Available online at www.elixirpublishers.com (Elixir International Journal)

Computer Engineering

Elixir Comp. Engg. 71 (2014) 24699-24703

ISSN: 2229-712X

Performance Evaluation of Bagged Ensemble Classifiers for Spam Review Detection

M.Govindarajan

Department of Computer Science and Engineering, Annamalai University, Annamalai Nagar, Tamil Nadu, India.

ARTICLE INFO

Article history:

Received: 10 April 2014;

Received in revised form:

25 May 2014;

Accepted: 4 June 2014;

Keywords

Accuracy,
Bagging,
Genetic Algorithm (GA),
Naive Bayes (NB),
Sentiment Mining,
Support Vector Machine.

ABSTRACT

The area of sentiment mining (also called sentiment extraction, opinion mining, opinion extraction, sentiment analysis, etc.) has seen a large increase in academic interest in the last few years. Researchers in the areas of natural language processing, data mining, machine learning, and others have tested a variety of methods of automating the sentiment analysis process. The feasibility and the benefits of the proposed approaches are demonstrated by means of spam review that is widely used in the field of sentiment classification. In the proposed work, a comparative study of the effectiveness of ensemble technique is made for sentiment classification. Bagging and boosting are two relatively new but popular methods for producing ensembles. In this work, bagging is evaluated on spam review data set in conjunction with Naive Bayes (NB), Support Vector Machine (SVM), Genetic Algorithm (GA) as the base learners. The proposed bagged NB, SVM, GA is superior to individual approaches for spam review in terms of classification accuracy.

Introduction

Spam, or unsolicited bulk mail, varies in shape and form (Gomes et al. 2004). Nonetheless, it tends to exhibit a number of similar traits in terms of structure, content, and diffusion approaches. There is a perceptible business reason and justification for its proliferation: from a spammer's perspective, the effort and cost of sending a substantial number of emails is minimal, and the potential reach in terms of the magnitude of the available audience size is enormous (Paul et al. 2005). The prospective profitability is clearly described by Gansterer et al. (2005), and the numbers speak for themselves—the overall cost of spam in 2009 was estimated at 130 billion U.S. dollars (Ferris Research 2008).

Spam filtering in Internet email can operate at two levels, an individual user level or an enterprise level. An individual user is typically a person working at home and sending and receiving email via an ISP. Such a user who wishes to identify and filter spam email installs a spam filtering system on her individual PC. This system will either interface directly with their existing mail user agent (MUA) (more generally known as the mail reader) or more typically will act as a MUA itself with full functionality for composing and receiving email and for managing mailboxes.

Spam filtering algorithms can be split into two overall umbrella approaches, namely machine and non-machine-learning methods. Approaches applied in the former category include Bayesian classification, neural-networks, Markov-based models, and pattern discovery. Rule, signature- and hash-based identification, blacklisting, and traffic analysis, among others, are techniques that are employed with respect to non-machine-learning variants.

Machine-learning variants can normally achieve effectiveness with less manual intervention and are more adaptive to continued changes in spam patterns. Furthermore, they do not depend on any predefined rulesets analogous with non-machine-learning counterparts.

The rest of this paper is organized as follows: Section 2 describes the related work. Section 3 presents proposed methodology and Section 4 explains the performance evaluation measures. Section 5 focuses on the experimental results and discussion. Finally, results are summarized and concluded in section 6.

Related work

Machine-learning techniques involve the analysis of email, mostly at content level, and employ classification algorithms, such as Bayesian, Support Vector Machines, and others to segregate spam from legitimate email. These approaches have been extensively applied in spam filtering and exhibit different capabilities (Hunt and Carpinter 2006).

The Naive Bayes classifier is a simple statistical algorithm with a long history of providing surprisingly accurate results. It has been used in several spam classification studies (I. Androutsopoulos and J. Koutsias, 2000, I. Androutsopoulos, G. Paliouras, 2000, J. Hidalgo, 2002, K. Schneider, 2003), and has become somewhat of a benchmark. It gets its name from being based on Bayes' rule of conditional probability, combined with the "naive" assumption that all conditional probabilities are independent (I. Witten, E. Frank, 2000).

Support vector machines (SVMs) are relatively new techniques that have rapidly gained popularity because of the excellent results they have achieved in a wide variety of machine learning problems, and because they have solid theoretical underpinnings in statistical learning theory (N. Cristianini, B. Schoelkopf, 2002).

