J Inf Process Syst, Vol.13, No.5, pp.1397~1409, October 2017
https://doi.org/10.3745/JIPS.02.0076

ISSN 1976-913X (Print)
ISSN 2092-805X (Electronic)

JOURNAL OF INFORMATION PROCESSING SYSTEMS JIPS

A Novel Statistical Feature Selection Approach for Text Categorization

Mohamed Abdel Fattah*,**

Abstract
For text categorization task, distinctive text features selection is important due to feature space high dimensionality. It is important to decrease the feature space dimension to decrease processing time and increase accuracy. In the current study, for text categorization task, we introduce a novel statistical feature selection approach. This approach measures the term distribution in all collection documents, the term distribution in a certain category and the term distribution in a certain class relative to other classes. The proposed method results show its superiority over the traditional feature selection methods.

Keywords
Electronic Texts, E-mail Filtering, Feature Selection, SMS Spam Filtering, Text Categorization

1. Introduction

Text classification (categorization) may be considered as one of the most interesting research points due to the necessity to categorize and organize the growing number of electronic texts on the internet. Normally, text categorization includes feature extraction step and a classifier which performs the categorization process based on labeled data. Text categorization has been exploited in some applications such as spam e-mail filtering [1,2], topic detection [3], web page categorization [4-6] and author identification [7,8]. To represent a document, a multi-dimensional feature vector is used. A weighted value such as TF.IDF is used to represent each dimension. Therefore many (may tend to be thousands) features are created for a certain text collection. Excessive number of features degrades classification accuracy and increase computational time. Hence, feature selection is a very essential phase in text classification task to improve the accuracy and speed up the computation. For feature selection, there are three approaches: filters, wrappers, and embedded approaches. Filters approach is computationally fast. It selects features that have the highest score values [9]. Wrappers approach evaluates features based on a certain search algorithm and learning model [10,11]. Compared with Filters approach, this approach is computationally expensive. Embedded approach integrates feature selection step into classifier training step. This approach is computationally less intensive than wrappers approach [9,12].

Manuscript received November 11, 2016; first revision February 21, 2017; second revision May 22, 2017; accepted May 29, 2017.

Corresponding Author: Mohamed Abdel Fattah (mohafi2003@helwan.edu.eg)
* Dept. of Computer Sciences, College of Computer Science and Engineering, Taibah University, Medina, Saudi Arabia (mohafi@hotmail.com)
** Dept. of Electronics Technology, Faculty of Industrial Education, Helwan University, Cairo, Egypt (mohafi2003@helwan.edu.eg)

A traditional text categorization paradigm includes preprocessing step, extraction of features, selection of features, and finally categorization phase. The preprocessing step normally includes tokenization, lowercase conversion, removing of stop words, and stemming. Using the bag-of-words approach, the feature extraction phase normally utilizes the vector space model [13-15]. Applying of stemming and stop word removal is used to decrease the feature vector dimensionality and increase the efficiency of the text classification task. In the classification step, classification models are used. Labeled documents are exploited to train the classification model, then the learned model is exploited to classify the unlabeled documents [16,17]. Support vector machine (SVM) [14,18] and naïve Bayes (NB) classifier [19,20] have been exploited in the text categorization field.

In the literature, there are many researches utilized feature selection paradigm for text categorization task. Feng et al. [21] utilized latent selection augmented naïve Bayes classifier for feature subset selection. In this method, the global selection index could be factorized to each local selection index. They could calculate the LSI for feature evaluation based on conjugate priors. By LSIs thresholding, the feature subset selection models could be pruned. Then by single feature model averages percentage product, the classifier could be achieved. This approach provided competitive results if compared with SVM classifier.

The feature selection method that is based on meaning measure was proposed by Tutkan et al. [22]. From Gestalt theory of human perception, this method is based on the Helmholtz principle. Based on this approach, a meaningfulness score value is assigned to important terms and used for text classification task in supervised and unsupervised manner. This approach performance is comparable with many common feature selection approaches performance.

Based on a term relative document frequencies in both negative and positive categories to find out a term rank, Rehman et al. [23] proposed a normalized difference measure method. Reasonable results have been achieved.

In [24], to eliminate redundant terms, the authors proposed to take the interactions of words into account. Hence, they proposed a feature selection method of two-stages, that employs a feature ranking as a first stage and a feature subset selection method as a second stage. When this approach performance is evaluated based on balanced error rate, this approach results are comparable with the results of bi-normal separation + Markov blanket filter and information gain + Markov blanket filter.

Seijo-Pardo et al. [25] provided homogeneous and heterogeneous approaches. In homogeneous case, the authors distributed the dataset over several nodes and used the same feature selection algorithm with different training data. On the other hand, in heterogeneous case with the same training data, they used different feature selection algorithms. SVM as a classifier was used in testing which provided comparable performance with individual feature selection algorithms.

