Journal of Computing and Security

October 2014, Volume 1, Number 4 (pp. 261–272)

http://www.jcomsec.org

PSA: A Hybrid Feature Selection Approach for Persian Text Classification

Ayoub Bagheri [a,*]
Mohamad Saraee [a]
Shiva Nadi [b]

[a] *Isfahan University of Technology, Isfahan, Iran.*
[b] *Islamic Azad University, Najafabad Branch, Isfahan, Iran.*

ARTICLE INFO.

Article history:
Received: 26 August 2014
Revised: 01 Sepetember 2014
Accepted: 22 September 2014
Published Online: 27 February 2015

Keywords:
Text Classification, Text Mining, Feature Selection, Simulated Annealing Algorithm, Persian Language

ABSTRACT

In recent decades, as enormous amount of data being accumulated, the number of text documents is increasing vastly. E-mails, web pages, texts, news and articles are only part of this grow. Thus the need for text mining techniques, including automatic text classification, is rising. In automatic text classification, feature selection from within any text appears to be the most important step. Since the feature space in textual data includes tens of thousands of words, feature selection is used for dimension reduction. Different techniques, from statistical to machine learning approaches for feature selection in text have been reported in literature, each with advantages and disadvantages. However up to now there have been very rare researches on utilizing advantages of both learning and statistical approaches. In this paper a new algorithm for feature selection in text is presented to improve the classification performance substantially. The proposed approach - PSA - is based on simulated annealing algorithm and document frequency method. So it can benefit from advantages of both statistical and learning techniques. The simulated annealing algorithm requires an appropriate function for fitness evaluation, where document frequency method as an evaluation function has low computational cost. In addition, a new Persian text dataset, i.e. *Persian 7-NewsGroups Dataset*, is introduced for evaluating the proposed approach. Therefore, to justify and evaluate our approach, the performance of the PSA is compared to famous methods such as chi-square and correlation coefficient on Persian 7-NewsGroups dataset. The results show that the PSA has overall better performance in comparison to the other methods.

1 Introduction

Approximately over 90 percent of today's knowledge is in texts, documents and other media such as audios, images and videos [1], [2]. However, with the rapid growth of the Internet, it is natural that texts are not

* Corresponding author.

Email addresses: a.bagheri@ec.iut.ac.ir (A. Bagheri), m.saraee@salford.ac.uk (M. Saraee), nadi.shiva@gmail.com (Sh. Nadi)

ISSN: 2322-4460 © 2014 JComSec. All rights reserved.

paper based any longer but mostly in electronic format. Analyzing and mining text is becoming essential tool for success, as most organizations including academies, industries, health sector are generating huge amount of texts. Text mining is the process of extraction of implicit, previously unknown, and potentially useful information from large amount of textual data, or an exploration and analysis of textual data by automatic and semi-automatic tools to discover new knowledge.

Text classification and text clustering are the most popular techniques in text mining [1]. In addition, many new functions have become main stream in text mining, including email Spam detection and web page filtering [1]. Text classification has several steps including Text pre-processing, Text feature selection, and learning algorithm for classification, Testing and evaluating the algorithm on text datasets.

Text feature selection is one of the most important steps in text classification [1]. Feature selection is about finding useful and important features from text. In text classification applications, features are words from texts which if selected correctly they can classify texts with high precisions. Up to now, in the fields of text classification and text feature selection, many approaches have been introduced [3–20].

With the rapid growth of Persian web pages and electronic textual data, automated techniques are needed. To overcome part of this issue, in this paper we focus on Persian text feature selection in a text classification system. We propose a Simulated Annealing based feature selection approach PSA, which works with Persian datasets. PSA is a combination of Simulated Annealing (SA) and Document Frequency (DF) approaches. SA is a random search method for optimization problems and DF is a simple and efficient method for feature selection. The idea behind the PSA is to benefit from the advantages of both methods.

The remainder of the paper is organized as follows. In the next section we present the feature selection problem. In Section 2 we review two related works for feature selection. In Section 3, we discuss main characteristics of the proposed system for text feature selection and the PSA. Section 4 presents structure of a text classification system and evaluation. Moreover, an experimental study based on some of evaluation measures is presented in this Section. Finally Section 5 concludes by summarizing the work.

2 Background and Related Works

In recent years, data mining and information extraction methods have extended rapidly, therefore feature selection has become a demanded challenge [7], [11]. Feature selection is one the most important steps in text classification. If we use all of the words in a text as features then the feature space will be enormous. Usually there are between $10,000$ to $100,000$ different words in each dataset of textual data. Many of these words are not suitable for classification. In other words, among these features, some of them have no productive value for the performance of the classification and may also reduce the accuracy of the classification. Limiting the set of words which are used for classification will increase efficiency and reduce overall error [3], [8], [19].

Up to now, a number of methods have been reported for reducing the size of the feature space. Some of them are, information gain, correlation coefficient, mutual information, chi-square, simplified-chi-square and document frequency [8], [19]. In this paper we present a new algorithm for text feature selection problem based on a heuristic local search approach, simulated annealing, combined with document frequency method.

Feature selection should be performed on a per category basis to compute relevance between classes [21]. That is, words that may be irrelevant to one class may be relevant and important with respect to another. Since many classifiers are composed of binary classifiers for each category, it seems that for highest performance, feature selection should be performed for each category.

Many researchers reported that correlation coefficient and chi-square methods performed best in their multi-class benchmarks [19], [22]. Therefore we compare our method with these two. In this section we discuss document frequency, correlation coefficient, chi-square and simulated annealing approaches.

2.1 Chi-Square Measure

The first information measure is Chi-square (CHI or χ^2). First we introduce the feasibility table as Table 1. The feasibility table records co-occurrence statistics for terms (words, features) and classes (categories). With this table, for example we can see that the number of times a category c has occurred without the presence of term t in the training dataset was C. Also the number of documents N is $N = A+B+C+D$. These statistics are very useful for estimating probability values.

