World Applied Sciences Journal 7 (Special Issue of Computer & IT): 69-74, 2009
ISSN 1818-4952

Text Feature Selection using Particle Swarm Optimization Algorithm

[1]*Bilal M. Zahran and* [2]*Ghassan Kanaan*

[1]Faculty of Engineering Technology, Al-Balqa Applied University, Amman, Jordan
[2]Faculty of Information Systems and Technology, Arab Academy for Banking and Financial Sciences, Amman, Jordan

Abstract: Text Categorization (TC) has become recently an important technology in the field of organizing a huge number of documents. Feature Selection (FS) is commonly used to reduce dimensionality of text datasets with huge number of features which would be difficult to process further. In this paper we have implemented an efficient feature selection algorithm based on Particle Swarm Optimization (PSO) to improve the performance of Arabic text categorization. PSO is a search algorithm that employs a population of particles existing within a multi-dimensional space. We have used Radial Basis Function (RBF) networks as a text classifier. The performance of the proposed algorithm is compared to the performance of document frequency, tf×idf and Chi-square statistic algorithms. Simulation results on the Arabic dataset show the superiority of the proposed algorithm.

Key words: Text categorization . feature selection . particle swarm optimization . radial basis function networks . wrapper feature selection method

INTRODUCTION

TC has become recently an important technology in data mining field. A major problem of text categorization is the high dimensionality of the feature space. Most of these dimensions are not relative to TC which results in reducing the performance of the classifier.

FS is the process of selecting a subset of features available from the data for application of a learning algorithm. The best feature subset contains the least number of features that most contribute to accuracy and efficiency. This is an important stage of preprocessing and is one of two ways of avoiding the high dimensional space of features (the other is feature extraction).

FS is found to be an NP-hard and combinatorial problem. Hence, Evolutionary Algorithms (EAs) are generally more suitable to solve this difficult problem because they are population-based stochastic approaches that uses heuristic information. A PSO is modeled after the simulation of the social behavior of bird flocks [8]. PSO is easy to implement and has been successfully applied to solve a wide range of optimization problems. Thus, due to its simplicity and efficiency in navigating large search spaces for optimal solutions and its superiority of other EAs techniques [6, 7] PSO algorithm is used in this research to develop an efficient algorithm to optimize FS problem oriented to Arabic text classification field. Several EAs algorithms have been used for English text as an FS [1, 3, 15]. To the best of our knowledge, this is the first research on Arabic dataset which utilizes PSO algorithm as a feature selection.

The rest of this paper is organized as follows: section 2 introduces the subject of feature selection process. Then it presents a brief overview of FS approaches followed by definition of some FS methods. Section 3 presents brief discussion of a TC system. Then it outline the properties of Arabic text briefly. After that it presents a brief overview of RBF classifier. Section 4 presents a PSO algorithm, furthermore the proposed algorithm is outlined. Section 5 presents the experimental setup and a discussion of the results obtained by applying the proposed algorithm to Arabic TC. Section 6 highlights the conclusions of this paper.

FEATURE SELECTION

An FS algorithm selects a subset of important features and removes irrelevant, redundant and noisy features for simpler and more accurate data representation. As a result, saving in the computational resources, storage and memory requirements could be achieved.

Corresponding Author: Dr. Bilal M. Zahran, Computer Engineeering Dept., Faculty of Engineering Technology, P.O. Box 15008 Amman 11134, Jordan Cell: 00962-777169352

The FS process is commonly performed by two approaches namely the filter approach and the wrapper approach [1]. While the filter approach is based on applying a scoring method to evaluate the feature, the wrapper approach wraps the features around the classifier to be used to anticipate the benefits of adding or removing a certain feature from the training set.