An elaborate genetic mechanism involving combinatorial association of a number of gene segments underlies the construction of these receptors. The overall immune response involves three evolutionary methods: gene library evolution generating effective antibodies, negative selection eliminating inappropriate antibodies and clonal selection cloning well performing antibodies (C. Wu, 2009).

Tele:
E-mail addresses: govind_aucse@yahoo.com

The ensemble technique, which combines the outputs of several base classification models to form an integrated output, has become an effective classification method for many domains (T. Ho, 1994; J. Kittler,, 1998).

In this work, bagging is evaluated on spam review in conjunction with NB, SVM, GA as the base learners. The performance of the proposed bagged (NB, SVM, GA) classifiers are examined in comparison with standalone classifiers.

Proposed Methodology

Several researchers have investigated the combination of different classifiers to from an ensemble classifier (D. Tax et al, 2000). An important advantage for combining redundant and complementary classifiers is to increase robustness, accuracy, and better overall generalization. This research work aims to make an intensive study of the effectiveness of ensemble techniques for sentiment classification tasks. In this work, first the base classifiers such as Naive Bayes (NB), Support Vector Machine (SVM), Genetic Algorithm (GA) are constructed to predict classification scores. The reason for that choice is that they are representative classification methods and very homogeneous techniques in terms of their philosophies and strengths. All classification experiments were conducted using 10 × 10-fold cross-validation for evaluating accuracy. Secondly, well known homogeneous ensemble technique is performed with base classifiers to obtain a very good generalization performance. The feasibility and the benefits of the proposed approaches are demonstrated by means of spam review that is widely used in the field of sentiment classification. A wide range of comparative experiments are conducted and finally, some in-depth discussion is presented and conclusions are drawn about the effectiveness of ensemble technique for sentiment classification.

This research work proposes new hybrid method for sentiment mining problem. A new architecture based on coupling classification methods using bagging classifier adapted to sentiment mining problem is defined in order to get better results. The main originality of the proposed approach is based on five main parts: Preprocessing phase, Document Indexing phase, feature reduction phase, classification phase and combining phase to aggregate the best classification results.

Data Pre-processing

Different pre-processing techniques were applied to remove the noise from our data set. It helped to reduce the dimension of our data set, and hence building more accurate classifier, in less time.

The main steps involved are i) document pre-processing, ii) feature extraction / selection, iii) model selection, iv) training and testing the classifier.

Data pre-processing reduces the size of the input text documents significantly. It involves activities like sentence boundary determination, natural language specific stop-word elimination and stemming. Stop-words are functional words which occur frequently in the language of the text (for example, „a‟, „the‟, „an‟, „of‟ etc. in English language), so that they are not useful for classification. Stemming is the action of reducing words to their root or base form. For English language, the Porter‟s stemmer is a popular algorithm, which is a suffix stripping sequence of systematic steps for stemming an English word, reducing the vocabulary of the training text by approximately one-third of its original size. For example, using the Porter‟s stemmer, the English word “generalizations” would subsequently be stemmed as “generalizations → generalization → generalize → general → gener”. In cases where the source documents are web pages, additional pre-processing is required to remove / modify HTML and other script tags.

Feature extraction / selection helps identify important words in a text document. This is done using methods like TF-IDF (term frequency-inverse document frequency), LSI (latent semantic indexing), multi-word etc. In the context of text classification, features or attributes usually mean significant words, multi-words or frequently occurring phrases indicative of the text category.

After feature selection, the text document is represented as a document vector, and an appropriate machine learning algorithm is used to train the text classifier. The trained classifier is tested using a test set of text documents. If the classification accuracy of the trained classifier is found to be acceptable for the test set, then this model is used to classify new instances of text documents.

Document Indexing

Creating a feature vector or other representation of a document is a process that is known in the IR community as indexing. There are a variety of ways to represent textual data in feature vector form, however most are based on word co-occurrence patterns. In these approaches, a vocabulary of words is defined for the representations, which are all possible words that might be important to classification. This is usually done by extracting all words occurring above a certain number of times (perhaps 3 times), and defining your feature space so that each dimension corresponds to one of these words.

