Semantic-Based Feature Reduction Approach for E-mail Classification

Eman M. Bahgat(✉) and Ibrahim F. Moawad

Faculty of Computer and Information Sciences,
Ain Shams University, Cairo, Egypt
{eman.bahgat4, ibrahim_moawad}@cis.asu.edu.eg

Abstract. E-mail is one of the most important applications for all the computer users due to its efficiency and low cost. However, some users use it in sending spam emails, which become a severe problem that has great effect on the users' performance. E-mail filtering is an important approach to identify those spam emails. In this paper, based on different machine learning algorithms, a novel semantic-based approach for email filtering is proposed. The approach analyses the content of the email and assigns a weight to each term that can help in classifying it into spam or ham email. We enhanced the traditional Email filtering approaches by applying semantic-based feature reduction model using the WordNet ontology in order to handle the high dimensionality problem of feature size. The experiments that have been conducted using Enron dataset showed great results. A comparative study has also been presented among different classifiers that prove the efficiency of the proposed approach. These classifiers are Naïve Bayes (NB), Support Vector Machine (SVM), Logistic Regression, J48 and Random Forest. The Logistic Regression classifier has the best accuracy with value of 0.96. Followed by the NB and SVM that almost have similar results of accuracy value 0.93. Finally, the Random Forest and J48 classifiers have the least accuracy values of 0.85 and 0.87 respectively.

Keywords: Email filtering · Wordnet ontology · Email classification · Spam email · Feature reduction

1 Introduction

E-mail service is one of the most important ways of communication in our life. Emails have many advantages as they are easy to use, are low cost, and can contain attached files. On the other hand, some users used email in sending computer worms and spam emails.

Spam E-mails, also known as junk emails or Unsolicited Bulk Emails (UBE), are unwanted or undesired E-mails that are sent to a group of users. According to a Cyberoam report [1], the average number of spam messages sent every day has reached 54 billion messages. The sender of spam email does not target the recipient personally, but the spam emails invade users without their assent and fill out their email inbox. Besides the time consumed in checking and deleting spam emails, they overload the network bandwidth by useless data packages. Therefore, many problems may arise:

A.E. Hassanien et al. (eds.), *Proceedings of the International Conference on Advanced Intelligent Systems and Informatics 2016*, Advances in Intelligent Systems and Computing 533, DOI 10.1007/978-3-319-48308-5_6

increasing the operating cost, impacting the work productivity and privacy, and harming the network infrastructure and the recipient's machine.

To solve this problem, an Email Filtering approach is required to identify the spam emails and to dispose huge number of spam emails efficiently. The filtering approaches can be classified into two groups: based on the email origin (i.e. email source) or based on the email content (email body).

Origin-based filtering monitors the e-mail source, which is stored in the address of the sender device and the domain name. This type of filtering preserves two types of email sources; white-list and black-list. Usually, the new email source is compared with a history database to know how it is classified [2]. The problem of using Origin-based filtering is that spammers regularly change the email address, source, and IP. On the other hand, content-based filtering approaches review the email content depending on a proposed analysis technique [2]. However, the content-based filtering generates a large number of features as well as ambiguity of the meaning of the email terms. The traditional Email filtering approaches classified the emails based on the occurrence of the term in the email and neglect the syntactic and semantic properties of the email text.

In this paper, a semantic based feature reduction approach is proposed for filtering emails using the WordNet ontology [3] to handle the problem of high dimensionality of email features. We used the synonyms set of each term and group the terms that have common synonyms. Moreover, we consider the determined (meaningful) words only using the WordNet ontology as an English dictionary. In addition, we assign a weight for each term according to their occurrences in the emails. Different machine learning algorithms are applied in our approach: Naïve Bayes, Support Vector Machine, Logistic Regression, J48 and Random Forest. The results prove the efficiency of the proposed approach. The Logistic Regression recorded the best accuracy with value of 0.96. This is followed by the NB and SVM with accuracy value 0.93 (having similar results). Finally, the Random Forest and J48 classifiers have the least accuracy values of 0.85 and 0.87 respectively.

The rest of the paper is organized as follows. Section 2 outlines the related work while Sect. 3 presents the proposed system architecture of our approach. Section 4 explains how we apply semantics and how we weight the email features. Section 5 gives a brief review about the five machine learning algorithms used and Sect. 6 discusses the experiments and the results. Finally, Sect. 7 includes the conclusion and future work.