The formula for CHI is:

$$\chi^2(t,c) = \frac{N*(AD-CB)^2}{(A+B)*(C+D)*(A+C)*(B+D)} \tag{1}$$

Table 1. The feasibility table

	c	c'
t	A	B
t'	C	D

2.2 Correlation Coefficient

The complexity of some information measures does not always allow to readily interpret why their performances are so good [2]. In this respect, many researchers have observed that the use of $\chi^2(t)$ for feature selection is against intuition. The reason is that the power of 2, appeared in its formula, has the effect of equating those factors that indicate a positive correlation between the term (word) and the category (i.e. $P(t, c_i)$ and $P(t, c_i)$) with those that indicate a negative correlation (i.e. $P(t, c_i)$ and $P(t, c_i)$) [2]. The Correlation Coefficient (CC), is the square root of $\chi^2(t)$ and thus emphasizes the former and deemphasizes the latter. Therefore the formula for CHI is given as:

$$CC(t,c) = \frac{\sqrt{N} * A[A * D - C * D]}{\sqrt{(A+B)*(C+D)*(A+C)*(B+D)}} \qquad (2)$$

2.3 Document Frequency Method (DF)

Document frequency is a statistical method which is used in various applications in Information Retrieval and other related fields. Document frequency is the number of documents in which a term or word occurs in a dataset. It is the simplest criterion for feature selection and easily scales to a large dataset with linear computational complexity [19].

2.4 Simulated Annealing Algorithm

Definition of the simulated annealing algorithm is common with many researchers as: “a random search technique which exploits an analogy between the way in which a metal cools and freezes into a minimum energy crystalline structure (the annealing process) and the search for a minimum in a more general system; it forms the basis of an optimization technique for combinatorial and other problems” [23–27].

The idea of SA was first introduced by Metropolis [27]. The process which includes heating a solid past melting point and then cooling it, is known as annealing. The SA algorithm simulates the cooling process by decreasing the temperature of the system slowly until it converges to a steady state. Application of this idea to optimization problems was initiated by Kirkpatrick et al [26].

The way that SA approximates the global maximization problem resembles the use of a bouncing ball that can bounce from mountain to mountain. It begins at a high “temperature” which gives the ball a very high bounce. This high bounce enables it to bounce over any mountain and access any valley. As the temperature decreases the ball cannot bounce too high and it will become trapped in relatively small ranges of valleys and mountains. The acceptance distribution determines probabilistically whether to stay in a new lower valley or to bounce out of it [26].

SA can find the global optimum by carefully controlling the changing rate of the temperature. The law of thermodynamics state that at temperature t, the probability of an increase in energy of magnitude, ΔE is given by

$$P(\Delta E) = exp(\frac{-\Delta E}{Kt}) \qquad (3)$$

Where K is a constant known as Boltzmann’s constant [26], [27].

This formula calculates the probability of the change of the system’s energy. If the energy is decreased then the system chooses to move to the new state, otherwise moving to the new state is accepted using the probability returned by the Equation (3) [26], [27]. Using this probability is the basic idea of SA. Unlike hill climbing approaches, the use of this probability will prevent the choice of bad states in solving the problem. SA uses this idea by iterating the procedure and lowering the temperature until the system reaches a steady state [26], [27]. Therefore, the probability of moving to a bad state in SA is given by the following equation:

$$P(P_{m_i} \rightarrow P_{m_{i+1}}) = exp(\frac{-\Delta E}{t}) > r \qquad (4)$$

In this conditional formula, t is the current temperature and r is a random number between 0 and 1 which can be assigned by experiments. It is remarkable that in SA with decreasing the temperature of the system the probability of moving to bad states is decreased and only better moves are accepted.

3 Proposed Model For Text Feature Selection

Text preprocessing can generate more than $10,000$ unique terms, words or phrases as features. Removing less or not informative terms and also irrelevant terms decreases the computational cost and often makes

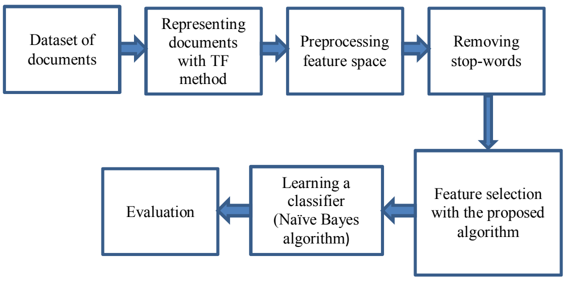

Dataset of documents
Representing documents with TF method
Preprocessing feature space
Removing stop-words
Feature selection with the proposed algorithm
Learning a classifier (Naïve Bayes algorithm)
Evaluation

Figure 1. Steps in the proposed model for text classification using feature selection

better classifiers. Feature selection process works by ranking all the terms and then selecting a subset containing best features. For this purpose, the first step is to select a method for representing the text documents.

The baseline method for representing a text document is to compute the weight of a term in a document. The simplest method for computing the weight of a term in a document is to count the number of times the term occurs in the document. In other words, in this model, a text is represented as a vector whose components are the frequencies of words. This method is usually called Term Frequency (TF) and is defined by the function

$$W_i = tf_i \tag{5}$$

where tf_i denotes the number of times the term i occurs in the corresponding document. After representing documents using term frequency method, preprocessing and feature selection steps are applied to datasets. Figure 1 shows steps of the proposed model for text classification including the core, feature selection.

3.1 Core of The Model - PSA

SA algorithm is one of the first heuristic methods and has a good strategy for avoiding local optimums. Its main idea is to allow for bad moves. Our proposed algorithm, PSA, is based on SA which we combined with DF method to reach a better performance (Algorithm 1). SA is a random search for optimization problems and DF is a simple and efficient method for feature selection. Therefore the idea behind the proposed algorithm is to benefit from the advantages of both methods and outputs with higher performances. PSA works on a feature vector of terms which is obtained from text corpora.