Part-of-speech and unigram-based feature sets were exploited for sentiment analysis in [26]. For feature ranking, five algorithms were employed to create feature vector. Then, to get a new feature vector, an ordinal-based integration of different feature vectors was proposed. The results of the part-of-speech patterns were better than that of unigram-based features.

The work of Lu et al. [27] exploited particle swarm optimization algorithm based on functional inertia weight and constant constriction factor to optimize feature selection algorithms. Then, asynchronously improved PSO and synchronously improved PSO models are proposed based on both functional constriction factor and functional inertia weight. For text categorization task, asynchronously improved PSO model achieved best results.

We have little works which handle feature sub-set selection problem for Arabic language text categorization tasks. Syiam et al. [28] have investigated the effect of some feature sub-set selection with Rocchio and kNN classifiers. They found that the use of any of those feature sub-set selection metrics separately provided close results for the Arabic text classification tasks. However, they have not exploited support vector machine classifier that is already considered to be superior to other classifiers.

Most of the other Arabic text classification works [29-32] have used feature sub-set selection metrics without any feature sub-set selection comparison. On the other hand, [33] investigated the effect of (Ngl, Gss score, Or, Ig, Df, and Chi) features based on support vector machine classifier for Arabic text classification and found that Chi provides good results using macro-averaging F1 measure, and macro-averaging recall. However, Ngl and Chi provide better results using macro-averaging precision.

In this study, we propose a novel filter based feature selection approach for text categorization. Our new approach selects distinctive text features and eliminates uninformative ones. Our approach is compared with some filter based methods such as mutual information, X^2 statistic, odds ratio, information gain, Gini index and weighted log likelihood ratio. We established the comparisons based on different datasets to be able to observe our approach effectiveness under different conditions. The experimental results proved that our approach provides good performance compared to the above-mentioned methods in terms of processing time, rate of dimension reduction and classification accuracy.

This manuscript is organized as follows: Section 2 describes the comparable feature selection approaches. Section 3 shows the proposed approach. Section 4 provides the results. Finally, Section 5 provides the conclusion of this work and the possible future works.

2. Traditional Feature Selection Approaches

For the distinctive text features selection in text categorization, there are many filter based techniques. Of these approaches, mutual information, chi square, odds ratio, Gini index, information gain, and weighted log likelihood ratio have been exploited [34-37].

2.1 Mutual Information

Depending on mutual information (MI), Deng et al. [38] and Church and Hanks [39] created a term weighting. Given term t_i and document set D^c (set of documents in a specific category), the MI between them is calculated as follows:

$$MI(t_i, D^c) = log\frac{P(t_i,D^c)}{P(t_i)\times P(D^c)} \tag{1}$$

The MI term weighting could be calculated as:

$$MI(t_i, D^c) \approx log\frac{\frac{n_c(t_i)}{|D|}}{\frac{D(t_i)}{|D|}\times\frac{n_c}{|D|}} = log\frac{n_c(t_i)\times|D|}{D(t_i)\times n_c} \tag{2}$$

n_c is the number of documents in class (c), $D(t_i)$ is the number of documents that contain term (t_i) in all classes, $n_c(t_i)$ is the number of documents which belong to class (c) and contain the term (t_i), and $|D| =$

the training corpus total number of documents.

The MI term weighting is given as:

$$TW_{MI}(t_i) = \max_c \{MI(t_i, D^c)\} \tag{3}$$

2.2 X² Statistic

CHI (X^2 statistic) specifies the independence lack between two random variables (D^c *and* t_i) [38,40]. The X^2 statistic between term t_i and document set D^c associated with a certain class (c) is given as follows:

$$CHI(t_i, D^c\ ) = \frac{|\ D\ | \times [(n_c(t_i) \times n_{\bar{c}}(\overline{t_\imath}) - n_{\bar{c}}(t_i) \times n_c(\overline{t_\imath})]^2}{[n_c(t_i) + n_c(\overline{t_\imath})] \times [n_{\bar{c}}(t_i) + n_{\bar{c}}(\overline{t_\imath})] \times [n_c(t_i) + n_{\bar{c}}(t_i)] \times [n_c(\overline{t_\imath}) + n_{\bar{c}}(\overline{t_\imath})]} \tag{4}$$

With $n_{\bar{c}}(t_i)$ is the documents number that are not belonging to class (c) and contain (t_i). $n_c(\overline{t_\imath})$ is the documents number which belong to class (c) but do not contain (t_i). $n_{\bar{c}}(\overline{t_\imath})$ is the documents number which neither belong to class (c) nor contain term (t_i).