2 Related Work

As we discussed above, email filtering can be executed based on origin or content of the email. Some researchers have used origin-based filtering. For example, in [4], an email classification model has been proposed based on four machine learning algorithms: Naïve Bayes, term frequency/inverse document frequency, K-nearest neighbor, and support vector machines. The experiments have been conducted on the header part only of the email.

Most of the literature applied the email filtering based on several classification methods. Some of these classification methods are Naive Bayes (NB) [5–7], logistic regression [7], k-Nearest Neighbor (KNN) [6], C4.5 Classifier [8, 9], Artificial Neural Net-works (ANN) [8], AdaBoost [10, 11], Random Forest (RF) [11], Support Vector Machine (SVM) [5, 6, 11], and Multi-Layer Perceptron (MLP) [8]. These methods have been applied based on the content of the email.

Concerning content-based approaches, in [9], an ensemble learning and decision tree based model has been used to detect the spam emails. Four classification algorithms were applied: C4.5, NB, SVM, and KNN. Also, in [12], another filtering approach has been proposed based on two behavioral features (the URL and the time the email was dispatched) and eight keywords known as bag of spam words to classify the spam and legitimate emails. The machine learning algorithm used was RF. However, both the authors of [9, 12] didn't apply any feature reduction method.

One of the challenges for email filtering is the high dimensionality of data or feature space used. The authors in [13] have proposed a spam detection approach based on Random Forests (RF) classifier, which enables parameters optimization and feature selection approach. In [14], statistical feature selection approach based on similarity coefficients is proposed to enhance the accuracy of spam filtering and detection rate. In [15], an improvement in mutual information algorithms combined with word frequency and average word frequency to measure the relation between a feature and a class. Other feature selection methods were also presented in [16].

Some of authors conduct comparative analysis for different machine learning algorithms, in order to compare the performance different classifiers, such as in [17–19]. In [17], a spam classification approach has been proposed that uses features based on the content of the email and some readability features related to the email (e.g. document length and word length). The work has been applied using five classification algorithms: Bagging, RF, SVM, AdaBoost, and Naïve Bayes. Authors in [18] introduced other comparative analysis. Four algorithms have been used for email filtering: Logistic Regression, Neural Network, NB, and RF. In [19], they presented a comparative analysis based on four classifiers; J48, SVM, BayesNet, and LazyIBK.

A classification-based email filtering approach has been presented in [20]. This approach tries to reduce the email content features by applying stemming on the content. After that, an English dictionary was exploited in order to regard the meaningful terms only. Five classification algorithms have been used in the experiments: Naïve Bayes, SVM, Logistic Regression, J48, and Random Forest.

On the other hand, there is another way to filter emails semantically to tackle the ambiguity problems. A semantic email categorization approach based on semantic vector space model has been proposed in [21] using WordNet. The experiments have been applied using SVM, Logistic Regression, and KNN. In [22], they presented a semantic feature based model for identifying emails using general ontology to overcome the terms mismatch problem. The work has been tested using SVM classifier. Other models based on semantic text classification have been introduced in [23].

3 Proposed System Architecture

Figure 1 shows the proposed system architecture that semantically reduces the email feature dimensions to classify the emails into Ham and Spam classes. It consists of four main modules: Email Pre-processing, Semantic Based Feature Reduction, Feature Weighting, and Classification modules.

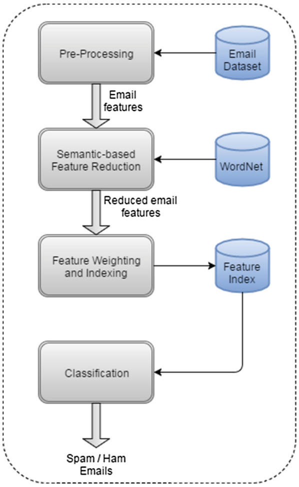


Fig. 1. The Proposed System Architecture.

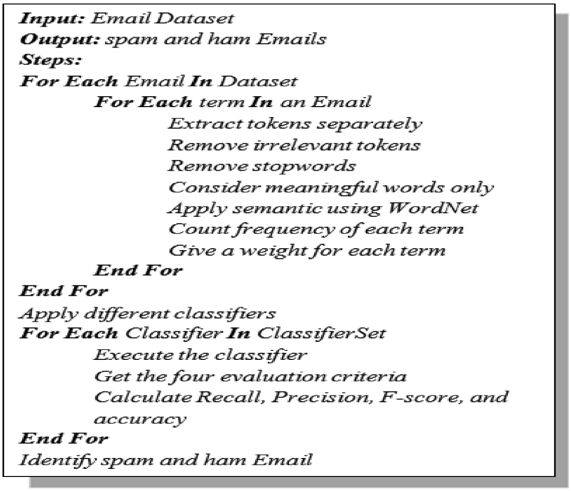
In the pre-processing module, the main purpose is to remove the irrelevant terms and reduce the number of extracted features for the classifier. We extract the tokens of both subject line and the content body of the email. Then the irrelevant tokens are removed such as symbols and numbers. This is followed by eliminating the stop words. We do not apply stemming in order to keep the meaning of the words.