Commonly, forward selection or backward elimination or randomized feature selction is used. In the forward selection approach, a wrapper examines the effect of adding each unselected feature and chooses the one that leads to the best accuracy. feature that cause the performance to the classifier to degrade are removed in the backward elimination approach. In the randomized FS, the features are selected randomly and are evaluated by a fitness function built on heuristic information. Most studies showed that the wrapper methods are more efficient than the filter methods in terms of classification efficiency [9]. However, the wrapper methods are very computationally expensive, as they involve calling the induction algorithm for each feature set considered [5]. We have implemented a wrapper method in this research.

Important FS filtering methods

Chi-Square Statistic (CHI): Chi-Square statistic is the common statistical test that measures divergence from the distribution expected if one assumes the feature occurrence is actually independent of the class value.

The CHI measure is defined as follows [16]:

$$CH1\ (t, c) = (N\times(AD-CB)^2)/((A+C)\times (B+D)\times(A+B)\times(C+D)) \quad (1)$$

Where A is the number of times t and c co-occur. B is the number of time the t occurs without c. C is the number of times c occurs without t. D is the number of times neither c nor t occurs. N is the total number of documents.

Document Frequency (DF): Document Frequency simply measures in how many documents the word appears. Selecting frequent words will improve the chances that the features will be present in future test cases. It performed much better than Mutual Information in the study by Yang and Pedersen [16], but was consistently dominated by information gain (IG) and CHI (which, they point out, each have a significant correlation with frequent terms).

TEXT CATEGORIZATION

Text Categorization (classification) (TC)is the process of classifying documents into a predefined set of categories based on their content.

Arabic language structure: Arabic is the mother language of millions of people all over the world. It is a highly inflected language, it has much richer morphology than English [14].

Some challenges of Arabic language in TC tasks: Among several sources that results in The difficulty of Arabic TC are the following [10]:

- Arabic language differs syntactically, morphologically and semantically from other Indo-European languages.
- Compared to English, Arabic language is more sparsed, which means that English words repeated more often than Arabic words for the same text length.
- In written Arabic, most letters take many forms of writing. Moreover, there is a punctuation associated with some letters that may change the meaning of two identical words.
- The omission of diacritics (vowels) in written Arabic “altashkiil”.
- Comparing to English roots, Arabic roots are more complex.

As a conclusion, the research in the field of Arabic natural language processing is still immature. So special care in document preprocessing should be taken before carrying out TC tasks.

The most popular classifiers used in TC are [4]: Decision Tree (DT), Decision rules, Maximum Entropy (ME), Neural network (NNet), Naïve Bayes (NB), K-Nearst Neighbors (kNN) and Support Vector Machine (SVM).

We have used the radial basis function (RBF) networks as a text classifier.

Radial basis function networks classifier: Radial basis function (RBF) networks are feed-forward networks trained using a supervised training algorithm. They are typically configured with a single hidden layer of units whose activation function is selected from a class of functions called basis functions [2, 11].

The RBF is based on the idea that the input patterns form clusters in the input space. If the centers of these clusters are known then the distance of a given input pattern from the cluster centre can be measured. RBF’s are embedded in a two layer neural network, where each hidden unit implements a radial activated function. The output units implement a weighted sum of hidden unit outputs.

RBF for classification: Probabilistic Neural Network (PNN) are feedforward networks that try to

approximate the underlying probability density function of the patterns being classified [11].

In RBF networks it is assumed that Gaussian functions make good approximations to the cluster distribution in the pattern space. If Gaussians are used, then they are centered over each data point from a class [11].

Each unit in the output layer has weights of 1 and a linear output function, so this layer simply adds all of the outputs from the hidden layer that correspond to data from the same class together. This output represents the probability that the input data belongs to the class represented by that unit. The final decision as to what class the data belongs to is simply the unit in the output layer with the largest value.

