ is Probability of feature; $P(C_k|\mathrm{feature})$ is Conditional Probability of class C_k given presence of feature; $P(\overline{\mathrm{feature}})$ is Probability of absence of feature; $P\left(C_k\middle|\overline{\mathrm{feature}}\right)$: Conditional Probability of class C_k given absence of feature.

3.5.3. Gini Index (GI)

GI is the modified version of attribute based feature selection and used as a feature selector for text classification problems. It is a global feature selection method and assigns a positive score to each feature. The maximum the score of the feature, the better is its rank [27]. The GI score is calculated using the following formula:

(3) $$\mathrm{GI}(\mathrm{feature}) = \sum_{k=1}^{m} \mathrm{P}(\mathrm{feature}|\mathrm{C_k})^2\ \mathrm{P}(\mathrm{C_k}|\mathrm{feature})^2,$$

where $1 \le k \le m$ and m is the number of classes; $P(C_k|\mathrm{feature})$ is Conditional Probability of class C_k given presence of feature; $P(\mathrm{feature}|C_k)$: Conditional Probability of feature given presence of class C_k.

3.5.4. Distinguished Feature Selector (DFS)

DFS is a filter-based global feature selection method that assign score to each features based on the discriminating power of that feature. The feature selector filters the uninformative or redundant features from the generated set of features. The feature which are assigned high score are distinctive and are considered to be best feature for text classification [27]. The score assigned to a feature using DFS is computed as follows:

(4) $$\mathrm{DFS}(\mathrm{feature}) = \sum_{k=1}^{m} \frac{P(C_k|\mathrm{feature})}{\left[P\left(\overline{\mathrm{feature}}\middle|C_k\right)+P\left(\mathrm{feature}\middle|\overline{C_k}\right)+1\right]},$$

where $1 \le k \le m$ and m is the number of classes; $P(C_k|\mathrm{feature})$ is Conditional Probability of class C_k given presence of feature; $P\left(\overline{\mathrm{feature}}\middle|C_k\right)$ is Conditional Probability of absence of feature given presence of class C_k; $P(\mathrm{feature}|C_k)$ is Conditional Probability of feature given presence of class C_k.

3.6. Classification methods

The commonly used classifiers for text classification problem used in our work are Multinomial Naïve Bayes (MNB) Classifier, Linear Support Vector Machine (SVM), Logistic Regression (LR) and Decision Tree Classifier (C4.5). SVM is the fast and accurate method for classification of both linear and non-linear data. The method searches for the maximal marginal hyperplane that separates the two classes using support vectors. It is the most simplest and probabilistic model that computes the

posterior probability of a class based on a given feature set. It uses Bayes theorem to predict the probability that a given feature set belongs to the particular class [11].

Logistic Regression is simple, flexible and effective method used for classification. It finds the best fitting model to describe the relationship between the outcome and a set of independent variables. The classifiers takes features as an input and returns posterior probability of that instance belonging to a class [2].

Decision tree classifier C4.5 creates a decision tree based on the features that discriminate the classes the most. Tree is split at an attribute that maximizes information gain and minimizes entropy for a class. Thus, feature chosen as root node of the tree discriminates most between the classes. The leaf nodes represent the class labels in the tree. Decision tree uses Information gain as attribute selection measure as it minimizes the expected number of tests needed to classify given tuple into a class [27].

3.7. Proposed algorithm for spam classification

The pseudocode for proposed work is shown in Algorithm 1. The data transformed into a review-feature matrix is fed as input to the Algorithm 1. The top-k features selected by the algorithm are then used for training the classifiers where k is empirically determined number.

In Algorithm 1, Lines 1-11 compute the local feature selection score of each feature for both the classes using Odds ratio. If the local score is negative, the feature is tagged as negative feature, otherwise it is tagged as positive feature. If the absolute value of local score of feature for spam class is more than the non-spam class, the feature represents a spam class, else it represents a non-spam class.

In Line 12, Negative Feature Ratio (NFR) is computed using ratio of negative features in the data out of total features. Lines 13-15 compute global feature score using any one of the global feature rankers. On the basis of the global feature score, features and their respective tags and labels are sorted in descending order in Lines 16-18. In Line 21, k is empirically determined number and its value varies from k_1 to k_2 for experimental testing. In Line 22, the negative features NFR is selected such that it is always less than the value of negative feature ratio. The class ratio is determined as $k/2$ in Line 23 so that the final feature set of size k equally represent both the classes to overcome the problem of unbalanced feature set that occurs in global feature selection methods.