Depending on *CHI* statistic, the term weighting is calculated as:

$$TW_{CHI}(t_i) = \max_c \{CHI(t_i, D^c)\} \tag{5}$$

2.3 Odds Ratio

In information retrieval, odds ratio (OR) is exploited [38,39,41]. In text categorization, to categorize documents, we use appearance of words as feature parameters. Given a term (t_i) and documents set associated with a specific class D^c, the odds ratio is calculated as:

$$OR(t_i, D^c\ ) = log \frac{P(t_i|D^c)(1 - P(t_i|\overline{D^c}))}{(1 - P(t_i|D^c))P(t_i|\overline{D^c})} \tag{6}$$

$$OR(t_i, D^c\ ) \approx log \frac{n_c(t_i) \times (|\ D\ | - n_c - n_{\bar{c}}(t_i))}{(n_c - n_c(t_i)) \times n_{\bar{c}}(t_i)} \tag{7}$$

Based on OR, the term weighting is calculated as:

$$TW_{OR}(t_i) = \max_c\ \{OR(t_i, D^c)\} \tag{8}$$

2.4 Information Gain

To make the right categorization decision on any category, information gain (IG) is exploited to measure the amount of information associated with the absence or presence of a term [42]. Information Gain for a term t_i is given as in the following formula:

$$IG(t_i) = -\textstyle\sum_{j=1}^{M} P(C_j) logP(C_j)\ + P(t_i) \sum_{i=j}^{M} P(C_j|t_i) logP(C_j|t_i)\ + P(\overline{t_\imath}) \sum_{i=j}^{M} P(C_j|\overline{t_\imath}) logP(C_j|\overline{t_\imath}) \tag{9}$$

$$IG(t_i) \approx -\sum_c \frac{n_c}{|D|} log \frac{n_c}{|D|} + \frac{D(t_i)}{|D|} \sum_c \frac{n_c(t_i)}{D(t_i)} log \frac{n_c(t_i)}{D(t_i)} + \frac{|D| - \mathrm{D}(t_i)}{|D|} \sum_c \frac{n_c(\overline{t_i})}{|D| - \mathrm{D}(t_i)} log \frac{n_c(\overline{t_i})}{|D| - \mathrm{D}(t_i)} \tag{10}$$

With, $P(\overline{t_i})$ and $P(t_i)$ are the probabilities of absence and presence of term t_i , $P(C_j)$ is class C_j probability, $P(C_j|\overline{t_i})$ and $P(C_j|t_i)$ are the conditional probabilities of the category C_j given absence and presence of the term t_i, respectively and M is the number of classes.

$$TW_{IG}(t_i) = IG(t_i) \tag{11}$$

2.5 Gini Index

In decision trees, to find the best split, Gini Index (GI) is used [34]. GI for a term t_i is given as in the following formula:

$$GI(t_i) = \sum_{j=1}^{M} P(t_i|C_j)^2 P(C_j|t_i)^2 \approx \sum_c (\frac{(n_c(t_i)|D|)^2}{n_c D(t_i)})^2 \tag{12}$$

With $P(t_i|C_j)$ is the term t_i probability given category C_j, $P(C_j|t_i)$ is class C_j probability given term t_i.

$$TW_{GI}(t_i) = GI(t_i) \tag{13}$$

2.6 Weighted Log Likelihood Ratio

For text categorization, weighted log likelihood ratio (WLLR) is effective [38,43]. For a term (t_i) and a set of documents D^c associated with a certain class c, the WLLR is calculated as:

$$WLLR(t_i, D^c) = P(t_i|D^c) log \frac{P(t_i|D^c)}{P(t_i|\overline{D^c})} \tag{14}$$

$$WLLR(t_i, D^c) \approx \frac{n_c(t_i)}{n_c} log \frac{n_c(t_i) \times (|D| - n_c)}{n_{\bar{c}}(t_i) n_c} \tag{15}$$

Based on $WLLR$, the term weighting is calculated as:

$$TW_{WLLR}(t_i) = \max_c \quad \{WLLR(t_i, D^c)\} \tag{16}$$

3. The Proposed Approach

3.1 The Proposed Feature Selection Method

A good filter for feature selection must be able to assign low score values to non-discriminative terms to be filtered out and assign high score values to discriminative terms. The following proposed formula might be exploited to rank text terms based on their discrimination ability for classification task:

$$W_{pro}(t_i\,, D_c) = \frac{n_c^2(t_i)}{D(t_i)n_c[1+\frac{n_c(\bar{t_i})}{n_c}+\frac{n_{\bar{c}}(t_i)}{D-n_c}]} \tag{17}$$