In order to use the proposed PSA algorithm, careful considerations should be given on the followings:

- Starting Step
- Stopping Criteria
- Temperature Decrement
- Cost Function (Fitness Function)
- Neighborhood Structure

Starting Step

The SA algorithm usually initializes with a random solution, and finds the optimal solution with a heating and cooling process. In the proposed PSA algorithm the initializing feature vector is acquired by all the terms in text documents. This vector, which the PSA algorithm wants to improve, is the solution for text classification problem. The proposed algorithm assumes the first vector is the solution vector and then this vector will be refined as the algorithm proceeds. In Algorithm 1 the current solution vector is named *offspring vector*.

Stopping Criteria

Two important matters in PSA are starting and final temperature. In PSA the starting temperature is shown by T_{max} and the final temperature is shown by T_{min}. Selecting the starting temperature in PSA is very important. It must be high enough to allow a move to almost any state of the problem, but this should be done carefully since a too high temperature can transform the search into a random search. For the stopping criteria the temperature can decrease until it reaches zero. But this can make the algorithm run for a lot longer. Therefore, the stopping criteria can be introduced as one of the followings [23]:

- A suitably low temperature
- When no better or bad moves are being accepted by the system
- When the cost function or performance is reached to a determined threshold

Temperature Decrement

Once we have our starting and stopping criteria we need to get from one to the other. Therefore it needs to decrement the temperature to arrive at the stopping criterion. The way in which PSA decrements the temperature is the key to the success of the algorithm. One way to decrement the temperature is a simple linear method:

$$\mathit{Temperature} *= \quad \mathit{CoolingRate} \tag{6}$$

Where the parameter *CoolingRate* is the rate of decrementing temperature. The experiments have shown that *CoolingRate* should be between 0.8 and 0.99, with better results being found in the higher end of the

Algorithm 1 The proposed PSA algorithm for text feature selection

```
Simulated_Annealing(problem, $T_{max}$, coolingRate)
initVector = Initializing Feature Vector which contains all words in text documents;
// bestVector: This is output, Final Feature Vector which contains the best and selected features
bestVector = initVector;
while !StopCondition() do
        Temperature = $T_{max}$;
        while Temperature > $T_{min}$do
                offspringVector = initVector; // generate an offspring
                for i ← 0 to initVector.Count do
                        if (rand0to1) < (initVector.Count/DF [i]*alpha && DF[i] < Average_DF) then
                                Mutate(offspringVector, i);
                        end if
                end for
                Evaluate(offspringVector);
                if Fitness(offspringVector) >= Fitness(initVector) then
                        initVector = offspringVector;
                else if rand0to1 () < p(Temperature, offspringVector, initVector) then
                        initVector = offspringVector;
                end if
                Temperature ∗ = coolingRate;
        end while
bestVector = Best(offspringVector, bestVector);
end while
End
```

range. Of course, the higher the value of *CoolingRate*, the longer it will take to decrement the temperature to the stopping criterion. Theory states that the PSA should allow enough iterations at each temperature so that the system stabilizes at that temperature.

Cost Function

Cost function or fitness function in PSA calculates cost of the solution in each of the iterations. In defining cost function it is important that the function can be computed as efficiently as possible. In the PSA algorithm the cost function is called with:

Evaluate (offspringVector);

Where the cost function of the PSA is based on document frequency method. Output of the feature selection process is the input for classification. Hence the output of proposed algorithm is a vector of features. The cost function for the PSA algorithm is shown in Algorithm 2.

As we can see from Algorithm 2, the cost function is based on DF method. For a solution vector, the cost can be computed by the average of the DF value of every feature. Besides increasing the efficiency, by using this cost function in the PSA algorithm the optimum solution can be reached with better time complexity than traditional methods.

Algorithm 2 Cost function for the PSA algorithm

```
evaluateFitness(offspringVector)
        computeDFforSA(offspringVector);
        sum = 0;
        for i ← 0 to offspringVector.Count do
                sum += DF[i];
        end for
        Fitness = sum / offspringVector.Count;
        return Fitness;
```

Neighborhood Structure

In text feature selection, the neighborhood function could be defined as changing, adding or removing the features of feature vector. The PSA algorithm creates a new solution vector as the neighborhood solution in each iteration. After that, by evaluating the new solution, if the new feature vector is a better solution than the current solution, the algorithm will accept it, and if it is not it will be accepted by PSA with a low probability related to temperature and cost. The procedure for creating a new neighborhood solution in our algorithm will be called as below:

Mutate (offspringVector, i);

It is worth mentioning that for creating a new neighborhood vector the probability of removing each feature has to be considered. In each of the iterations the new feature vector can be the previous feature vector with some little changes or it can be obtained with many changes and many feature removals. In this procedure *offspringVector* expression is the feature vector that we are going to check and i is the feature which the algorithm checks for removing. From Fig.2, it can be seen that in PSA, a feature will be removed from feature vector when the following condition is met:

$$(Rand < \frac{initVector.Count}{DF[i]} * alpha \qquad \&\& \qquad DF[i] < Average_D F)$$

In the above condition, the parameter $Rand$ is a random value between zero and one. Expression $initVector.Count$ shows number of features in current feature vector. $DF[i]$ is the value of document frequency measure for feature i, and $Average_D F$ shows the average of document frequency values. In this condition, beside consideration of probability computations in SA, the DF measure is also checked. In addition, there is a variable named $alpha.alpha$ is a variable which controls the condition with respect to the number of features and the values of DF measure. $alpha$ is selected by experiments.
After creating and evaluating a new feature vector, the PSA algorithm replaces it with previous feature vector when the new one has a better fitness. When the cost value of the new solution is worse than the previous one, the new solution will be replaced with the previous one with a probability condition as shown in:

$$Rand < e^{\frac{-delta}{temperature}}$$

Where $delta$ is the difference of the costs of two solution vectors. $temperature$ shows the temperature of the current iteration and $Rand$ is a random value between zero and one.

4 Experiments

In this section we discuss the experimental results for the proposed PSA algorithm and the previous algorithms in a text classification model.