When representing a given textual instance (perhaps a document or a sentence), the value of each dimension (also known as an attribute) is assigned based on whether the word corresponding to that dimension occurs in the given textual instance. If the document consists of only one word, then only that corresponding dimension will have a value, and every other dimension (i.e., every other attribute) will be zero. This is known as the ``bag of words" approach. One important question is what values to use when the word is present. Perhaps the most common approach is to weight each present word using its frequency in the document and perhaps its frequency in the training corpus as a whole. The most common weighting function is the *tfidf* (term frequency-inverse document frequency) measure, but other approaches exist. In most sentiment classification work, a binary weighting function is used. Assigning 1 if the word is present, 0 otherwise, has been shown to be most effective.

Dimensionality Reduction

Dimension Reduction techniques are proposed as a data pre-processing step. This process identifies a suitable low-dimensional representation of original data. Reducing the dimensionality improves the computational efficiency and accuracy of the data analysis.

Steps:

- ✓ Select the dataset.
- ✓ Perform discretization for pre-processing the data.
- ✓ Apply Best First Search algorithm to filter out redundant & super flows attributes.
- ✓ Using the redundant attributes apply classification algorithm and compare their performance.
- ✓ Identify the Best One.

Best first Search

Best First Search (BFS) uses classifier evaluation model to estimate the merits of attributes. The attributes with high merit value is considered as potential attributes and used for classification Searches the space of attribute subsets by augmenting with a backtracking facility. Best first may start

with the empty set of attributes and search forward, or start with the full set of attributes and search backward, or start at any point and search in both directions.

Existing Classification Methods

Naive Bayes (NB)

The Naïve Bayes assumption of attribute independence works well for text categorization at the word feature level. When the number of attributes is large, the independence assumption allows for the parameters of each attribute to be learned separately, greatly simplifying the learning process.

There are two different event models. The multi-variate model uses a document event model, with the binary occurrence of words being attributes of the event. Here the model fails to account for multiple occurrences of words within the same document, which is a more simple model. However, if multiple word occurrences are meaningful, then a multinomial model should be used instead, where a multinomial distribution accounts for multiple word occurrences. Here, the words become the events.

Support Vector Machine (SVM)

The support vector machine (SVM) is a recently developed technique for multi dimensional function approximation. The objective of support vector machines is to determine a classifier or regression function which minimizes the empirical risk (that is the training set error) and the confidence interval (which corresponds to the generalization or test set error).

Given a set of N linearly separable training examples $S = \{x_i \in R^n | i = 1,2,\ldots,N\}$, where each example belongs to one of the two classes, represented by $y_i \in \{\div 1, -1\}$, the SVM learning method seeks the optimal hyperplane **w . x** +b = 0, as the decision surface, which separates the positive and negative examples with the largest margins. The decision function for classifying linearly separable data is:

$$f(\mathbf{X}) = sign(W.X + b) \qquad (3.1)$$

Where w and b are found from the training set by solving a constrained quadratic optimization problem. The final decision function is

$$f(x) = sign\left(\sum_{i=1}^{N} a_i y_i (x_i . x) + b\right) \qquad (3.2)$$

The function depends on the training examples for which a_i s is non-zero. These examples are called support vectors. Often the number of support vectors is only a small fraction of the original data set. The basic SVM formulation can be extended to the non linear case by using the nonlinear kernels that maps the input space to a high dimensional feature space. In this high dimensional feature space, linear classification can be performed. The SVM classifier has become very popular due to its high performances in practical applications such as text classification and pattern recognition.

The support vector regression differs from SVM used in classification problem by introducing an alternative loss function that is modified to include a distance measure. Moreover, the parameters that control the regression quality are the cost of error C, the width of tube ε and the mapping function ϕ .

In this research work, the values for polynomial degree will be in the range of 0 to 5. In this work, best kernel to make the prediction is polynomial kernel with epsilon = 1.0E-12, parameter d=4 and parameter c=1.0.

Genetic Algorithm (GA)

The genetic algorithm is a model of machine learning which derives its behaviour from a metaphor of some of the mechanisms of evolution in nature. This done by the creation within a machine of a population of individuals represented by chromosomes, in essence a set of character strings.

The individuals represent candidate solutions to the optimization problem being solved. In genetic algorithms, the individuals are typically represented by n-bit binary vectors. The resulting search space corresponds to an n–dimensional boolean space. It is assumed that the quality of each candidate solution can be evaluated using a fitness function.