The construction of final feature set is done in Lines 24-29. To select final feature list, we iterate over the sorted feature list with their tag and class label till best k features are selected. If NFR is 0.1, it means that maximum 10% negative features can be selected while final feature selection process. The i-th feature from the globally sorted feature list is appended to the final feature list if it satisfies both NFRs and class ratio. In Lines 30-33, data is divided into 70% training and 30% testing data. The model is constructed using four different classifiers and AUC score is computed for performance analysis of the proposed approach.

The overall complexity of the Algorithm 1 is $O(nm^2+m+m\lg m)$ in the worst case where m is the total number of features, n is the number of review instances and c is the number of classes. In the algorithm, Odds ratio takes $2cnm$ for c=2 as there are

only two classes – spam or non-spam. The comparison of odds ratio and assignment of label and tag to a feature takes constant time. Since these steps are repeated for total m features, thus it takes $2cnm^2+m$ iterations to compute odds ratio for m features. The computation of negative feature ratio takes constant time. For each feature, global score is computed in $2nm^2$ iterations. The sorting of features using global feature score takes $O(m\lg m)$ for m features using best sorting algorithm. In the final step of feature selection, the sorted feature list is scanned and the features that satisfy negative feature ratio and class ratio are appended into a Fnal Feature List (FFL). This process repeats until the k-features are appended into the list.

Algorithm 1. Pseudocode for Improved global feature selection for spam classification

Input: n instances of reviews, set of extracted features fs, size of feature set |fs|=m, Review-Feature Matrix of size n x m, class label of reviews, Number of selected features k in the range (k1,k2)
Output: AUC score for top-k features
//Local feature ranker
1. for i ∈ fs do
2. Compute odds_ratio (i, C_j) using (1) //where j is 1 for spam class or 2 for non-spam class
// orC is class: spam (1) and non-spam (2) assigned to feature
// orL is tag: negative or positive assigned to feature
3. if abs(odds_ratio(i,C_1))>abs(odds_ratio(i,C_2)):
4. orC(i)=1
5. else:
6. orC(i)=2
7. if odds_ratio(i,C_1)<0:
8. orL(i)="negative"
9. else:
10. orL(i)="positive"
11. end for
// Compute negative feature score
12. $$\boldsymbol{nfr} = \frac{\boldsymbol{negative\_count(orL)}}{\boldsymbol{count(orL)}}$$
//Global feature ranker
13. for i ∈ fs do
14. Compute global feature score score (i) using (2) or (3) or (4);
15. end for
// Sort feature set fs, feature tag orL and feature class orC
according to descending order of global feature score
16. fs'=sort_score(fs)
17. orC'=sort_score(orC)

```
18. orL'=sort_score(orL)
19. ffs=[ ]//Initialize final feature set containing
selected
    features
20. count =0 //count variable stores the count of
selected features
21. Select the value of k in the range (k1, k2)
22. Select the value of nfrs in the range (0, nfr)
23. class ratio=k/2
// Final Feature Selection
24. for i ∈fs' do
25.  append feature i in ffs if orL'(i) and orC'(i)
satisfy nfrs
       and class ratio respectively
26.  count = count+1
27.  if count==k://Top-k features are determined
28.     break;
29. end for
// Data samples are divided into training and testing
samples
30. Train the classifiers using training instances and
final feature
    set ffs[1...k]
31. Predict the class of test instances
32. Compute the AUC score from the predicted results
33. Return AUC score for selected k features
```

In the worst case, m features can be scanned before selection. So it takes $O(m)$ for selecting top-k features. Ignoring the constant terms, the time complexity of algorithm is $O(nm^2+m+m\lg m)$. The results obtained by execution of the algorithm for different values of k and NFRs is shown in the Section 4.

4. Experimental results

In this section, an in-depth investigation is carried out on two different datasets of hotel domains to analyze the performance of improved global feature selector. The different global feature selectors employed in the experiments are GI, IG and DFS and local feature selector used is Odds ratio. Scikit-learn in Python is used for feature extraction, classification algorithms and performance metrics. In the following subsections, performance metrics used for evaluation and results obtained on this metric are discussed.