We have chosen Radial Basis Function (RBF) Network as a classifier for the following reasons:

- Compared with back propagation networks, RBF networks are less susceptible to problems with non-stationary inputs because of the behavior of the RBF hidden units and they usually train much faster.
- Previous studies which implemented on Arabic dataset, have not considered RBF networks.
- Compared with Support Vector Machine (SVM) which considered among the best classifiers, they scale poorly on large data size. So we can show the effectiveness of our FS proposed method.

PARTICLE SWARM OPTIMIZATION

PSO is a population-based stochastic optimization technique, which was developed by Kennedy and Eberhart in 1995 [8]. PSO is initialized with a population of particles. Each particle is treated as a point in an *S*-dimensional space. The *i*th particle is represented as $X_i = (x_{i1}, x_{i2},\ldots, x_{iS})$. The best previous position (pbest, the position giving the best fitness value) of any particle is $P_i = (p_{i1}, p_{i2},\ldots, p_{iS})$. The index of the global best particle is represented by 'gbest'. The velocity for particle *i* is $V_i = (v_{i1}, v_{i2},\ldots, v_{iS})$. The particles are manipulated according to the following equations:

$$v_{id} = w*v_{id}+c1*rand()*(p_{id}-x_{id})+c2*Rand()*(p_{ad}-x_{id}) \quad (2)$$

$$x_{id} = x_{id} + v_{id} \quad (3)$$

where *w* is the inertia weight, The acceleration constants c1 and c2 in equation (2) represent the weighting of the stochastic acceleration terms that pull each particle toward pbest and gbest positions. rand() and Rand() are two random functions in the range [0,1]. Particle's velocities on each dimension are limited to a maximum velocity V_{max}.

The Original PSO is basically developed for continuous optimization problems. To perform FS, the standard PSO concept needs to be extended in order to deal with binary data. In particular, the search space *D* may be a finite set of states and the fitness function *f* a discrete function. Several versions of discrete and binary PSO are proposed in the literature [7, 13].

The proposed algorithm: PSO-based feature selection: We represent the particle's position as binary bit strings of length N, where N is the total number of attributes. Every bit represents an attribute, the value '1' means the corresponding attribute is selected while '0' not selected. Each position is an attribute subset. Velocity and Position are calculated as in Formulas (2) & (3) then we apply a sigmoid transformation (Eq.4) of the velocity component, which can compress velocities in a range [0, 1].

$$S(v_{id}^{new}) = \frac{1}{1+e^{-v_{id}^{new}}}$$
$$\text{if } (rand < S(v_{id}^{new})) \text{ then } x_{id}^{new} = 1; \quad (4)$$
$$\text{else } x_{id}^{new} = 0$$

where: x_{id}^{new} current value of the dimension "d" of the individual "i". v_{id}^{new} current velocity of the dimension "d" of the individual "i".

Fitness function: We use the following fitness function

$$\text{Fitness} = \alpha * \gamma(F_i(t)) + \beta * \frac{|N|-|F|}{|N|} \quad (5)$$

Where F_i (t) is the feature subset found by particle i at iteration t, γ F_i (t) is the classification quality of the features selected, |F| is the length of selected feature subset. |N| is the total number of features. α and β are two parameters that correspond to the importance of classification quality and subset length, with $\alpha \in [0,1]$ and $\beta = 1-\alpha$. In our experiment we assume that classifier performance is more important than subset length, so they were set as $\alpha = 0.85$, $\beta = 0.15$.

In the algorithm, the inertia weight decreases along with the iterations. The initial value of the weighting coefficient was set to 1.2 and the final value of the weighting coefficient was set to 0.4, The swarm size was set to 30. C1, c2: positive acceleration constants were set to 2. The block diagram of the proposed algorithm is illustrated in Fig. 1.