The term $\frac{n_c(\bar{t_i})}{n_c}$ in the denominator of the above formula decreases the weight score value of a term which is rarely appearing in a certain category and is not appearing in the rest categories. The term $\frac{n_{\bar{c}}(t_i)}{D-n_c}$ in the denominator of the above formula is used to decrease the weight score values of the terms that appear in all classes. The term $\frac{n_c(t_i)}{n_c}$ in the nominator of the above formula is used to increase the weight score values of the terms that appear in the most of a certain class documents. The term $\frac{n_c(t_i)}{D(t_i)}$ in the nominator of the above formula is used to increase score values of the terms that appear in the most of a certain class documents while rarely appear in the rest of class documents. The term $\frac{n_c(t_i)}{n_c}$ provides the distribution of the term in all documents of a certain class, whereas the term $\frac{n_c(t_i)}{D(t_i)}$ gives the distribution of the term in a specific category relative to other classes. The term weighting based on the proposed approach is calculated as:

$$TW_{pro}(t_i) = \max_{c} \quad \{W_{pro}(t_i, D^c)\} \tag{18}$$

All terms are ranked based on the above formula.

3.2 Decision Tree Classifier (DTC)

In decision tree, categories are consecutively rejected till we reach an accepted category [38]. Each class corresponds to unique region in the feature space. Binary classification tree is the most commonly exploited one. In binary classification tree, through sequence of yes/no decisions along nodes path, an unknown feature vector is assigned to a specific category. At any node, the splitting rule is to provide as much decrease in node impurity as possible. To define impurity, we exploit entropy that may be calculated as:

$$I(t) = -\sum_{j=1}^{M} P\big(C_j|t\big)log_2P\big(C_j|t\big) \tag{19}$$

$P\big(C_j|t\big)$ is the conditional probability that a certain feature vector associated with a certain node *t* belongs to a specific class *C* for *j*=1 *to M.* To perform split at a certain node N_t, N_{tN} points are sent to *"No"* node and N_{tY} points are sent to *"Yes"* node. The following formula measures the node impurity reduction:

$$\Delta I(t) = I(t) - \frac{N_{tY}}{N_t}I(t_{Yes}) - \frac{N_{tN}}{N_t}I(t_{No}) \tag{20}$$

where $I(t_{No})$ *and* $I(t_{Yes})$ are *"No"* and *"Yes"* node impurity respectively. Splitting process is stopped when we obtain a single class after a split or when the node impurity highest decrease is less than a specific threshold. A class assignment is considered for a leaf node. We consider the majority rule to assign a leaf to a specific category that has maximum number of vectors.

3.3 Multi-Class Support Vector Machine Classifier (MSVM)

For binary classification tasks, SVMs were originally created [44,45]. For multiclass problems, appropriate approach should be used. In this work, we use one class against the rest. In this approach, L hyper-planes are constructed. Each hyper-plane separates one class from the rest. Then an observed vector X is mapped to a category based on the highest generated decision function. By focusing on the training data, SVM tries to locate the optimal separating hyper-plane among classes. Therefore, with small training sets, high classification accuracy is obtained. For a binary classification task, let $\{x_i, y_i\}$ represent training data, $y_i \in \{-1, +1\}$, $i = 1,2,\ldots, N.$, N = number of training data. $y_i = +1$ and -1 for classes ω_1 and ω_2 respectively. With bias w_0, assume a vector w can separate the categories without error:

$$f(x) = w.x + w_0 = 0 \tag{21}$$

To find this hyper-plane, the condition $y_i\ (w.x_i + w_0\ ) -1 \geq 0$ should be satisfied. The vectors are located on two hyper-planes that are parallel to the optimal and calculated as:

$$w.x_i + w_0 = \pm 1 \tag{22}$$

Then the margin may be calculated as: $\frac{2}{||w||}$. We may find the optimal hyper-plane from the following formula:

$$Minimize\ \frac{1}{2}||w||^2 \tag{23}$$

This problem may be solved using Lagrangian formula as follows:

$$Maximize\ \sum_{i=1}^{N} \lambda_i - \frac{1}{2}\sum_{i,j=1}^{N} \lambda_i\, \lambda_j y_i y_j (x_i . x_j) \tag{24}$$

Subject to $\sum_{i=1}^{N} \lambda_i\, y_i = 0\ and\ \lambda_i \geq 0, i = 1,2, \ldots N$, since λ_i is Lagrange multiplier. Then the optimal function becomes:

$$f(x) = \sum_{i \in S} \lambda_\mathrm{i} y_i\ (x_i . x) + w_0 \tag{25}$$

with S is a training data subset for nonzero Lagrange multipliers. A cost function is used to combine the minimization of error criteria and the maximization of margin by exploiting a set of variables ξ_i. The cost function can be calculated as:

$$Minimize\ J(w, w_0, \xi) = \frac{1}{2}\left||w|\right|^2 + C\sum_{i=1}^{N} \xi_i \tag{26}$$