4.1 Evaluation Measures

The evaluation of a text classification model is based on test samples that have been already labeled by human experts. Therefore to compare the matches between human assigned classes and classifier assigned ones, we can summarize four possible situations in the following contingency table:

Table 2. Contingency table

Class C_i	Assigned by human expert		
Assigned by classifier		YES	NO
	YES	TP_i	FP_i
	NO	FN_i	TN_i

Where

- TP_i (True Positive): Those assignments where the classifier and human expert agree for a label, in other words those assignments that the classifier labeled correctly as positive (belonging to class C_i).
- FP_i (False Positive): Those assignments where the classifier and human expert do not agree for a label, in other words those assignments that the classifier labeled incorrectly as positive (belonging to class C_i).
- FN_i (False Negative): Those assignments where the classifier and human expert do not agree for a label, in other words those assignments that the classifier labeled incorrectly as negative (not belonging to class C_i).
- TN_i (True Negative): Those assignments where the classifier and human expert agree, in other words those assignments that the classifier labeled correctly as negative (not belonging to class C_i).

By combining these values some well-known measures can be computed:

$$precision(P) = \frac{TP}{TP + FP} \tag{7}$$

$$recall(R) = \frac{TP}{TP + FN} \tag{8}$$

$$F1 = \frac{2PR}{P + R} \tag{9}$$

Precision shows how well the labels are assigned by text classification model and how many of labels assigned correctly as to the corresponding class label. Recall computes the fraction of expert labels found by the model. These two measures are well known in information retrieval systems, but the balance between these two values is a difficult task, since usually the improvement in one leads to reduction in the other. The classifier can achieve a trade-off between precision and recall by adjusting the decision boundary between the positive and negative classes. Since we are looking for systems showing both high precision and high recall, we have to select a measure which shows the results in a better way. The most used measure for this matter in a text classification model is $F1$ measure. This measure is a trade-off between precision and recall [2], [11, 12], [22]. In our experiments, the $F1$

measure plays a central role for evaluating the model.

4.2 Global Measures

Precision, recall and $F1$ measures are computed for each class, therefore to evaluate the performance across all classes, these measures have to be averaged. There are two kinds of averaged values, Micro and Macro averaging. Micro-averaging is obtained by first computing the precisions and the recalls for all the classes and then using them to compute the measures [2], [11, 12], [22]. Macro-averaging is calculated by first computing the measures for all the classes and then taking their average. Micro-averaging tends to emphasize the large-sized classes, but Macro-averaging by small-sized ones. In other words, Macro-averaging measure gives the classes the same importance, but Micro-averaging measure gives more priority to classes with more documents. Considering the contingency table, Table 2, we would have Micro-averaging Formulas for the measures as:

$$precision^m = \frac{\sum_i \sum_j TP_{ij}}{\sum_i \sum_j TP_{ij} + \sum_i \sum_j FP_{ij}} \tag{10}$$

$$recall^m = \frac{\sum_i \sum_j TP_{ij}}{\sum_i \sum_j TP_{ij} + \sum_i \sum_j FN_{ij}} \tag{11}$$

$$F1^m = \frac{2 * precision^m * recall^m}{precision_m + recall_m} \tag{12}$$

These measures work for each document. TP_{ij}, FP_{ij} and FN_{ij} are the number of true positives, false positives and false negatives respectively, found for class i in the evaluation of document j. Like Micro-averaging we have Macro-averaging equations as:

$$precision_j = \frac{\sum_i TP_{ij}}{\sum_i FP_{ij}} \tag{13}$$

$$recall_j = \frac{\sum_i TP_{ij}}{\sum_i FN_{ij}} \tag{14}$$

$$precision^M = \frac{\sum_j precision_j}{n} \tag{15}$$

$$recall^M = \frac{\sum_j recall_j}{n} \tag{16}$$

$$F1^M = \frac{2 * \sum_j precision_j * \sum_j recall_j}{n * (\sum_j precision_j + \sum_j recall_j)} \tag{17}$$

Where n is the total number of documents in textual datasets.

4.3 Cross Validation

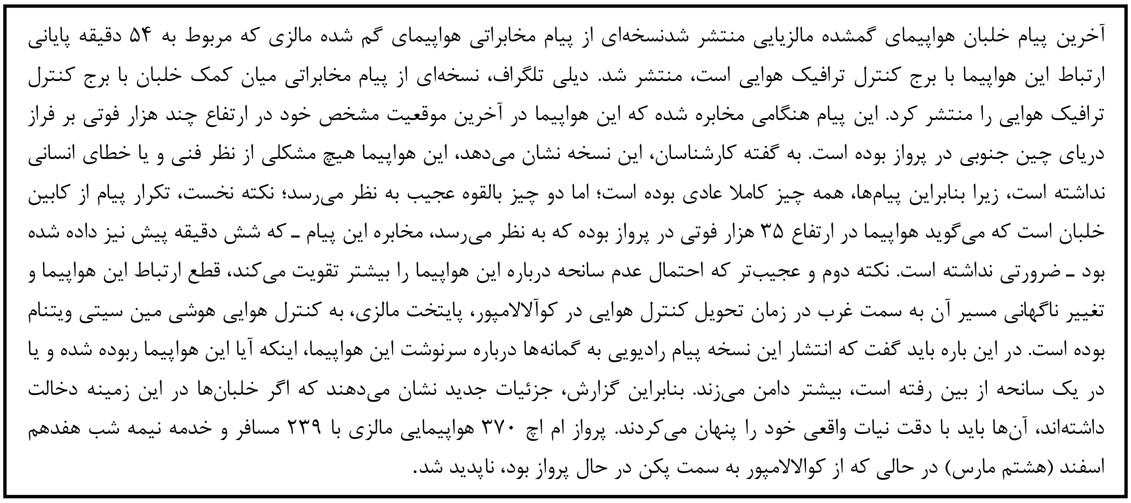
A supervised learning algorithm needs some of the data to be labeled as training data and some of them as test data [2]. These two datasets must be separate to prevent false results in evaluating the performances of the methods. Therefore multiple runs of the experiments are usually needed, with different datasets (train and test) at each run. For this purpose data must be split into separate datasets for training and testing. One of the approaches in splitting the dataset and running the experiment is cross validation [2]. N-fold cross validation consists of splitting the dataset into N subsets of equal size. At each turn, one set is used for testing and the rest for training the system. In our case, 5-fold cross validation is used. At each turn, 4 folds will be used for training and one for testing, in such a way that every subset will be used once for testing purposes. Then, the average over all 5 experiments will be an estimate of the performance of the classifier.