After the pre-processing step, the features of each email are processed semantically to reduce the size of the complete email features. We used the WordNet ontology by considering the synonymy relation among terms. This is followed by assigning a weight for each term and building the feature index.

Finally, different machine learning algorithms are applied to classify the emails into spam and ham emails. For more clearness, Fig. 2 shows the proposed system algorithm and how our semantic-based email filtering approach works.

4 Feature Reduction and Weighting

In this step, a semantic technique is used to reduce the size of extracted features in the email using the WordNet, which is a lexical database for English language. WordNet is like a supercharged dictionary/thesaurus with a graph structure. It contains a collection



```
Input: Email Dataset
Output: spam and ham Emails
Steps:
For Each Email In Dataset
    For Each term In an Email
        Extract tokens separately
        Remove irrelevant tokens
        Remove stopwords
        Consider meaningful words only
        Apply semantic using WordNet
        Count frequency of each term
        Give a weight for each term
    End For
End For
Apply different classifiers
For Each Classifier In ClassifierSet
    Execute the classifier
    Get the four evaluation criteria
    Calculate Recall, Precision, F-score, and
    accuracy
End For
Identify spam and ham Email
```

Fig. 2. Semantic-based Email Filtering Algorithm.

of English words (nouns, adjectives, verbs and adverbs) that are linked together by their semantic relations. English words are grouped into sets of synonyms called synsets. A synset corresponds to an abstract concept.

The purpose of using WordNet is to reduce the high dimensionality of feature using the synonyms set of each feature. We group the terms that have the same synonyms together, and scale up their weight in each email. WordNet comprises large number of English words. Hereby, it is used as an English dictionary in order to identify the meaningful words in our work, and hence the undetermined words will be discarded.

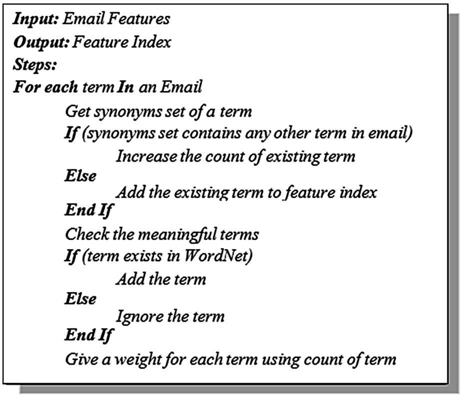
Figure 3 shows the feature reduction and weighting algorithm. After extracting the email features from pre-processing module, we used WordNet ontology as semantic resource. We tried to reduce the high dimensionality of features by considering the synonyms set of each term in an email. If one synonym in synonym set is mutual with other terms in the same email, then we can scale up the count and ignore that term. If no mutual terms, then add the new term to the feature index. In addition, we used WordNet ontology as an English dictionary to get the meaningful terms only.

After extracting the meaningful terms, each term is assigned a weigh, which is the number of occurrences of this term and its synonyms in each email. We use the term frequency/inverse document frequency Eq. (3), which is used for indexing the documents in information retrieval.

Term Frequency (TF): is defined as the number of times that term (t) occurs in document (d).

$$\mathrm{TF}(\mathrm{t},\mathrm{d}) = \frac{\mathrm{f_d}(\mathrm{t})}{\max[\mathrm{f_d}(\mathrm{t})]} \tag{1}$$

Where $\mathrm{f_d}(\mathrm{t})$ is the frequency of term (t) in document (d).



```
Input: Email Features
Output: Feature Index
Steps:
For each term In an Email
        Get synonyms set of a term
        If (synonyms set contains any other term in email)
                Increase the count of existing term
        Else
                Add the existing term to feature index
        End If
        Check the meaningful terms
        If (term exists in WordNet)
                Add the term
        Else
                Ignore the term
        End If
        Give a weight for each term using count of term
```

Fig. 3. Feature Reduction and Weighting Algorithm.