Then the vectors inner product in the mapping space may be achieved using the function:

$$\emptyset(x)\emptyset(z) = K(x, z) \tag{27}$$

$K(x, z)$ is the kernel function. We have exploited polynomial kernel function as follows:

$$K(x_i, x_j) = (\gamma . x_i^T x_j + r)^d, \gamma > 0 \tag{28}$$

where d, γ, & r are kernel function parameters. Then the dual optimization task may be calculated as:

$$Maximize\ \sum_{i=1}^{N} \lambda_i - \frac{1}{2} \sum_{i,j=1}^{N} \lambda_i \lambda_j y_i y_j K(x_i . x_j) \tag{29}$$

Subject to $\sum_{\mathrm{i}=1}^{\mathrm{N}} \lambda_{\mathrm{i}}\, \mathrm{y}_{\mathrm{i}}=0$ *and* $\lambda_i \geq 0, i = 1.2 \ldots . N$. Then the final classifier formula is defined as:

$$f(x) = \sum_{i=S} \lambda_i\, y_i K(x_i x) + w_0 \tag{30}$$

The process of classification depends upon one against others based on the above formula.

4. Experimental Results

4.1 Training and Testing Data

In this work, two data sets have been exploited to measure the proposed approach performance. The first dataset is Reuters-21578 collection which is an imbalanced (different documents number in each class) data set. Reuter's categories have been manually classified into 135 subclasses. We used the ten most frequent categories (Wheat, Trade, Ship, Money-fx, Interest, Grain, Earn, Crude, Corn, and Acquisition). The second dataset is the 20 Newsgroups collection which is a balanced (number of documents per category are equal) dataset. This corpus contains 18,828 documents in 20 different categories.

Preprocessing step is established to make the two datasets suitable for experiments. Stop-word removal (e.g., conjunctions, prepositions, articles, etc.), lowercase conversion (since uppercase and lowercase forms of words are assumed to have no difference), and stemming (since derived words are semantically similar) to their stem forms are considered.

4.2 Performance Measure

To measure the overall performance, micro-average and macro-average of F-measure are used. Firstly, precision, recall and F-measure are defined as follows:

$$P = \frac{Correct}{Correct+Wrong}\ , \quad R = \frac{Correct}{Correct+Missing}\ , \quad F = \frac{2.P.R}{P+R} \tag{31}$$

The macro-average F measure (F^M) is defined as follows:

$$F^M = \frac{1}{m} \sum_{k=1}^{m} F(C_k) \tag{32}$$

The micro-average F measure (F^μ) is defined as follows:

$$F^\mu = \frac{2.P^\mu.R^\mu}{P^\mu+R^\mu} \tag{33}$$

where R^μ and P^μ are recall and precision over all the categorization decisions in the entire dataset rather than individual categories.

4.3 Performance Analysis

In these experiments, we vary features number based on each selection method. Then we fed the feature vectors to DTC and MSVM classifiers. Results based on micro and macro F score values are shown in Tables 1 and 2 for each dataset. From Tables 1 and 2, the proposed approach results outperform most of the other approach results in both datasets.

Table 1. Performance measure for Reuters collection based on DTC and MSVM classifiers for different feature parameter size

Performance criteria	Feature parameter size											
	50		100		200		300		400		500	
	F^M	F^{μ}	F^M	F^{μ}	F^M	F^{μ}	F^M	F^{μ}	F^M	F^{μ}	F^M	F^{μ}
DTC												
MI	56.8	80.9	57.7	81.8	57.8	81.9	57.9	82.4	58.1	82.6	58.3	82.4
CHI	56.5	80.3	56.7	81.4	57.1	81.6	57.3	82.5	57.4	82.7	57.7	82.3
OR	57.8	81.2	57.6	81.6	57.5	82.7	57.7	83.1	57.6	83.3	57.8	82.9
IG	58.1	81.4	58.2	81.5	58.4	81.8	58.8	82.6	58.9	83.1	59.4	82.7
GI	58.3	81.7	58.5	82.1	58.7	81.9	58.6	82.7	58.8	83.2	59.1	83.1
WLLR	57.2	80.5	57.6	81.4	58.2	82.0	58.7	82.3	58.5	82.8	58.7	82.6
pro	60.8	82.3	59.9	82.4	60.3	82.7	59.8	83.2	60.6	83.5	60.7	82.8
MSVM												
MI	57.6	83.3	59.7	84.8	61.6	84.7	61.9	84.6	61.7	84.8	61.6	84.5
CHI	57.4	83.1	60.3	84.5	62.5	84.6	63.4	84.4	63.2	84.5	63.3	84.3
OR	58.7	83.5	61.4	84.7	63.3	84.8	64.2	84.7	63.9	84.6	63.8	84.4
IG	59.2	83.7	62.3	85.1	64.2	85.3	64.9	85.2	64.7	85.3	64.8	85.4
GI	59.3	83.8	62.4	85.3	64.5	85.4	65.3	85.5	65.2	85.4	65.3	85.3
WLLR	58.3	82.9	61.8	84.5	63.9	84.7	64.7	84.6	64.5	84.7	64.3	84.6
pro	61.7	83.9	62.6	85.6	64.8	85.8	65.6	85.6	65.3	85.7	65.2	85.5