4.4 Data Description

Feature selection approaches are not for a specific language and can be used for every language. Because of very rare works on Persian language, in this paper we tested the feature selection methods on a Persian text dataset. Persian language (also named Farsi) is the formal language of some countries like Iran, Tajikistan and Afghanistan. Persian is the second language of some countries in Middle East too. Persian language has its own structure and complexity. Because of few works on Persian text there is no large dataset in this area. In this research, we used a dataset named Persian 7-NewsGroups. This dataset belongs to Intelligent Databases, Data Mining and Bioinformatics research lab, faculty of electrical and computer engineering, Isfahan university of technology, of Iran. Table 3 shows the description of the Persian 7-NewsGroups dataset. Also Figure 2 exhibits a sample Persian text document.

Table 3. Description of Persian 7-NewsGroups dataset

Class(Category)	Number of Documents
Social	804
Economic	806
Politic	819
Scientific	802
Cultural	803
International	802
Sport	802



آخرین پیام خلبان هواپیمای گمشده مالزیایی منتشر شدنسخه‌ای از پیام مخابراتی هواپیمای گم شده مالزی که مربوط به ۵۴ دقیقه پایانی
ارتباط این هواپیما با برج کنترل ترافیک هوایی است، منتشر شد. دیلی تلگراف، نسخه‌ای از پیام مخابراتی میان کمک خلبان با برج کنترل
ترافیک هوایی را منتشر کرد. این پیام هنگامی مخابره شده که این هواپیما در آخرین موقعیت مشخص خود در ارتفاع چند هزار فوتی بر فراز
دریای چین جنوبی در پرواز بوده است. به گفته کارشناسان، این نسخه نشان می‌دهد، این هواپیما هیچ مشکلی از نظر فنی و یا خطای انسانی
نداشته است، زیرا بنابراین پیام‌ها، همه چیز کاملا عادی بوده است؛ اما دو چیز بالقوه عجیب به نظر می‌رسد؛ نکته نخست، تکرار پیام از کابین
خلبان است که می‌گوید هواپیما در ارتفاع ۳۵ هزار فوتی در پرواز بوده که به نظر می‌رسد، مخابره این پیام ـ که شش دقیقه پیش نیز داده شده
بود ـ ضرورتی نداشته است. نکته دوم و عجیب‌تر که احتمال عدم سانحه درباره این هواپیما را بیشتر تقویت می‌کند، قطع ارتباط این هواپیما و
تغییر ناگهانی مسیر آن به سمت غرب در زمان تحویل کنترل هوایی در کوآلالامپور، پایتخت مالزی، به کنترل هوایی هوشی مین سیتی ویتنام
بوده است. در این باره باید گفت که انتشار این نسخه پیام رادیویی به گمانه‌ها درباره سرنوشت این هواپیما، اینکه آیا این هواپیما ربوده شده و یا
در یک سانحه از بین رفته است، بیشتر دامن می‌زند. بنابراین گزارش، جزئیات جدید نشان می‌دهند که اگر خلبان‌ها در این زمینه دخالت
داشته‌اند، آن‌ها باید با دقت نیات واقعی خود را پنهان می‌کردند. پرواز ام اچ ۳۷۰ هواپیمایی مالزی با ۲۳۹ مسافر و خدمه نیمه شب هفدهم
اسفند (هشتم مارس) در حالی که از کوالالامپور به سمت پکن در حال پرواز بود، ناپدید شد.

Figure 2. Sample Persian text document

4.5 Implementation and Results

Feature selection is the process of selecting a subset of features in textual data based on a measurement factor. This process removes dimensionality of the input data and it can raise efficiency of text classification process. As Figure 1 shows, the main steps of the text classification model are:

- Feature extraction and preprocessing
- Feature Selection
- Learning Classifier

In the following, these three steps with their sub steps are discussed. The first step of classification process is preprocessing the text documents.

Preprocessing

This step contains three phases:

(1) Extracting features and terms
(2) Eliminating Stop-words
(3) Removing low frequency features

In the first phase, we used a set of delimiters like space to find the terms in textual datasets. Among those terms and words, some words are too frequent to be a helpful feature. For example, the words “are” and “in” can be seen in every English text documents and just like them the words به , با and که are as frequent in Persian text documents. Such words are called stop-words and often removed from the feature space.

It needs to mention that in stop-word elimination, we need to have some experience and information on structure of Persian language. In our work, the stop-words are listed in a text document by a linguistic expert. As we mentioned before, words like and are stop-words in Persian and have no value in classification. One of the benefits of elimination stop-words is decreasing the time for learning process.

Phase three is for terms with low frequency in text documents, for example terms with frequencies below four have no benefit for classification. These terms are often noises that writers produce by mistake. One example in Persian is شسیبقل , which is a typo and has no meaning.

Feature Selection

The second step of implementing a text classification system is feature selection. In this paper, we selected two feature selection approaches to compare with our proposed PSA algorithm. As we mentioned before, these methods are:

- Chi-square method or χ^2 which we refer to it as CHI in figures and tables.
- Correlation coefficient which we refer to it as CC in figures and tables.

Learning an Algorithm as a Classifier

In this section we present Naïve Bayes algorithm as the text classifier for our model. Naïve Bayes algorithm is animportant algorithm in text classification, because it has high speed and is easy to implement. This algorithm is a known and popular algorithm in text classification problems. The Naïve Bayes algorithm is traditionally trained using a collection of labeled documents [3], [28].