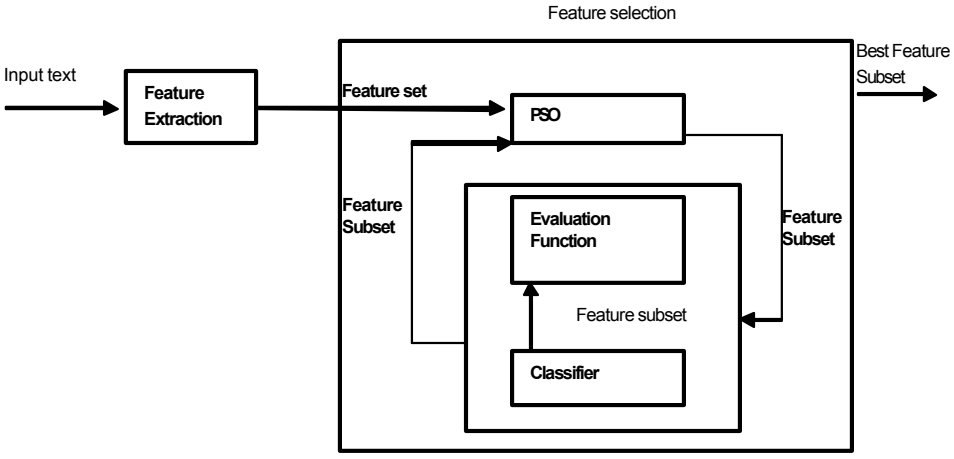

Fig. 1: Block diagram of proposed feature selection algorithm

The process for implementing the PSO algorithm is as follows:

- Initialize a population of particles with random positions and velocities on *S* dimensions in the feature space. Initialize P_i with a copy of X_i and initialize P_g with the index of the particle with the best fitness function value among the population.
- For each particle, evaluate the desired optimization fitness function (Formula 5) in *d* variables.
- Compare the particle's fitness evaluation with particle's pbest. If the current value is better than pbest, then set pbest value equal to the current value and the pbest location equal to the current location in *d* dimensional space.
- Compare fitness evaluation with the population's overall previous best. If current value is better than gbest, then reset gbest to the current particle's array index and value.
- Change the velocity and position of the particle according to Formulas (2) and (3).
- Loop to 2) until a criterion is met, usually a sufficiently good fitness or a maximum number of iterations (generations).

EXPERIMENTAL WORK

Arabic dataset: The Arabic corpus has been collected form online Arabic news papers archives, including Al-Jazeera, Al-Hayat, Al-Ahram and Al-Dostor as well as a few other specialized web sites. In this Arabic dataset, each document was saved in a separate file within the directory for the corresponding category, i.e., the documents in this dataset are single-labeled. Table 1 shows the number of documents for each category.
The Arabic dataset are preprocessed as follows:

Table 1: Arabic dataset

Category name	Train	Test	Total
Politics	342	149	491
Economics	579	253	832
Religion	715	313	1028
Art	208	91	299
Education	155	68	223
Medicine	152	67	219
Science	216	84	300
Engineering	292	120	412
Law	492	204	696
Computer	483	200	683

Total number of articles 5183

- Each article in the Arabic dataset is processed to remove digits, numbers, hyphens, punctuation marks and all the non Arabic characters.
- We have normalized some letters to unify the writing forms.
- Arabic stop words like pronouns, articles and prepositions were removed.
- Stemming: We have not applied any stemming, because it is not always beneficial for Arabic TC tasks, since many terms may be conflated to the same root form [10].
- Rare words (occur in two documents or less) are removed from the specified category.
- The vector space representation [12] is used to represent the Arabic text articles. the weight is calculated using the formula

$$w_{kj} = \frac{tf \times idf(t_k, d_j)}{maxtf} \quad (6)$$

where w_{kj} is the weight of the term k in document j. *tf* is the term frequency (measures the importance of the

Table 2: The performance (precision and recall) of CHI, DF, TF×IDF and PSO on Arabic-dataset