Table 2. Performance measure for Newsgroups collection based on DTC and MSVM classifiers for different feature parameter size

Performance criteria	Feature parameter size											
	50		100		200		300		400		500	
	F^M	F^{μ}	F^M	F^{μ}	F^M	F^{μ}	F^M	F^{μ}	F^M	F^{μ}	F^M	F^{μ}
DTC												
MI	95.9	96.3	97.2	96.8	97.3	97.4	97.4	97.2	97.2	97.3	97.1	96.8
CHI	96.5	96.4	97.3	97.1	97.5	97.5	97.4	97.3	97.3	97.5	96.8	96.9
OR	96.6	96.5	97.5	96.8	97.6	97.4	97.5	97.6	97.4	97.6	96.9	97.4
IG	96.7	96.7	97.3	97.4	97.5	97.6	97.5	97.4	97.3	97.6	97.2	97.3
GI	96.8	97.2	97.6	97.5	97.6	97.5	97.6	97.5	97.5	97.5	97.4	97.6
WLLR	97.7	97.4	97.7	97.5	97.7	97.8	97.7	97.6	97.6	97.8	97.5	97.6
pro	96.9	97.5	97.5	97.7	97.9	97.7	98.1	98.0	97.9	97.8	97.8	97.8
MSVM												
MI	97.4	97.3	97.3	97.4	97.3	97.4	97.3	97.5	97.4	97.4	97.3	97.2
CHI	97.5	97.4	97.4	97.5	97.4	97.6	97.3	97.7	97.3	97.7	97.3	97.6
OR	97.6	97.5	97.6	97.6	97.5	97.5	97.4	97.6	97.5	97.7	97.4	97.5
IG	97.7	97.7	97.7	97.7	97.7	97.7	97.6	97.6	97.7	97.5	97.6	97.4
GI	97.7	97.8	97.6	97.7	97.5	97.6	97.6	97.7	97.6	97.6	97.7	97.6
WLLR	97.7	97.6	97.6	97.7	97.7	97.7	97.8	97.8	97.7	97.7	97.6	97.6
pro	97.9	97.8	97.8	97.7	97.9	98.1	98.0	97.9	97.8	97.9	97.7	97.8

4.4 Discussion

As shown in the previously mentioned results, the most of the proposed approach results are better than other approaches result in terms of Macro-F and Micro-F score values. For a very small feature parameter size, the processing time is low. On the other hand, the accuracies for moderate feature parameter size are better. For a large feature parameter size, the results are lower or almost the same as moderate feature parameter size. The benefit of decreasing feature parameter size is to reduce the processing time as well as increase the system efficiency based on accuracy. The most effective text features have been selected in the proposed method to achieve better results. The results of MSVM classifier are slightly better than the results of DTC classifier in both datasets.

5. Conclusions & Future Works

In this work, a novel statistical feature selection method for text classification was presented. The proposed method specified distinctive text features and eliminates uninformative ones. The proposed approach was compared with some successful filter based methods such as mutual information, X^2 statistic, odds ratio, information gain, Gini Index, and weighted log likelihood ratio. Two datasets have been used for comparison; hence performance of our approach could be measured under different conditions. The experimental results showed that our approach provides a competitive performance compared with other mentioned methods in terms of processing time, rate of dimension reduction, and classification accuracy.

In the future work, enhancement of the proposed approach will be considered with other successful methods to create a hybrid model.

References

[1] S. Gunal, S. Ergin, M. B. Gulmezoglu, and O. N. Gerek, "On feature extraction for spam e-mail detection," in *International Workshop on Multimedia Content Representation, Classification and Security*, Berlin, Germany: Springer, 2006, pp. 635-642.

[2] T. S. Guzella and W. M. Caminhas, "A review of machine learning approaches to Spam filtering," *Expert Systems with Applications*, vol. 36, no. 7, pp. 10206-10222, 2009.

[3] D. B. Bracewell, J. Yan, F. Ren, and S. Kuroiwa, "Category classification and topic discovery of Japanese and English news articles," *Electronic Notes in Theoretical Computer Science*, vol. 225, pp. 51-65, 2009.

[4] I. Anagnostopoulos, C. Anagnostopoulos, V. Loumos, and E. Kayafas, "Classifying web pages employing a probabilistic neural network," *IEE Proceedings-Software*, vol. 151, no. 3, pp. 139-150, 2004.