We used the MAP (maximum a posteriori) Naïve

Bayes algorithm in our experiments as a classifier for Persian text classification model [28]. As it is mentioned before we used a feature vector model to represent the text documents. As we know in text classification problem, training and testing datasets have to be labeled by a human expert and the classifier predicts the class of each text document in test dataset. Naïve Bayes algorithm assigns a new document to a class with the maximum probability. This maximum value can be calculated by:

$$NaiveBayesClassifier : \nu_{NB} = argmax_{\nu_j \in V} P(\nu_j) \prod_i P(a_i|\nu_j) \tag{18}$$

Where ν_{NB} is the assigned class or output of Naïve Bayes algorithm, ν_j shows the class j^{th}, $P(\nu_j)$ is the prior probability of class j in the set of all classes V and $P(a_i \nu_j)$ shows conditional probability of feature i in class j. Output of the Naïve Bayes algorithm is the class with maximum probability among all classes. To calculate ν_{NB}, we require estimates for the probability terms $P(\nu_j)$ and $P(a_i \nu_j)$. The former parameter can simply be estimated based on the fraction of each class in the training data:

$$P(\nu_j) = \frac{number\ of\ text\ documents\ with\ class\ label\ j}{total\ number\ of\ text\ documents} \tag{19}$$

and $P(a_i|\nu_j)$ can be computed by:

$$P(w_k|\nu_j) = \frac{n_k + 1}{n + |Vocabulary|} \tag{20}$$

Where n is the total number of words in all training data whose class value is ν_j, n_k is the number of times word w_k is found among these n words and $Vocabulary$ is the set of all separate words in any text document in the dataset.

Since $P(\nu_j)$ maybe zero in some circumstances, we use a modified version of that equation:

$$P(\nu_j) = \frac{1 + numberoftextdocumentswithclasslabelj}{totalnumberoftextdocuments + |V|} \tag{21}$$

After training the classifier, we can use the following equation for estimating the class label of a new document:

$$\nu_{NB} = argmax_{\nu_j \in V} P(\nu_j) \prod_{i \in words} P(a_i|\nu_j) \tag{22}$$

In this formula, *words* is the set of all words in the new document, and Naïve Bayes algorithm selects a class which has the maximum probability among all class labels [28].

Experimental Results and Comparing Methods

To compare the feature selection methods, we train text classification model with obtained feature vectors and then assess performance of the model in each method by testing the classifier. In our experiments, we use the following settings for the proposed PSA algorithm:

- Starting temperature: 100
- Final temperature: zero
- - Temperature decrement: we use the $CoolingRate$ parameter set to 0.999 in the following equation: $Temperature * = CoolingRate$
- PSA iterates in each temperature once.
- Fitness function is based on the DF method.
- Neighborhood structure: we use eliminating features strategy.

Based on Persian 7-NewsGroups, the number of features in all text documents which are obtained by feature extraction is 26050. After the preprocessing step, 7794 features remain from all text documents. After using the feature selection methods, only between 10 to 30 percent of the features remained for training NB learning algorithm.

Figure 3 shows the Micro-averaging precision measure for the feature selection methods combined with the Naïve Bayes classifier. The horizontal axis exhibits the percentage of removed features and the vertical axis shows the value of the measure. As we can see from this figure, PSA has better results in contrast to CHI and CC methods, while the best performance of PSA is 88%, the CHI 88% and the CC 87.7%.

From multiple runs of the methods we found that with 70% feature removal, the results of the methods reach a steady state and good performance. As we have shown in Figure 3, the proportion of used features are reduced from 30% to 20%. As this figure indicates, when 30% of the features are used, PSA has not a good result with respect to CHI and CC methods. This is because of the nature of the evolutionary probability of the PSA. comparing these methods based on Micro-averaging precision measure, we can see that

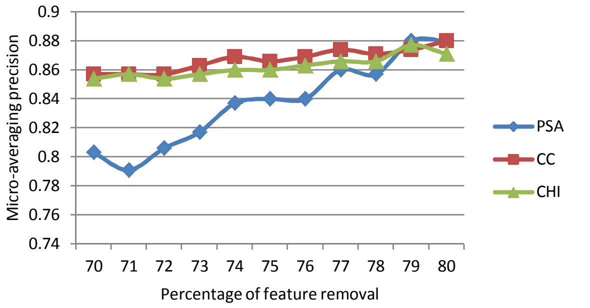


Figure 3. Comparing PSA algorithm with CHI and CC based on Micro-averaging precision

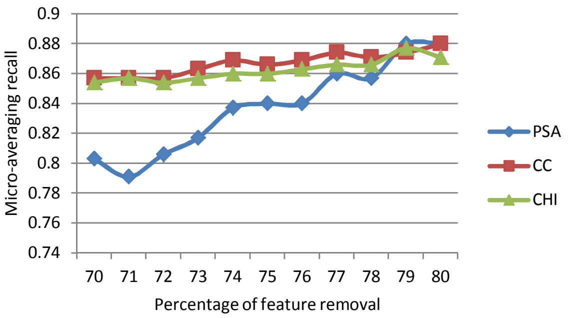


Figure 4. Comparing PSA algorithm with CHI and CC based on Micro-averaging recall

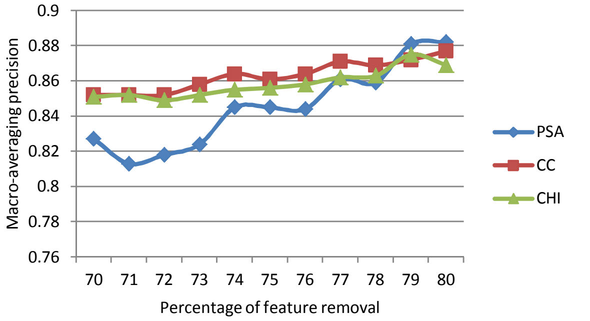


Figure 5. Comparing PSA algorithm with CHI and CC based on Macro-averaging precision