Inverse Document Frequency (IDF): estimates the rarity of a given term in the whole document collection (If a term occurs in all documents of the collection, its IDF is zero.) and measures how important a term is within a particular document, by computing it using Eq. (2):

$$\mathrm{IDF(t)} = \log\left(\frac{\mathrm{N}}{\mathrm{df_t}}\right) \tag{2}$$

Where N is the total no. of documents and df_t is the no. of documents with term (t).

Finally, The TF-IDF is the product of its TF weight in Eq. (1) and its IDF weight in Eq. (2).

$$\mathrm{TF-IDF} = \mathrm{tf(t, d)} \times \mathrm{IDF(t)} \tag{3}$$

Our experiments have been conducted using TF-IDF.

5 Classifiers

In this step, we tried different classifiers to evaluate the proposed model, which are Naïve Bayes, SVM, Logistic Regression, J48 and Random Forest.

5.1 Naïve Bayes

The Naive Bayes algorithm is a simple probabilistic classifier that measures a set of probabilities by calculating the number of combinations and frequency of terms in a certain dataset. The Bayesian classifier makes a conditional independence assumption between the attributes and that significantly reduce the number of attributes, therefore it tends to proceed well and learn rapidly in different supervised classification problems [6, 7].

5.2 Support Vector Machine

Support vector machine (SVM) is a classification algorithm with strong theoretical base. SVM is a group of related supervised learning methods used in classification and regression. It separates the data into two classes by constructing a straight line (1 dimension), flat plane (2 dimensions) or an N-dimensional hyperplane. SVM can handle high dimensional feature space effectively [5, 11].

5.3 J48 Classifier

J48 is one of the most popular decision tree algorithms. J48-classifier J48 builds decision trees from a group of training data. According to the splitting node strategy, j48 selects one attribute from the training data that effectively splits its set of instances into smaller subsets. J48 classifier then visits each decision node recursively and chooses the most effective split until each leaf is pure and no more splits are available, meaning that the data has been classified as perfectly as possible [8, 9].

5.4 Logistic Regression

Logistic regression can handle the relationship between a dependent nominal variable and one or more independent variables [7]. It collects the independent variables to assess the probability that a particular event will occur.

Logistic Regression can be used when the target variable is a categorical variable with two categories – for example spam/ham emails. For a given case, logistic regression measure the probability that a case with a certain set of values of the independent variable is a member of the modeled category. If the probability is greater than 0.5, the case is classified in the modeled category. If the probability is less than 0.50, the case is classified in the other category.

5.5 Random Forest

Random Forest (RF) is an ensemble classifier that consists of a collection of decision trees. A random selected subset of training data features is used to split each tree independently and with the same distribution for all trees in the forest. A randomized selection of variables is used to divide the nodes. The main idea for using ensemble methods is that a group of weak classifiers can come together to form a strong classifier. RF runs efficiently on large datasets and its learning is fast [11].

6 Experimental Evaluation and Discussion

In our approach, we exploit the Enron-Spam, which is a large public email database collection. It contains data from about 150 employees. The corpus contains a total of about 0.5 M messages. It focuses on six Enron employees with large mail-boxes.

The Enron dataset is divided into six different subsets [24]. Our Experiments are done on a subset of the Enron Corpus containing 300 emails (32 % spam, 68 % ham). The experiments were executed on a machine with hardware specification of processor: Intel® core i7 and main memory of 8 GB.

The data set was separated randomly into two parts, the first part is used as training data set to produce the prediction model, and the other part is used for testing data set to evaluate the accuracy of our model. Testing is done by using 10-fold cross validation method. In our experimentations, we have used WEKA tool [25] to apply different machine learning algorithms. WEKA tool is an open source software that provides a collection of machine learning algorithms for data mining tasks.

6.1 Evaluation Measures

The performance of the classifiers is measured using recall, precision and accuracy.

Recall represents the percentage of correctly identified positive cases and defined as in Eq. (4):

$$\mathrm{Recall} = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FN}} \tag{4}$$

Precision reflects the number of real predicted examples and defined as:

$$\mathrm{Precision} = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FP}} \tag{5}$$

The overall accuracy has been also defined by:

$$\mathrm{Accuracy} = \frac{\mathrm{TP} + \mathrm{TN}}{\mathrm{TP} + \mathrm{TN} + \mathrm{FP} + \mathrm{FN}} \tag{6}$$

Where TP is the true positive instances, TN is the true negative instances, FP is the false positive instances and FN is the false negative instances.