Genetic algorithms use some form of fitness-dependent probabilistic selection of individuals from the current population to produce individuals for the next generation. The selected individuals are submitted to the action of genetic operators to obtain new individuals that constitute the next generation. Mutation and crossover are two of the most commonly used operators that are used with genetic algorithms that represent individuals as binary strings. Mutation operates on a single string and generally changes a bit at random while crossover operates on two parent strings to produce two offsprings. Other genetic representations require the use of appropriate genetic operators.

The process of fitness-dependent selection and application of genetic operators to generate successive generations of individuals is repeated many times until a satisfactory solution is found. In practice, the performance of genetic algorithm depends on a number of factors including: the choice of genetic representation and operators, the fitness function, the details of the fitness-dependent selection procedure, and the various user-determined parameters such as population size, probability of application of different genetic operators, etc. The basic operation of the genetic algorithm is outlined as follows:

Procedure:

```
begin
t <- 0
initialize P(t)
while (not termination condition)
t <- t + 1
select P(t) from p(t - 1)
crossover P(t)
mutate P(t)
evaluate P(t)
  end
end.
```

The contribution relies on the association of all the techniques used in our method. First the small selection in grammatical categories and the use of bi-grams enhance the information contained in the vector representation, then the space reduction allows getting more efficient and accurate computations, and then the voting system enhance the results of each classifier. The overall process comes to be very competitive.

Proposed Bagged Ensemble Classifiers

Given a set D, of d tuples, bagging (Breiman, L. 1996a) works as follows. For iteration i (i =1, 2,…..k), a training set, D_i, of d tuples is sampled with replacement from the original set of tuples, D. The bootstrap sample D_i, by sampling D with replacement, from the given training data set D repeatedly. Each example in the given training set D may appear repeated times or not at all in any particular replicate training data set D_i. A classifier model, M_i, is learned for each training set, D_i. To classify an unknown tuple, X, each classifier, M_i, returns its

class prediction, which counts as one vote. The bagged (NB, SVM, GA), M^*, counts the votes and assigns the class with the most votes to X.

Algorithm: Bagged ensemble classifiers using bagging

Input:

- D, a set of d tuples.
- k = 3, the number of models in the ensemble.
- Base Classifier (NB, SVM, GA)

Output: A Bagged (NB, SVM, GA), M^*

Method:

1. for i = 1 to k do // create k models
2. Create a bootstrap sample, D_i, by sampling D with replacement, from the given training data set D repeatedly. Each example in the given training set D may appear repeated times or not at all in any particular replicate training data set D_i
3. Use D_i to derive a model, M_i;
4. Classify each example d in training data D_i and initialized the weight, W_i for the model, M_i, based on the accuracies of percentage of correctly classified example in training data D_i.
5. endfor

To use the bagged ensemble models on a tuple, X:

1. if classification then
2. let each of the k models classify X and return the majority vote;
3. if prediction then
4. let each of the k models predict a value for X and return the average predicted value;

Performance Evaluation Measures

Cross Validation Technique

Cross-validation, sometimes called rotation estimation, is a technique for assessing how the results of a statistical analysis will generalize to an independent data set. It is mainly used in settings where the goal is prediction, and one wants to estimate how accurately a predictive model will perform in practice. 10-fold cross validation is commonly used. In stratified K-fold cross-validation, the folds are selected so that the mean response value is approximately equal in all the folds.

Criteria for Evaluation

The primary metric for evaluating classifier performance is classification Accuracy - the percentage of test samples that are correctly classified. The accuracy of a classifier refers to the ability of a given classifier to correctly predict the label of new or previously unseen data (i.e. tuples without class label information). Similarly, the accuracy of a predictor refers to how well a given predictor can guess the value of the predicted attribute for new or previously unseen data.

Experimental results

Dataset Description

The data set used is the spam base set, consisting of tagged emails from a single email account. Given the data set train a base and hybrid classifier are trained to distinguish spam from regular email by fitting a distribution of the number of occurrences of each word for all the spam and non-spam e-mails.