[5] R. C. Chen and C. H. Hsieh, "Web page classification based on a support vector machine using a weighted vote schema," *Expert Systems with Applications*, vol. 31, no. 2, pp. 427-435, 2006.

[6] S. A. Ozel, "A web page classification system based on a genetic algorithm using tagged-terms as features," *Expert Systems with Applications*, vol. 38, no. 4, pp. 3407-3415, 2011.

[7] N. Cheng, R. Chandramouli, and K. P. Subbalakshmi, "Author gender identification from text," *Digital Investigation*, vol. 8, no. 1, pp. 78-88, 2011.

[8] E. Stamatatos, "Author identification: using text sampling to handle the class imbalance problem," *Information Processing & Management*, vol. 44, no. 2, pp. 790-799, 2008.

[9] I. Guyon and A. Elisseeff, "An introduction to variable and feature selection," *Journal of Machine Learning Research*, vol. 3, pp. 1157-1182, 2003.

[10] S. Gunal, O. N. Gerek, D. G. Ece, and R. Edizkan, "The search for optimal feature set in power quality event classification," *Expert Systems with Applications*, vol. 36, no. 7, pp. 10266-10273, 2009.

[11] R. Kohavi and G. H. John, "Wrappers for feature subset selection," *Artificial Intelligence*, vol. 97, no. 1-2, pp. 273-324, 1997.

[12] Y. Saeys, I. Inza, and P. Larranaga, "A review of feature selection techniques in bioinformatics," *Bioinformatics*, vol. 23, no. 19, pp. 2507-2517, 2007.

[13] G. Salton, A. Wong, and C. S. Yang, "A vector space model for automatic indexing," *Communications of the ACM*, vol. 18, no. 11, pp. 613-620, 1975.

[14] T. Joachims, "A probabilistic analysis of the Rocchio algorithm with TFIDF for text categorization," in *Proceedings of the 14th International Conference on Machine Learning*, Nashville, TN, 1997, pp. 143-151.

[15] A. McCallum and K. Nigam, "A comparison of event models for naive Bayes text classification," in *Proceeding of the AAAI-98 Workshop on Learning for Text Categorization*, Madison, WI, 1998, pp. 41-48.

[16] M. A. Fattah, F. Ren, and S. Kuroiwa, "Effects of phoneme type and frequency on distributed speaker identification and verification," *IEICE Transactions on Information and Systems*, vol. E89-D, no. 5, pp. 1712-1719, 2006.

[17] M. A. Fattah, "A hybrid machine learning model for multi-document summarization," *Applied Intelligence*, vol. 40, no. 4, pp. 592-600, 2014.

[18] D. D. Lewis, "Naive (Bayes) at forty: the independence assumption in information retrieval," in *European Conference Machine Learning ECML-98*, Berlin, Germany: Springer, 1998, pp. 4-15.

[19] T. Joachims, "Text categorization with support vector machines: learning with many relevant features," in *European Conference Machine Learning ECML-98*, Berlin, Germany: Springer, 1998, pp. 137-142.

[20] Y. Yang and X. Liu, "A re-examination of text categorization methods," in *Proceedings of the 22nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, Berkeley, CA, 1999, pp. 42-49.

[21] G. Feng, J. Guo, B. Y. Jing, and T. Sun, "Feature subset selection using naive Bayes for text classification," *Pattern Recognition Letters*, vol. 65, pp. 109-115, 2015.

[22] M. Tutkan, M. C. Ganiz, and S. Akyokus, "Helmholtz principle based supervised and unsupervised feature selection methods for text mining," *Information Processing & Management*, vol. 52, no. 5, pp. 885-910, 2016.

[23] A. Rehman, K. Javed, and H. A. Babri, "Feature selection based on a normalized difference measure for text classification," *Information Processing & Management*, vol. 53, no. 2, pp. 473-489, 2017.

[24] K. Javed, S. Maruf, and H. A. Babri, "A two-stage Markov blanket based feature selection algorithm for text classification," *Neurocomputing*, vol. 157, pp. 91-104, 2015.

[25] B. Seijo-Pardo, I. Porto-Diaz, V. Bolon-Canedo, and A. Alonso-Betanzos, "Ensemble feature selection: homogeneous and heterogeneous approaches," *Knowledge-Based Systems*, vol. 118, pp. 124-139, 2017.

[26] A. Yousefpour, R. Ibrahim, and H. N. A. Hamed, "Ordinal-based and frequency-based integration of feature selection methods for sentiment analysis," *Expert Systems with Applications*, vol. 75, pp. 80-93, 2017.

[27] Y. Lu, M. Liang, Z. Ye, and L. Cao, "Improved particle swarm optimization algorithm and its application in text feature selection," *Applied Soft Computing*, vol. 35, pp. 629-636, 2015.