6.2 Results and Discussion

As explained above, the Enron dataset has been preprocessed. After applying the proposed approach, comparing to the related work in [20], the number of features was reduced from 3636 to 2309 with reduction rate equals 36.5 %. Therefore, the execution time of the classifiers was decreased due to the feature reduction.

The experiments have been conducted on our extracted feature set as mentioned above using the TF-IDF (Eq. 3) for email weighting. The classifier performance is measured using the precision, recall and accuracy.

Table 1 shows the performance of different classifiers using our proposed approach. Logistic Regression recorded the best precision with value of 0.96. Followed by the Naïve Bayes and SVM that recording almost similar results of precision value 0.93.

Table 1. Performance results for the proposed work

Evaluation Criteria	Classifiers				
	Naïve Bayesian	SVM	Logistic Regression	J48	Random Forest
Precision	0.927	0.936	0.96	0.857	0.872
Recall	0.927	0.937	0.96	0.857	0.87
Execution Time (Sec)	0.06	0.24	0.13	0.32	0.06

The J48 classifiers recorded the least precision value of 0.857. The Recall almost recorded the same performance like Precision.

In addition, the proposed work has been compared to the related work in [18, 20]. In Fig. 4, we compare the common classifiers of our proposed work with these related works, which also used the Enron dataset. As shown the proposed work has a higher accuracy than the related work in [18] with respect to Naïve Bayes, Logistic Regression, and J48 classifiers. However, for Random Forest it is slightly lower than the related work in [18]. Figure 4 also shows that the accuracy of J48 for the proposed work is 0.85, while in the related work [20] is 0.76, which is significantly better. In addition, the Naïve Bayes, Logistic Regression, and Random Forest have a slightly higher accuracy than the related work in [20]. It is clear that the proposed work has very good results.

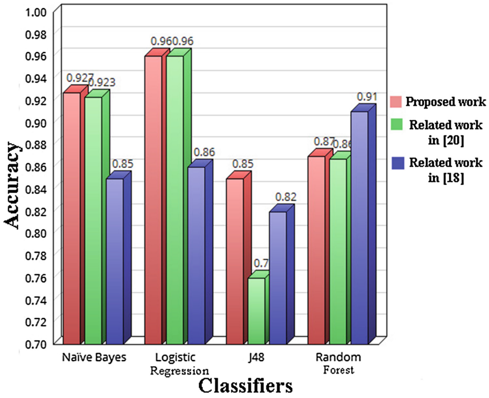


Fig. 4. Comparing accuracy value of proposed work with the related work in [20] and the related work in [18].

7 Conclusion and Future Work

In this paper, a semantic-based email filtering approach has been introduced. In this approach, we enhanced the traditional email filtering models by using WordNet ontology as a semantic resource for feature reduction. Experimental studies have been conducted using different classification algorithms. Enron dataset has been used in the experiments. A comparative study has been presented with other related work using the same dataset. The semantic-based feature reduction approach showed high performance with faster filtering execution and better accuracy. In the future work, we will enhance our proposed approach by considering some other semantic relations between term concepts for reducing the dimensionality of features.