	CHI		DF		TF×IDF		PSO	
Category	Recall	Precision	Recall	Precision	Recall	Precision	Recall	Precision
Politics	94.6	92.8	63.8	97.0	95.3	90.8	99.0	90.9
Economics	99.0	91.1	95.0	93.0	92.5	91.9	99.0	89.0
Religion	98.0	89.0	98.0	92.3	91.0	91.5	99.0	92.1
Art	97.8	93.7	92.3	89.4	96.0	92.9	98.0	87.0
Education	99.0	90.7	47.1	94.0	93.0	90.7	98.0	90.7
Medicine	98.0	82.7	97.0	84.8	96.0	84.8	99.0	90.0
Science	25.0	99.0	13.1	99.0	94.8	89.2	97.6	89.1
Engineering	99.0	90.8	98.0	95.0	91.0	90.2	93.0	93.0
Law	99.0	89.9	99.0	89.4	92.5	89.4	99.5	90.0
Computer	50.0	90.0	50.0	99.0	99.0	91.3	98.0	91.3

Table 3: Macro-F1 of four algorithms

Feature selection algorithms	Macro-F1
CHI	85.5
DF	79.0
TF×IDF	92.1
PSO	93.9
W/O FS	65.0

terms inside the document). *idf* is the inverse document frequency(measures the importance of the terms in the whole collection).

PSO-based FS experimental setup: The main steps of the PSO-based FS experiment is listed here:

- We have formed ten data groups. Each group contains training and testing articles for each category in the Arabic dataset. We have added negative examples from other categories for each group for both training and testing articles with ten percent.
- The documents in each group are preprocessed (section 5.1).
- Then the PSO-based FS method is applied to the whole feature space for each group separately to select the best FS subset that better represent the FS space (PSO algorithm is outlined in Section 4.1).
- The optimized FS subset is applied to a machine learning algorithm to perform categorization task (binary classification) according to the specified learning algorithm.
- To measure the performance of the overall TC system, reca1l, precision, F1 measure and other measures are calculated and macro-averaged.

Experimental results: We have used the Arabic dataset described in Table 3 for training and testing the Arabic text classifier. We have used the Matlab R2007a, Neural network toolbox, probabilistic RBF network (NEWPNN) as the text classifier. Table 2 and 3 shows the results of the PSO-based FS algorithm applied on the Arabic dataset.

Analyzing the precision and recall shown in Table 2, we see that on average, the PSO-based FS algorithm obtained a higher accuracy value than the tf×idf, document frequency and CHI.

CHI method had a good results excepts in two data groups namely science and computer. The performance was very low. This is because the structure of these categories was broadband and resembles other scientific categories like engineering and medicine. tf×idf showed a good results in all data groups. This is indicates that the formula used for calculating tf×idf is robust one (Formula 6).

Figure 2 shows the macro-averaged F1 measure for each of the feature selection algorithms as we change the number of selected features. We see the superiority of the PSO-based FS algorithm compared with other FS methods. Also we observe that when the number of the dimensions were very low, the overall performance is decreased significantly.

Table 3 describes macro-F1 for four feature selection algorithms and the RBF classifier without FS. From this table, we can see that the best categorization performance is achieved with PSO, tf×idf and then CHI.

It is clear from Table 3 that the PSO based FS had significantly outperformed the RBF classifier in terms of macro averaging precision, macro averaging recall and macro averaging F1 measure.

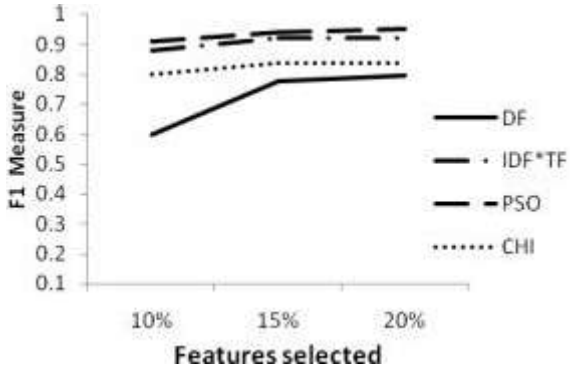

Fig. 2: Macro-averaging F1 measure values of four algorithms

CONCLUSION

In this paper, we have introduced a PSO-based feature selection algorithm, a set of experiments were carried out on the Arabic dataset. The experimental results showed the superiority of the developed PSO algorithm compared with Document frequency, tf×idf and CHI in terms of better classification accuracy. Among the statistical approaches, the CHI and tf×idf have the best classification accuracy.