Results and Discussion

Table 1: The performance of base and proposed bagged nb classifier for spam review data

Dataset	Classifiers	Accuracy
Spam -Review Data	Existing NB Classifier	88.54 %
	Proposed Bagged NB Classifier	91.13 %

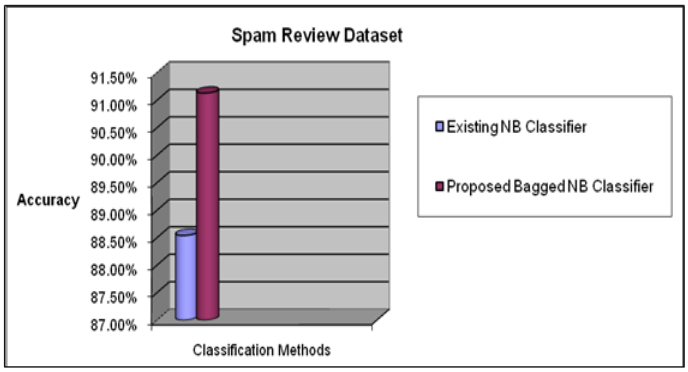

Figure 1: Classification Accuracy of Existing and Proposed Bagged NB Classifier using Spam Review Data

Table 2: The performance of base and proposed bagged svm classifier for spam review data

Dataset	Classifiers	Accuracy
Spam-Review Data	Existing SVM Classifiers	93.21 %
	Proposed Bagged SVM Classifiers	94.60 %

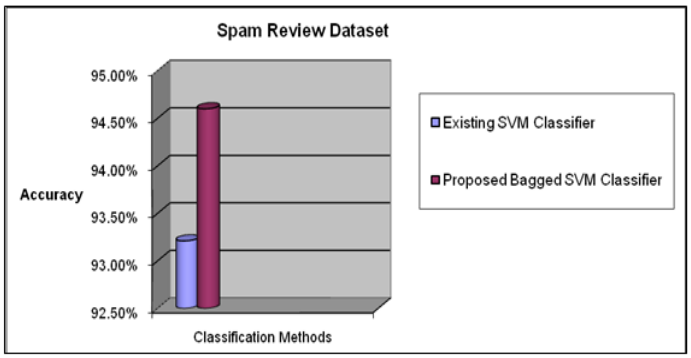

Figure 2: Classification Accuracy of Existing and Proposed Bagged SVM Classifier using Spam Review Data

Table 3: The performance of base and proposed bagged ga classifier for spam review data

Dataset	Classifiers	Accuracy
Spam-Review Data	Existing GA Classifier	87.22 %
	Proposed Bagged GA Classifier	90.67 %

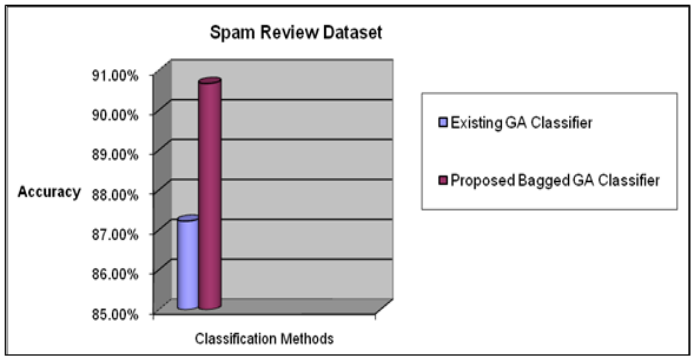

Figure 3: Classification Accuracy of Existing and Proposed Bagged GA Classifier using Spam Review Data

In this research work, new ensemble classification method is proposed using bagging classifier in conjunction with NB, SVM, GA as the base learner and the performance is analyzed in terms of accuracy. Here, the base classifiers are constructed using NB, SVM, GA. 10-fold cross validation (Kohavi, R, 1995) technique is applied to the base classifiers and evaluated classification accuracy. Bagging is performed with NB, SVM, GA to obtain a very good classification performance. Table 1 to 3 shows classification performance for spam review using

existing and proposed bagged NB, SVM, GA. The analysis of results shows that the proposed bagged NB, SVM, GA are shown to be superior to individual approaches for spam review in terms of classification accuracy. According to Fig. 1 to 3 proposed combined models show significantly larger improvement of classification accuracy than the base classifiers. This means that the combined methods are more accurate than the individual methods for the spam reviews.