References

1. Internet Threats Trend Report. Cyberoam® A SOPHOS Campany (2014)
2. Castillo, M.D., Serrano, J.I.: An interactive hybrid system for identifying and filtering unsolicited e-mail. In: Corchado, E., Yin, H., Botti, V., Fyfe, C. (eds.) IDEAL 2006. LNCS, vol. 4224, pp. 779–788. Springer, Heidelberg (2006). doi:10.1007/11875581_94
3. Hristea, F.T.: Semantic WordNet-based feature selection. In: Hristea, F.T. (ed.) The Naïve Bayes Model for Unsupervised Word Sense Disambiguation, pp. 17–33. Springer, Heidelberg (2013)
4. Lai, C.C., Tsai, M.C.: An empirical performance comparison of machine learning methods for spam e-mail categorization. In: Fourth International Conference on Hybrid Intelligent Systems, pp. 44–48. IEEE (2004)
5. Islam, M., Mahmud, A.A., Islam, M.: Machine Learning Approaches for Modeling Spammer Behavior. In: Kan, M.-Y., Lam, W., Nakov, P., Cheng, P.-J. (eds.) AIRS 2010. LNCS, vol. 6458, pp. 251–260. Springer, Heidelberg (2010)
6. Blanzieri, E., Bryl, A.: A survey of learning-based techniques of email spam filtering. Technical report DIT-06-056, University of Trento, Information Engineering and Computer Science Department (2008)
7. Mitchell, T.: Generative and discriminative classifiers: naive Bayes and logistic regression (2005). Manuscript http://www.cs.cm.edu/~tom/NewChapters.html
8. Renuka, D.K., Hamsapriya, T., Chakkaravarthi, M.R., Surya, P.L.: Spam classification based on supervised learning using machine learning techniques. In: International Conference on Process Automation, Control and Computing (PACC), pp. 1–7. IEEE (2011)
9. Shi, L., Wang, Q., Ma, X., Weng, M., Qiao, H.: Spam email classification using decision tree ensemble. J. Comput. Inf. Syst. **8**(3), 949–956 (2012)
10. Islam, M.R., Zhou, W.: Architecture of adaptive spam filtering based on machine learning algorithms. In: Jin, H., Rana, O.F., Pan, Y., Prasanna, V.K. (eds.) ICA3PP 2007. LNCS, vol. 4494, pp. 458–469. Springer, Heidelberg (2007). doi:10.1007/978-3-540-72905-1_41
11. Islam, R., Xiang, Y.: Email classification using data reduction method. In: Proceedings of the 5th International ICST Conference on Communications and Networking in China, pp. 1–5. IEEE (2010)
12. Bhat, V.H., Malkani, V.R., Shenoy, P.D., Venugopal, K.R., Patnaik, L.M.: Classification of email using beaks: behavior and keyword stemming. In: TENCON IEEE Region 10 Conference, pp. 1139–1143. IEEE (2011)
13. Lee, S.M., Kim, D.S., Kim, J.H., Park, J.S.: Spam detection using feature selection and parameters optimization. In: Intelligent and Software Intensive Systems International Conference on Complex, pp. 883–888. IEEE (2010)
14. Abdelrahim, A.A., Elhadi, A.A.E., Ibrahim, H., Elmisbah, N.: Feature selection and similarity coefficient based method for email spam filtering. In: International Conference on Computing, Electrical and Electronics Engineering (ICCEEE). IEEE (2013)

15. Ting, L., Qingsong, Y.: Spam feature selection based on the improved mutual information algorithm. In: Fourth International Conference on Multimedia Information Networking and Security (MINES). IEEE (2012)
16. Wang, R., Youssef, A.M., Elhakeem, A.K.: On some feature selection strategies for spam filter design. In: Canadian Conference on Electrical and Computer Engineering, pp. 2186–2189, CCECE 2006. IEEE (2006)
17. Shams, R., Mercer, R.E.: Classifying spam emails using text and readability features. In: 13th International Conference on Data Mining (ICDM). IEEE (2013)
18. More, S., Kulkarni, S.: Data mining with machine learning applied for email deception. In: International Conference on Optical Imaging Sensor and Security. IEEE (2013)
19. Sharaff, A., Nagwani, N.K., Dhadse, A.: Comparative study of classification algorithms for spam email detection. In: Emerging Research in Computing, Information, Communication and Applications, pp. 237–244. Springer, India (2016)
20. Bahgat, E.M., Rady, S., Gad, W.: An e-mail filtering approach using classification techniques. In: Gaber, T., Hassanien, A.E., El-Bendary, N., Dey, N. (eds.) The 1st International Conference on Advanced Intelligent System and Informatics. AISC, vol. 407, pp. 321–331. Springer, Heidelberg (2016). doi:10.1007/978-3-319-26690-9_29
21. Lu, Z., Ding, J.: An efficient semantic VSM based email categorization method. In: International Conference on Computer Application and System Modeling, vol. 11, pp. 511–525. IEEE (2010)
22. Yoo, S., Gates, D., Levin, L., Fung, S., Agarwal, S., Freed, M.: Using semantic features to improve task identification in email messages. In: Kapetanios, E., Sugumaran, V., Spiliopoulou, M. (eds.) NLDB 2008. LNCS, vol. 5039, pp. 355–357. Springer, Heidelberg (2008)
23. Tang, H.J., Yan, D.F., Yuan, T.I.A.N.: Semantic dictionary based method for short text classification. J. China Univ. Posts Telecommun. **20**, 15–19 (2013)
24. Enron-Spam datasets: CSMINING group, http://csmining.org/index.php/enron-spam-datasets.html. Accessed 7 July 2016
25. Hall, M., Frank, E., Holmes, G., Pfahringer, B., Reutemann, P., Witten, I.H.: The WEKA data mining software: an update. ACM SIGKDD Explor. Newsl. **11**(1), 10–18 (2009)