4.1. Performance metrics

To evaluate the performance of improved feature selector on selected classifiers, Area under the receiver operator curve metrics is chosen as it shows model performance

across all decision thresholds. The metrics is a graph of false positive rate versus true positive rate. The larger the value of AUC, the better the classifier performance.

4.2. Evaluation

In this section, performance analysis of IGFS over global feature selector on the two datasets using different classification methods discussed have been shown. The AUC score achieved by classifiers without feature selection, global feature selectors and improved global feature selection have been analyzed using different values of k in this section. The results are shown on computed negative feature ratio (NFRs) while using IGFS. The bold values in the Tables 1-3 show the best AUC score for the selected number of features and classifier used. In all cases, 70% data is used for training and 30% data for testing.

Table 1. AUC score on Dataset 1 and Dataset 2 without feature selection

Dataset/Classifier	SVM	MNB	LR	C4.5
Dataset 1: TripAdvisor Hotel Review dataset	0.8496	**0.872254**	0.845593	0.684802
Dataset 2: Yelp Filtered review dataset	0.650491	0.573422	**0.664019**	0.551756

Table 1 shows the AUC score achieved by classifiers on both the datasets without feature selection. The total number of unigram and bigram extracted are 90735 and 117428 for Dataset 1 and Dataset 2, respectively. MNB classifier performs best on dataset 1 achieving an AUC score of 0.8722 and Logistic Regression performs best on dataset 2 achieving AUC score of 0.6640.

Table 2a. AUC score on Dataset 1 using GI and IGFS for NFRs=0.1

Classifier	Number of selected features																
	2000	2500	3000	3500	4000	4500	5000	5500	6000	6500	7000	7500	8000	8500	9000	9500	10,000
SVM	0.83	0.83	**0.84**	0.84	0.84	0.84	0.83	0.84	0.83	0.83	0.83	0.84	0.84	0.84	0.84	0.84	0.84
	0.84	0.83	0.83	0.83	0.83	0.84	0.85	0.84	**0.85**	0.84	0.84	0.85	0.85	0.85	0.84	0.84	0.83
MNB	0.84	0.86	0.91	0.90	0.93	0.95	0.95	0.93	0.92	0.93	0.94	0.95	0.96	0.96	**0.97**	0.97	0.97
	0.87	0.87	0.87	0.88	0.91	0.92	0.94	0.95	0.96	0.97	**0.98**	0.97	0.95	0.94	0.94	0.94	0.95
LR	**0.85**	0.84	0.84	0.85	0.85	0.85	0.84	0.84	0.84	0.84	0.84	0.84	0.84	0.84	0.84	0.85	0.84
	0.85	0.85	0.84	0.84	0.85	0.84	0.84	0.84	0.84	0.84	0.84	0.84	0.84	0.84	0.84	0.84	0.84
C4.5	**0.73**	0.70	0.71	0.73	0.70	0.69	0.70	0.71	0.70	0.69	0.71	0.69	0.69	0.71	0.73	0.71	0.72
	0.72	0.71	0.72	0.71	0.74	0.71	0.70	0.69	0.71	0.71	0.70	0.70	0.71	0.74	0.72	0.71	0.72

Table 2b. AUC score on Dataset 1 using IG and IGFS for NFRs=0.1

Classifier	Number of selected features																
	2000	2500	3000	3500	4000	4500	5000	5500	6000	6500	7000	7500	8000	8500	9000	9500	10,000
SVM	0.55	0.55	0.54	0.59	0.57	0.63	0.65	0.65	0.64	0.65	0.65	0.65	0.67	0.67	0.66	0.67	**0.69**
	0.52	0.54	0.54	0.56	0.59	0.61	0.62	0.65	0.65	0.64	0.66	0.66	0.66	0.67	0.67	0.66	**0.67**
MNB	0.61	0.62	0.62	0.64	0.64	0.65	0.68	0.69	0.68	0.68	0.68	0.68	0.69	0.68	0.70	0.69	**0.71**
	0.60	0.58	0.60	0.60	0.65	0.64	0.66	0.69	0.70	0.70	0.69	0.68	0.68	0.70	0.71	0.71	**0.72**
LR	0.55	0.56	0.57	0.59	0.62	0.65	0.66	0.66	0.65	0.69	0.67	0.68	0.68	0.69	0.68	0.69	**0.70**
	0.51	0.55	0.56	0.58	0.65	0.65	0.64	0.68	0.68	0.65	0.65	0.67	0.68	0.69	0.69	0.70	**0.69**
C4.5	0.54	0.52	0.51	0.58	0.57	0.58	0.59	0.63	0.60	0.59	0.61	0.59	0.62	0.63	0.62	0.62	**0.63**
	0.53	0.54	0.52	0.52	0.53	0.57	0.57	0.59	0.57	0.58	0.60	0.60	0.60	0.61	0.60	0.62	**0.63**