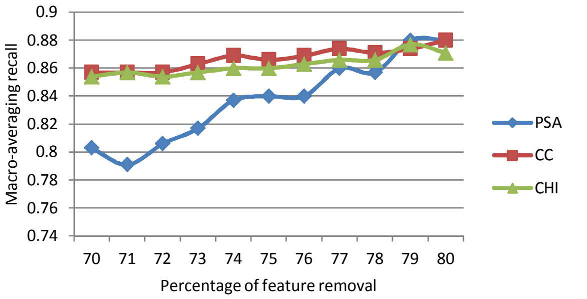


Figure 6. Comparing PSA algorithm with CHI and CC based on Macro-averaging recall

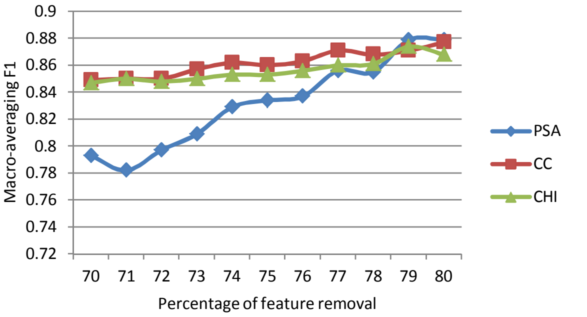


Figure 7. Comparing PSA algorithm with CHI and CC based on Micro-averaging F1

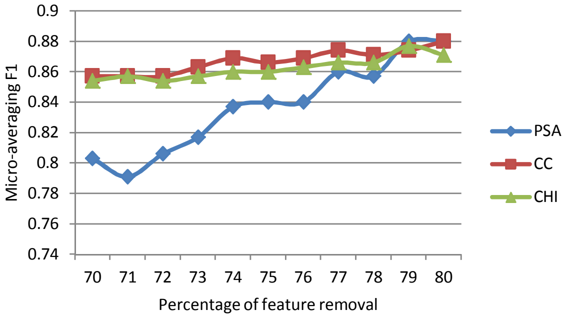


Figure 8. Comparing PSA algorithm with CHI and CC based on Macro-averaging F1

the minimum value is for PSA and is about 79.1%. This happens when 71% of features are removed. The maximum value is 88% for PSA where 79% of features eliminated. We can also see the Micro-averaging recall measure for the feature selection methods combined with Naïve Bayes classifier in Figure 4.

Figure 5 shows the Macro-averaging precision measure for three feature selection methods combined with Naïve Bayes classifier.

Figure 6 shows the Macro-averaging recall measure for the methods. When we tested the methods, the performance was stable after removing more than 70% of features. As can be seen from these figures the performance of PSA algorithm is quite good and comparable to the other methods. In the next figures, we can see three more measures for the feature selection methods which show the superiority of our proposed PSA algorithm.

Figure 7 and Figure 8 show Micro-averaging F1 and Macro-averaging F1 measures respectively for the methods.

Form Figure 7 and Figure 8, we found two points. The first point is the superiority of the proposed algorithm PSA over the CHI and CC methods. The second point is that the maximum performance happens when we reduce 80% of the features from the feature space.

To complete the analysis of the results, we compared each of the six evaluation measures for the PSA, CHI

Table 4. Comparing values of measures for PSA, CC and CHI with 80% feature removal

Measures	PSA	CC	CHI
Micro-averaging precision	0.88	0.88	0.871
Micro-averaging recall	0.88	0.88	0.871
Macro-averaging precision	0.882	0.877	0.869
Macro-averaging recall	0.88	0.88	0.871
Micro-averaging F1	0.88	0.88	0.871
Macro-averaging F1	0.879	0.877	0.868

Table 5. Results of Precision, Recall and F1 measures for PSA algorithm with 80% feature removal

Class	Precision	Recall	F1-Measure
Cultural	0.887	0.94	0.913
Economic	0.839	0.94	0.887
International	0.957	0.9	0.928
Politic	0.739	0.68	0.708
Scientific	0.925	0.98	0.951
Social	0.827	0.86	0.843
Sport	1	0.86	0.925

and CC in Table 4 when 80% of features are removed.

In addition, Table 5 shows the precision, recall and F1 measures for all class labels. This table is the result of PSA algorithm when 80% of features are eliminated.

As can be seen from Table 5, documents with *politic* class labels have minimum values among other classes. Therefore the proposed feature selection method, PSA, has weakness in text documents with *politic* category label. When we examined and analyzed the results we reached the conclusion that all the *politic* text with wrong labeling, were assigned to the social category.

As can be seen from Table 5 text documents with sport, scientific and international class labels have the best values in evaluation measures. The precision measure in sport documents is equal to one which means there is no document in the test dataset which is classified wrongly as sport category. The proposed PSA algorithm has the best performance in text documents from sport, scientific and international categories.

By theoretical and experimental analyses, it is proved that PSA algorithm combined with Naïve Bayes classifier results in higher classification performance, and the solution is practical and effective, in addition PSA outperforms CHI and CC methods.

5 Conclusion

Nowadays, text classification on real documents has become popular in many fields, such as natural language processing, information retrieval, artificial intelligence, and opinion mining. For text classification, feature selection is a key part with effective impact on performance. In this paper, a new algorithm named PSA for text feature selection is proposed. In this algorithm, we used a modified version of simulated annealing algorithm which we combined with document frequency method. The proposed PSA algorithm along with chi-square and correlation coefficient methods, have been tested on a Persian text dataset. With comparing the results, we found superiority of the proposed algorithm over chi-square and correlation coefficient methods.

References

[1] Atreya Basu, C Walters, and M Shepherd. Support vector machines for text categorization. In *System Sciences, 2003. Proceedings of the 36th Annual Hawaii International Conference on*, pages 7–pp. IEEE, 2003.

[2] Ronen Feldman and James Sanger. *The text mining handbook: advanced approaches in analyzing unstructured data*. Cambridge University Press, 2007.

[3] Jingnian Chen, Houkuan Huang, Shengfeng Tian, and Youli Qu. Feature selection for text classification with naïve bayes. *Expert Systems with Applications*, 36(3):5432–5435, 2009.