[28] M. M. Syiam, Z. T. Fayed, and M. B. Habib, "An intelligent system for Arabic text categorization," *International Journal of Intelligent Computing and Information Sciences*, vol. 6, no. 1, pp. 1-19, 2006.

[29] G. Kanaan, R. Al-Shalabi, S. Ghwanmeh, and H. Al-Ma'adeed, "A comparison of text-classification techniques applied to Arabic text," *Journal of the Association for Information Science and Technology*, vol. 60, no. 9, pp. 1836-1844, 2009.

[30] L. Khreisat, "A machine learning approach for Arabic text classification using N-gram frequency statistics," *Journal of Informetrics*, vol. 3, no. 1, pp. 72-77, 2009.

[31] M. J. Bawaneh, M. S. Alkoffash, and A. I. Al Rabea, "Arabic text classification using K-NN and naive Bayes," *Journal of Computer Science*, vol. 4, no. 7, pp. 600-605, 2008.

[32] S. Al-Harbi, A. Almuhareb, A. Al-Thubaity, M. S. Khorsheed, and A. Al-Rajeh, "Automatic Arabic text classification," in *Proceedings of the 9th International Conference on the Statistical Analysis of Textual Data*, Lyon, France, 2008, pp. 77-83.

[33] A. M. Mesleh, "Support vector machines based Arabic language text classification system: Feature selection comparative study," in *Advances in Computer and Information Sciences and Engineering*, Dordrecht, Netherlands: Springer, 2008, pp. 11-16.

[34] H. Ogura, H. Amano, and M. Kondo, "Feature selection with a measure of deviations from Poisson in text categorization," *Expert Systems with Applications*, vol. 36, no. 3(Part 2), pp. 6826-6832, 2009.

[35] W. Shang, H. Huang, H. Zhu, Y. Lin, Y. Qu, and Z. Wang, "A novel feature selection algorithm for text categorization," *Expert Systems with Applications*, vol. 33 no. 1, pp. 1-5, 2007.

[36] Y. Yang and J. O. Pedersen, "A comparative study on feature selection in text categorization," in *Proceedings of the 14th International Conference on Machine Learning*, Nashville, TN, 1997, pp. 412-420.

[37] M. A. Fattah, "New term weighting schemes with combination of multiple classifiers for sentiment analysis," *Neurocomputing*, vol. 167, pp. 434-442, 2015.

[38] Z. H. Deng, K. H. Luo, and H. L. Yu, "A study of supervised term weighting scheme for sentiment analysis," *Expert Systems with Applications*, vol. 41, no. 7, pp. 3506-3513, 2014.

[39] K. W. Church and P. Hanks, "Word association norms, mutual information and lexicography," in *Proceedings of the 27th Annual Meeting of the Association for Computational Linguistics*, Vancouver, Canada, 1989, pp. 76-83.

[40] A. K. Uysal and S. Gunal, "A novel probabilistic feature selection method for text classification," *Knowledge-Based Systems*, vol. 36, pp. 226-235, 2012.

[41] D. Mladeni'c and M. Grobelnik, "Feature selection for classification based on text hierarchy," in *Proceeding of the Conference on Automated Learning and Discovery (CONALD)*, Pittsburgh, PA, 1998.

[42] G. Forman, "An extensive empirical study of feature selection metrics for text classification," *Journal of Machine Learning Research*, vol. 3, pp. 1289-1305, 2003.

[43] V. Ng, S. Dasgupta, & S. M. Niaz Arifin, "Examining the role of linguistic knowledge sources in the automatic identification and classification of reviews," in *Proceedings of the COLING/ACL Main Conference Poster Sessions*, Sydney, Australia, 2006, pp. 611-618.

[44] M. A. Fattah, "The use of MSVM and HMM for sentence alignment," *Journal of Information Processing Systems*, vol. 8, no. 2, pp. 301-314, 2012.

[45] M. Elmarhoumy, M. A. Fattah, M. Suzuki, and F. Ren, "A new modified centroid classifier approach for automatic text classification," *IEEJ Transactions on Electrical and Electronic Engineering*, vol. 8, no. 4, pp. 364–370, 2013.

Mohamed Abdel Fattah

He received the B.Sc. and M.Sc. degrees in Electronics from the Faculty of Engineering, Cairo University, Cairo, Egypt, in 1994 and 2003, respectively, and the Ph.D. degree in information science and intelligent systems from the University of Tokushima, Japan, in 2007. He was awarded a Japan Society of the Promotion of Science (JSPS) postdoctoral fellowship from 2007 to 2009 in Department of Information Science and Intelligent Systems, Tokushima University. He is currently an Associate Professor with FIE, Helwan University, Cairo. His research interests include information retrieval, natural language processing, speech recognition and document processing.