Table 2c. AUC score on Dataset 1 using DFS and IGFS for NFRs=0.1

Classifier	Number of selected features																
	2000	2500	3000	3500	4000	4500	5000	5500	6000	6500	7000	7500	8000	8500	9000	9500	10,000
SVM	0.88	0.85	0.88	0.89	0.89	0.90	0.90	**0.91**	0.91	0.91	0.90	0.90	0.90	0.90	0.90	0.89	0.90
	0.87	0.89	0.89	0.89	0.89	0.89	0.90	**0.92**	0.91	0.91	0.92	0.89	0.90	0.89	0.90	0.89	0.90
MNB	0.94	0.94	0.95	0.95	0.95	0.96	0.96	0.96	0.96	0.96	**0.97**	0.96	0.96	0.96	0.96	0.96	0.96
	0.94	0.95	0.95	0.95	0.95	0.96	0.96	0.96	0.96	**0.97**	0.97	0.96	0.96	0.96	0.96	0.96	0.96
LR	0.90	0.90	0.91	0.92	0.93	0.93	0.93	0.94	**0.94**	0.93	0.93	0.93	0.93	0.93	0.93	0.92	0.93
	0.90	0.91	0.92	0.92	0.93	0.93	0.94	0.94	**0.94**	0.94	0.94	0.93	0.93	0.93	0.93	0.93	0.93
C4.5	0.70	0.72	0.71	0.72	0.71	**0.72**	0.70	0.70	0.71	0.69	0.68	0.66	0.67	0.67	0.67	0.68	0.67
	0.72	0.72	**0.72**	0.70	0.70	0.72	0.70	0.71	0.69	0.69	0.70	0.70	0.67	0.67	0.65	0.68	0.69

Table 2 shows comparison on AUC score achieved by chosen classifiers using IGFS and global selection method on dataset 1. The number of features selected for analysis varies from 2000 up to 10,000 by increment of 500. The results in Table 2a show that all classifiers using IGFS for feature selection outperforms GI in terms of AUC score by 1% except Logistic Regression. The best value of AUC score is achieved by MNB classifier when NFRs=0.1 and number of selected features using IGFS is 7000. Table 2b depicts an improvement of 1% AUC score by MNB classifier and IGFS using Information gain and odds ratio. As shown in Table 2c, only SVM shows improvement in AUC score on using ensemble of distinguishing global feature selector and odds ratio for feature selection.

Table 3a. AUC score on Dataset 2 using GI and IGFS for NFRs=0.08

Classifier	Number of selected features											
	5000	6000	7000	8000	9000	10,000	11,000	12,000	13,000	14,000	15,000	16,000
SVM	0.681	0.669	0.683	0.687	**0.689**	0.687	0.685	0.670	0.666	0.671	0.665	0.665
	0.679	0.679	0.681	0.691	**0.694**	0.687	0.690	0.668	0.671	0.668	0.668	0.673
MNB	0.817	0.821	0.864	0.879	0.903	**0.906**	0.879	0.841	0.827	0.798	0.786	0.786
	0.831	0.860	0.837	0.875	0.898	**0.910**	0.879	0.837	0.827	0.827	0.827	0.827
LR	0.687	0.692	0.696	0.692	**0.696**	0.687	0.683	0.685	0.686	0.688	0.686	0.684
	0.696	**0.702**	0.696	0.700	0.702	0.696	0.694	0.690	0.690	0.692	0.692	0.690
C4.5	**0.603**	0.593	0.591	0.591	0.589	0.581	0.572	0.578	0.584	0.589	0.589	0.576
	0.588	0.593	0.576	0.599	0.591	0.574	0.576	0.594	**0.622**	0.560	0.566	0.598