[4] Mohamad Saraee and Ayoub Bagheri. Feature selection methods in persian sentiment analysis. In *Natural Language Processing and Information Systems*, pages 303–308. Springer, 2013.

[5] Mehdi Hosseinzadeh Aghdam, Nasser Ghasem-Aghaee, and Mohammad Ehsan Basiri. Text feature selection using ant colony optimization. *Expert systems with applications*, 36(3):6843–6853, 2009.

[6] Bong Chih How and Kulathuramaiyer Narayanan. An empirical study of feature selection for text categorization based on term weightage. In *Proceedings of the 2004 IEEE/WIC/ACM International Conference on Web Intelligence*, pages 599–602. IEEE Computer Society, 2004.

[7] Huan Liu and Hiroshi Motoda. *Computational methods of feature selection*. CRC Press, 2007.

[8] Luying Liu, Jianchu Kang, Jing Yu, and Zhongliang Wang. A comparative study on un-

supervised feature selection methods for text clustering. In *Natural Language Processing and Knowledge Engineering, 2005. IEEE NLP-KE'05. Proceedings of 2005 IEEE International Conference on*, pages 597–601. IEEE, 2005.

[9] Thomas W. Miller. *Data And Text Mining: A Business Application Approach.* Prentice-Hall, Inc., 2004.

[10] Harun Uğuz. A two-stage feature selection method for text categorization by using information gain, principal component analysis and genetic algorithm. *Knowledge-Based Systems*, 24 (7):1024–1032, 2011.

[11] Rudy Prabowo and Mike Thelwall. A comparison of feature selection methods for an evolving rss feed corpus. *Information processing & management*, 42(6):1491–1512, 2006.

[12] Jason DM Rennie. *Improving multi-class text classification with naive Bayes.* PhD thesis, Massachusetts Institute of Technology, 2001.

[13] Fabrizio Sebastiani. Machine learning in automated text categorization. *ACM computing surveys (CSUR)*, 34(1):1–47, 2002.

[14] Ayoub Bagheri, Mohamad Saraee, and Franciska De Jong. Care more about customers: unsupervised domain-independent aspect detection for sentiment analysis of customer reviews. *Knowledge-Based Systems*, 52:201–213, 2013.

[15] Wenqian Shang, Houkuan Huang, Haibin Zhu, Yongmin Lin, Youli Qu, and Zhihai Wang. A novel feature selection algorithm for text categorization. *Expert Systems with Applications*, 33 (1):1–5, 2007.

[16] Stefano Baccianella, Andrea Esuli, and Fabrizio Sebastiani. Using micro-documents for feature selection: The case of ordinal text classification. *Expert Systems with Applications*, 40(11):4687–4696, 2013.

[17] Shiva Nadi, Mohammad Hossein Saraee, and Ayoub Bagheri. A hybrid recommender system for dynamic web users. *International Journal Multimedia and Image Processing (IJMIP)*, 1(1):3–8, 2011.

[18] Guozhong Feng, Jianhua Guo, Bing-Yi Jing, and Lizhu Hao. A bayesian feature selection paradigm for text classification. *Information Processing & Management*, 48(2):283–302, 2012.

[19] Yiming Yang and Jan O Pedersen. A comparative study on feature selection in text categorization. In *ICML*, volume 97, pages 412–420, 1997.

[20] Zhu Zhen-fang, Liu Pei-yu, and Lu Ran. Research of text classification technology based on genetic annealing algorithm. In *Computational Intelligence and Design, 2008. ISCID'08. International Symposium on*, volume 1, pages 265–269. IEEE, 2008.

[21] A. Bagheri, M. Saraee, and F. de Jong. Sentiment classification in persian: Introducing a mutual information-based method for feature selection. In *Electrical Engineering (ICEE), 2013 21st Iranian Conference on*, pages 1–6. IEEE, May 2013.

[22] Stefano Baccianella, Andrea Esuli, and Fabrizio Sebastiani. Feature selection for ordinal text classification. *Neural computation*, 26(3):557–591, 2014.

[23] Kathryn A Dowsland and Jonathan M Thompson. Simulated annealing. In *Handbook of Natural Computing*, pages 1623–1655. Springer, 2012.

[24] Ruslan Salakhutdinov and Geoffrey Hinton. An efficient learning procedure for deep boltzmann machines. *Neural computation*, 24(8):1967–2006, 2012.

[25] Yang-Lang Chang. A simulated annealing feature extraction approach for hyperspectral images. *Future Generation Computer Systems*, 27(4):419–426, 2011.

[26] Dimitris Bertsimas and Omid Nohadani. Robust optimization with simulated annealing. *Journal of Global Optimization*, 48(2):323–334, 2010.

[27] Reuven Y Rubinstein and Dirk P Kroese. *Simulation and the Monte Carlo method*, volume 707. John Wiley & Sons, 2011.

[28] T.M. Mitchell. Machine learning. 1997.

Ayoub Bagheri received his PhD in Computer Engineering from Isfahan University of Technology (IUT), Iran in 2014. He received his BSc in computer engineering in 2007 from Ferdowsi University of Mashhad, Iran, and the MSc in computer engineering in 2009 from IUT. From 2007 he became a member of Data Mining, Bioinformatics and Databases Laboratory at IUT. He also was a guest researcher at the Human Media Interaction group at the University of Twente, the Netherlands in 2012. His main research interests are in text mining, sentiment analysis and natural language processing.

Mohamad Saraee received his PhD from University of Manchester in Computation, MSc from University of Wyoming, USA in Software Engineering and BSc from Shahid Beheshti University, Iran. His main areas of research are intelligent databases, Mining advanced and complex data including medical and Bio, Text Mining and E-Commerce. He has published extensively in each of these areas and served on scientific and organizing committee on number of journals and conferences.

Shiva Nadi received her BSc and MSc degrees in computer engineering from Islamic Azad University of Najafabad, Iran, in 2008 and 2010 respectively. Her research interests are in several areas of artificial intelligence, data mining, web mining, evolutionary algorithms and software engineering.