The Inertia parameters (w), position-updating strategy and the fitness function have an important impact on the performance of PSO. In our work we have adjusted and tuned PSO parameters empirically.

From the results obtained, we conclude that PSO has powerful exploration ability; it is a gradual searching process that approaches optimal solutions. Interaction in the PSO enhances progress toward the solution.

REFERENCES

1. Aghdam, M., N. Ghasem-Aghaee and M. Basiri, Text feature selection using ant colony optimization. Expert Systems with Applications doi:10.1016/j.eswa.2008.08.022
2. Bishop, C., 1995. Neural Networks for Pattern Recognition. Clarendon Press, Oxford.
3. Chen, C., H. Lee and Y. Chang, 2009. Two Novel Feature Selection Approaches for Web Page Classification. Expert Systems with Applications, 36: 260-272.
4. Duda, R., P. Hart and D. Stork, 2000. Pattern Classification. Whiley-Interscience, 2nd Edn.
5. Forman, G., 2003. An Extensive Empirical Study of Feature Selection Metrics for Text Classification. Journal of Machine Learning Research, 3: 1289-1305.
6. Elbeltagi, E., T. Hegazy and D. Grierson, 2005. Comparison among five evolutionary-based optimization algorithms. Advanced Engineering Informatics, 19 (1): 43-53.
7. Karthi, R., S. Arumugam and K. RameshKuma, 2009. A Novel Discrete Particle Swarm Clustering Algorithm for Data Clustering. In Proceedings of the 2nd Bangalore Annual Compute Conference on 2nd Bangalore Annual Compute Conference (Bangalore, India, January 09-10, 2009). COMPUTE '09. ACM, New York, NY, 1-4. DOI= http://doi.acm.org/10.1145/1517303.1517321
8. Kennedy, J. and R.C. Eberhart, 1995. Particle Swarm Optimization. Proc. IEEE, International Conference on Neural Networks. Piscataway.
9. Kohavi, R. and G. John, 1997. Wrappers for Feature Subset Selection. Artificial Intelligence, 97 (1-2): 273-324.
10. Mesleh, 2008. A. Support Vector Machine Text Classifier for Arabic Articles: Ant Colony Optimization-Based Feature Subset Selection. Ph.D Thesis. The Arab Academy for Banking and Financial Sciences, Jordan.
11. Picton, P., 2000. Neural Networks. 2nd Edn. Palgrave.
12. Sebastiani, F., 2002. Machine learning in automated text categorization. ACM Computing Surveys, 34 (1): 1-47.
13. Shi, Y. and R.C. Eberhart, 1998. A modified particle swarm optimizer. In: Proc. IEEE Int. Conf. on Evolutionary Computation. Anchorage, AK, USA, pp: 69-73.
14. Syiam, M., Z. Fayed and M. Habib, 2006. An Intelligent System for Arabic Text Categorization. International Journal of Intelligent Computing and Information Sciences, 6 (1): 1-19.
15. Yang, J. and V. Honavar, 1998. Feature Subset Selection Using a Genetic Algorithm. IEEE Intelligent Systems, 13 (2): 44-49.
16. Yang, Y.M. and J.O. Pedersen, 1997. A Comparative Study on Feature Selection in Text Categorization. In J. D.H. Fisher, Editor, The 14th International Conference on Machine Learning (ICML'97), Morgan Kaufmann, pp: 412-420.