Table 3b. AUC score on Dataset 2 using IG and IGFS for NFRs=0.08

Classifier	Number of selected features											
	5000	6000	7000	8000	9000	10,000	11,000	12,000	13,000	14,000	15,000	16,000
SVM	0.577	0.604	0.612	0.606	0.610	0.629	0.640	**0.636**	0.631	0.619	0.619	0.602
	0.579	0.593	0.610	0.606	0.608	0.631	0.619	0.612	**0.640**	0.598	0.606	0.602
NB	0.552	0.552	0.562	0.586	0.598	0.601	0.615	0.586	0.592	0.605	0.624	**0.632**
	0.560	0.560	0.567	0.579	0.588	0.592	0.603	0.592	0.598	0.611	0.619	**0.619**
LR	0.593	0.602	0.608	0.636	0.642	0.625	**0.644**	0.644	0.623	0.614	0.612	0.608
	0.595	0.593	0.608	**0.644**	0.642	0.637	0.640	0.625	0.633	0.614	0.616	0.608
DTree	0.581	0.596	0.570	0.558	0.564	0.587	0.558	**0.604**	0.574	0.606	0.557	0.581
	0.579	0.579	0.583	0.583	0.554	0.540	0.583	0.587	**0.613**	0.555	0.573	0.555

Table 3c. AUC score on Dataset 2 using DFS and IGFS for NFRs=0.08

Classifier	Number of selected features											
	5000	6000	7000	8000	9000	10,000	11,000	12,000	13,000	14,000	15,000	16,000
SVM	**0.802**	0.747	0.752	0.756	0.758	0.763	0.765	0.774	0.768	0.763	0.766	0.770
	0.781	0.779	0.783	0.752	0.743	0.758	0.760	0.750	0.752	0.750	0.756	0.754
NB	**0.850**	0.839	0.843	0.847	0.843	0.844	0.833	0.835	0.806	0.810	0.790	0.771
	0.856	0.862	**0.868**	0.849	0.847	0.839	0.843	0.843	0.843	0.839	0.837	0.837
LR	**0.811**	0.778	0.788	0.790	0.788	0.786	0.788	0.789	0.789	0.787	0.787	0.793
	0.807	**0.813**	0.809	0.786	0.790	0.788	0.790	0.788	0.786	0.786	0.784	0.788
DTree	**0.645**	0.640	0.659	0.623	0.630	0.655	0.652	0.621	0.636	0.646	0.536	0.536
	0.652	**0.658**	0.652	0.650	0.634	0.644	0.648	0.644	0.642	0.634	0.642	0.642

Table 3 shows comparison on AUC score achieved by chosen classifiers using IGFS and global selection method on dataset 2. The performance is analysed on subset of selected features varying from 5000 to 16,000 by increment of 1000. The results in Table 3a show that all classifiers using IGFS for feature selection outperforms GI in terms of AUC score by around 1%. As shown in Table 3b IGFS using ensemble of IG and Odds ratio benefits only SVM and Decision Tree classifier in terms of AUC score. Table 3c depicts the classifier performance when ensemble of DFS and Odds ratio for improved feature selection method is used. The results show that only Decision Tree and MNB classifier perform better on using this ensemble.

Table 4 shows the comparison of proposed approach with existing approaches that used feature selection method for spam classification. In the experimental study conducted by Crawford, Khoshgoftaar and Prusa [4] for spam classification, ten different global feature selectors were used with five classifiers – C4.5, MNB, NB, SVM and LR on 2836 reviews of hotel, restaurant and doctors domain. There were 1200 truthful reviews collected from actual review sites and 1636 fake reviews generated from AMT and experts in the dataset used for their investigation. The best AUC was reported using Chi-Square and Signal-to-Noise ratio as feature selector and MNB as classifier. The other study by [7] on same dataset was conducted to determine the effectiveness of combination of ensemble technique and feature selection methods. The results achieved in their work showed that the combination of Select-Boost, MNB and Chi-Square or Signal-to-Noise ratio as feature selector outperforms all other classifiers except Random Forest classifier with tree size 500.