Conclusions

In this research work, new combined classification methods are proposed using bagging classifier in conjunction with NB, SVM, GA as the base learners and the performance comparison has been demonstrated using spam reviews in terms of accuracy. This research has clearly shown the importance of using ensemble approach for spam reviews. An ensemble helps to indirectly combine the synergistic and complementary features of the different learning paradigms without any complex hybridization. Since all the considered performance measures could be optimized, such systems could be helpful in several real world sentiment mining applications. The high classification accuracy has been achieved for the ensemble classifiers compared to that of single classifiers. The proposed bagged NB, SVM, GA are shown to be significantly higher improvement of classification accuracy than the base classifiers. The spam reviews could be detected with high accuracy for homogeneous model. The future research will be directed towards developing more accurate base classifiers particularly for the sentiment mining applications.

Acknowledgment

Author gratefully acknowledges the authorities of Annamalai University for the facilities offered and encouragement to carry out this work. This work is supported by DST-SERB Fast track Scheme for Young Scientists by the Department of science and technology, Government of India, New Delhi.

References

[1] I. Androutsopoulos, J. Koutsias, (2000), "An evaluation of naïve bayesian anti-spam filtering". In Proceedings of the Workshop on Machine Learning in the New Information Age, 11th European Conference on Machine Learning (ECML 2000),pp 9–17, Barcelona, Spain.

[2] I. Androutsopoulos, G. Paliouras, (2000), "Learning to filter spam E-mail: A comparison of a naïve bayesian and a memorybased approach". In Proceedings of the Workshop on Machine Learning and Textual Information Access, 4th European Conference on Principles and Practice of Knowledge Discovery in Databases, pages 1–13, Lyon, France.

[3] Breiman, L. (1996a). "Bagging predictors", Machine Learning, 24(2), pp.123–140.

[4] N. Cristianini, B. Schoelkopf, (2002), "Support vector machines and kernel methods, the new generation of learningmachines". Artificial Intelligence Magazine, 23(3), pp. 31–41.

[5] Ferris Research. (2008), Industry Statistics-Ferris Research. http://www.ferris.com/research-library/industry-statistics/.

[6] Gansterer, W., Ilger, M., Lechner, P., Neumayer, R., and Straub, J. (2005), "Anti-spam methods: state of the art". Techn. rep., FA384018-1, Institute of Distributed and Multimedia Systems, University of Vienna.

[7] Gomes, L. H., Cazita, C., Almeida, J. M., Almeida, V., and Meira, J. (2004), "Characterizing a spam traffic", In Proceedings of the 4th ACM SIGCOMM Conference on Internet Measurement, ACM, New York, NY, pp. 356–369.

[8] J. Hidalgo, (2002), "Evaluating cost-sensitive unsolicited bulk email categorization". In Proceedings of SAC-02, 17th ACM Symposium on Applied Computing, Madrid, ES, pp. 615–620.

[9] T. Ho, J. Hull, S. Srihari, (1994), Decision combination in multiple classifier systems, IEEE Transactions on Pattern Analysis and Machine Intelligence, 16, pp. 66–75.

[10] Hunt, R. and Carpinter, J. (2006), "Tightening the net: a review of current and next generation spam filtering tools", Comput. Security, *2*5, pp. 566–578.

[11] J. Kittler, (1998), Combining classifiers: a theoretical framework, Pattern Analysis and Applications, 1, pp.18–27.

[12] Kohavi, R. (1995), "A study of cross-validation and bootstrap for accuracy estimation and model selection", Proceedings of International Joint Conference on Artificial Intelligence, pp.1137–1143.

[13] Paul, P. P., Judge, P., Alperovitch, D., and Yang, W. (2005), "Understanding and reversing the profit model of spam", In Proceedings of the Workshop on the Economics of Information Security (WEIS), pp. 1-11.

[14] K. Schneider, (2003), "A comparison of event models for naïve bayes anti-spam e-mail filtering". In Proceedings of the 10th Conference of the European Chapter of the Association for Computational Linguistics, pp. 307-314.

[15] D. Tax, M. Breukelen, R. Duin, and J. Kittler, (2000), "Combining multiple classifiers by averaging or by mutliplying?", Pattern Recognition, Vol 33, pp.1475-1485.

[16] I. Witten, E. Frank, (2000), "Data Mining: Practical Machine Learning Tools and Techniques with Java Implementations". Morgan Kaufmann.

[17] C. Wu, (2009), "Behavior-based spam detection using a hybrid method of rule-based techniques and neural networks", Expert Systems with Applications, Volume 36, Issue 3, Part 1, April 2009, pp. 4321–4330.