Table 4. Comparison of proposed work with existing approaches [4, 7]

Method	Crawford, Khoshgoftaar and Prusa [4]	Crawford et al. [7]	Proposed approach	
Dataset used	1200 Truthful + 1636 fake reviews on three domains-restaurants, hotels and doctors	1200 Truthful + 1636 fake reviews on three domains-restaurants, hotels and doctors	TripAdvisor Hotel Review dataset	Yelp Filtered review dataset
Feature selection method	Chi-Square or Signal –to-Noise ratio	Chi-Square or Signal to Noise ratio	Gini index + Odds ratio	Gini index + Odds ratio
Classifier used	Multinomial Naïve Bayes	Select-Boost and MNB classifier	Multinomial Naïve Bayes	Multinomial Naïve Bayes
AUC score	0.89	0.91	**0.98**	**0.91**

In our proposed work, an ensemble of local and global filter-based feature selection method is used for spam classification. To prove the effectiveness of this method on the classifiers, testing was done on both real and synthetic dataset of hotel reviews. The results showed that combination of Gini index or distinguishing global feature selection methods with Odds ratio improve the performance of the classifiers. MNB classifier using ensemble of Odds ratio with Gini index provide the best AUC score of 0.98 and 0.91 on synthetic and real dataset respectively.

5. Conclusion

The research [26] found that text classification could benefit by integrating local feature selection method with global feature selection method. In our work, we tested the performance of various classifiers using improved global feature selection method for content-based spam detection. The experiments were conducted on different combination of global feature rankers with odds ratio. The results showed that the integration of local feature selectors with global feature selectors lead to selection of better feature set thus increasing the classifier performance.

Future work may involve testing with some other combination of local and global feature selection metrics to further improve the results. Instead of hotel reviews, proposed feature selection method can also be used on other domain for spam detection. Apart from text-based features, spam detection also benefits from other meta-features associated with reviews. So further work can be extended to include meta-features along with textual features for improving the performance of spam detection on online reviews.

References

1. Banerjee, S., A. Y. K. Chua. Applauses in Hotel Reviews: Genuine or Deceptive? – In: Science and Information Conference (SAI), London, IEEE, 2014.
2. Bishop, C. M. Pattern Recognition and Machine Learning. Springer, 2006.
3. Crawford, M., H. Al Najada. Survey of Review Spam Detection Using Machine Learning Techniques. – Springer, Journal of Big Data, Vol. **2**, 2015, No 1, p. 23.
4. Crawford, M., T. M. Khoshgoftaar, J. D. Prusa. Reducing Feature Set Explosion to Facilitate Real-World Review Spam Detection. – In: Proc. of 29th International Florida Artificial Intelligence Research Society Conference, AAAI, 2016, pp. 304-309.
5. Fei, G., A. Mukherjee, B. Lui, M. Hsu, M. Castellenos, R. Ghosh. Exploiting Burstiness in Reviews for Review Spammer Detection. – In: Proc. of 7th International Conference on Weblogs and Social Media, AAAI, 2013. pp.175-184.
6. Forman, G. An Extensive Empirical Study of Feature Selection Metrics for Text Classification. – Journal of Machine Learning Research, Vol. **3**, 2003, pp. 1289-1305.
7. Heredia, B., T. M. Khoshgoftaar, J. D. Prusa, M. Crawford. Improving Detection of Untrustworthy Online Reviews Using Ensemble Learners Combined with Feature Selection. – Springer, Social Network Analysis and Mining, Vol. **7**, 2017, No 1, pp. 1-37.
8. Heydari, et al. Detection of Review Spam: A Survey. – Expert Systems with Applications, Vol. **42**, 2014, No 7, pp. 3634-3642.
9. Hu, N., I. Bose, S. K. Koh, L. Liu. Manipulation of Online Reviews: An Analysis of Ratings, Readability and Sentiments. – Elsevier, Decision Support Systems, Vol. **52**, 2011. pp. 674-684.
10. Jindal, N., B. Liu. Opinion Spam and Analysis. – In: Proc. of 2008 International Conference on Web Search and Data Mining, ACM, 2008, pp. 219-230.

11. K a m b e r, et al. Data Mining: Concepts and Techniques. Second Edition. Elsevier, 2008.
12. L i, F., M. H u a n g, Y. Y a n g, X. Z h u. Learning to Identify Review Spam. – In: Proc. of 22nd International Joint Conference in Artificial Intelligence, 2011, pp. 2488-2493.
13. L i, H., Z. C h e n, A. M u k h e r j e e, B. L i u, J. S h a o. Analyzing and Detecting Opinion Spam on Large Scale Dataset via Temporal and Spatial Patterns. – In: Proc. of 9th International Conference on Web and Social Media, AAAI, 2015, pp. 634-637.
14. L i, H., B. L i u, A. M u k h e r j e e, J. S h a o. Spotting Fake Reviews Using Positive Unlabeled Learning. – Computacion y Sistemas, Vol. **18**, 2014, No 3, pp. 467-475.
15. L i, S., R. X i a, C. Z o n g, C. H u a n g. A Framework of Feature Selection Methods for Text Categorization. – In: Proc. of 47th Annual Meeting of the ACL and the 4th IJCNLP of the AFNLP, 2009, pp. 692-700.
16. L i m, E. P., J. N. N g u y e n, B. L i u, H. W. L a u w. Detecting Product Review Spammers Using Rating Behaviors. – In: Proc. of 19th ACM International Conference on Information and Knowledge Management, ACM, 2010, pp. 939-948.
17. L o n g, N. H., P. H. T. N g h i a, N. M. V u o n g. Opinion Spam Recognition Method for Online Reviews Using Ontological Features. – Tap Chi KHOA HOC DHSP TPHCM, Vol. **61**, 2014, pp. 44-59.
18. M u k h e r j e e, A., B. L u i, N. G l a n c e. Spotting Fake Review Groups in Consumer Reviews. – In: Proc. of 21st International Conference on World Wide Web, ACM, 2012, pp. 191-200.
19. M u k h e r j e e, A., et al. Spotting Opinion Spammers Using Behavioral Footprints. – In: Proc. of 19th International Conference on Knowledge Discovery and Data Mining, ACM, 2013, pp. 632-640.
20. N a j a d a, H. A l, X. Z h u. iSRD: Spam Review Detection with Imbalanced Data Distributions. – In: Proc. of 15th International Conference on Information Reuse and Integration (IRI), IEEE, 2014, pp.553-560.
21. O t t, et al. Finding Deceptive Opinion Spam by Any Stretch of Imagination. – In: 49th Annual Meeting of the Association for the Computational Linguistics, Portland, Oregon, 2011, pp 309-319.
22. O t t, et al. Estimating the Prevalence of Deception in Online Review Communities. – In: Proc. of 21st International Conference on World Wide Web, ACM, 2012.
23. R a s t o g i, A., M. M e h r o t r a. Opinion Spam Detection in Online Reviews. – Journal of Information & Knowledge Management, Vol. **16**, 2017, No 4, World Scientific Press, pp. 1750036 (38 pages).
24. R a y a n a, S., L. A k o g l u. Collective Opinion Spam Detection: Bridging Review Networks and Metadata. – In: Proc. of 21th ACM Sigkdd International Conference on Knowledge Discovery and Data Mining, ACM, 2015, pp. 985-994.
25. S a v a g e, D., X. Z h a n g, X. Y u, P. C h o u, Q. W a n g. Detection of Opinion Spam Based on Anomalous Rating Deviation. – Expert Systems with Applications, Vol. **42**, Elsevier, 2015, pp. 8650-8657.
26. U y s a l, A. K. An Improved Global Feature Selection Scheme for Text Classification. – Elsevier, Expert Systems with Applications, Vol. **43**, 2016, pp. 82-92.
27. U y s a l, A. K., S. G u n a l. A Novel Probabilistic Feature Selection Method for Text Classification. – Knowledge-Based Systems, Vol. **36**, 2012, pp. 226-235.
28. W a n g, G., S. X i e, B. L u i, P. S. Y u. Review Graph Based Online Store Review Spammer Detection. – In: Proc. of 11th IEEE International Conference on Data Mining (ICDM'11), 2011, pp. 1242-1247.
29. W a n g, G., S. X i e, B. L u i, P. S. Y u. Identify Online Store Review Spammers via Social Review Graph. – ACM Transactions on Intelligent Systems and Technology, Vol. **3**, 2012, No 4, pp. 1-21.

Received 15.05.2018; Second Version 16.11.2018; Accepted 21.11.2018