Journal of Advances in Computer Research
Quarterly pISSN: 2345-606x eISSN: 2345-6078
Sari Branch, Islamic Azad University, Sari, I.R.Iran
(Vol. 9, No. 1, February 2018), Pages: 29-40
www.jacr.iausari.ac.ir

An Improved Flower Pollination Algorithm with AdaBoost Algorithm for Feature Selection in Text Documents Classification

Hiwa Majidpour, Farhad Soleimanian Gharehchopogh✉
Department of Computer Engineering, Urmia Branch, Islamic Azad University, Urmia, Iran
hiwa.majidpour@gmail.com; bonab.farhad@gmail.com

Received: 2015/11/25; Accepted: 2016/03/14

Abstract

In recent years, production of text documents has seen an exponential growth, which is the reason why their proper classification seems necessary for better access. One of the main problems of classifying text documents is working in high-dimensional feature space. Feature Selection (FS) is one of the ways to reduce the number of text attributes. So, working with a great bulk of the feature space without FS increases the computational cost which is a function of the length of the vector, and also, it helps to remove irrelevant attributes. The general approach in this paper combines the hybrid of Flower Pollination Algorithm (FPA) with Ada-Boost algorithm. The FPA is used for FS and the Ada-Boost is used for classification of text documents. Tests were conducted on Reuters-21578, WEBKB and CADE 12 datasets. The results show that the hybrid model has higher detection accuracy in FS compared with Ada-Boost algorithm with model. And comparisons are indicative of higher detection accuracy of the proposed model compared with KNN-K-Means, NB-K-Means and learning models.

Keywords: *Classification of Text Documents, Feature Selection, Flower Pollination Algorithm, ADA-Boost Model*

1. Introduction

Nowadays, many documents are stored in the form of electronic and text files. Classification of text documents is an essential model to extract knowledge from this large volume of text information. Information classification as one of the most important techniques in information retrieval and natural language processing is challenging and an effective way to organize text databases [1]. We can consider information classification as the most important classification technique without monitoring, that classifies input space to k zones based on the similarities and differences [2]. So that, targets in the same groups must be similar and the targets in the different groups must be different.

With the development of electronic texts and documents, using an efficient way to retrieve information is essential. To retrieve data, understand the meaning of the original text, classify texts, and to find the right words to search articles extracting keywords is the best method. Keywords are a series of important words in a document which describes the content of the document and are used for different purposes. By finding

keywords, we can grasp the concept of text documents more smoothly and in a shorter time.

After the initial preparation of text documents where the documents are in different classes, pre-processing phase is done. In fact, preprocessing is the first step towards complying documentation by displaying them in a convenient format [3]. The purpose of this phase is to find useful words and delete useless words. Based on the frequency of words and a consideration of the text as a series of words, the classification is conducted. In this paper, in order to reduce the number of features and choose the major characteristics of the original text, FPA [4] will be used. In fact, using FPA, the main characteristics of the texts are selected, and using Ada-Boost algorithm they are classified [5].

The FPA [4] was inspired by the pollination of flowers. This algorithm uses an iterative strategy in order to achieve optimal and near-optimal solution. In Ada-Boost algorithm, in each iteration, weights are assigned to all samples which determine the chances of the samples to be selected for the training set. In this method, all of the samples are initially assigned a primary weight. In the next step, according to the results of the previous stage, newly- applied weights, sample selection, and classification are performed according to the new weights. This cycle continues until the error is less than a certain amount.

There are certain classes in classifying a training set of documents. With this set, the model of classification is specified, and the class of new document is determined [6]. To measure the efficiency of the classification model of text documents, a set of is considered independently of the training set. The estimated labels are compared with the real labels of documents. The proportions of correctly classified documents in comparison with the total number of documents are achieved based on health care accuracy. The hybrid model makes use of FPA for feature extraction and Ada-Boost algorithm for classification. To select features, some vectors are formed, each of which contains the weight of key words; and then the fitness function of the vectors are assessed. Afterwards, the best properties are selected on the basis of the fitness function.

The overall structure of the present paper is organized as follows: in the Section 2, the related works are presented in conjunction with classified documents and proposed models. The Section 3, discusses the proposed model. In the Section 4, the proposed model is evaluated and compared with other models; and finally, in the Section 5, conclusion and future work are presented.

2. Related Works

In recent years, the importance of classifying texts as highly information-potential is widely regarded in a way that discovering knowledge is one of the most important interests of processing practitioners. Many researchers have been conducted in this field which we will be examining in this section.

Naive Bayes (NB), Support Vector Machine (SVM), Logistics Regression, Ada-Boost and NB, Ada-Boost and SVM hybrid models have been proposed to classify text documents [7]. Evaluation was done on ACM data base. The results have indicated that accuracy of NB model is better on the diagnosis criteria for the frequency of words compared to other models. K-Nearest Neighbor (KNN) and TF-IDF has been proposed for classifying text documents [8]. KNN has been used to classify documents and TF-IDF algorithm has been used to extract keywords. Results were conducted on WEBKB

data set. The results showed that the highest accuracy in KNN Classification is equal to 0.92%.

Hybrid SVM, NB and KNN hybrid models have been proposed to classify text documents [9]. Assessment was done on TCFPF dataset. The accuracy of WEBKB, Reuters, Newsgroups datasets fort the three datasets are respectively 86.19, 75.47 and 89.09, which compared with SVM models, NB and KNN models are more accurate. A new model has been proposed to classify documents based on WordNet [10]. WordNet is a large lexical database composed of English vocabulary. This database classifies nouns, verbs, adjectives and adverbs into a set of synonymous word clusters each of which having its own concept. The synonymous sets are linked together using conceptual semantic relations and lexical relations. In the proposed model, word weighting is based on the similarity.

KNN algorithm and SVM models have been proposed to classify 20 News-Groups documents [11]. To obtain the frequency numbers of words, we have made use of TF-IDF. Also in SVM model, kernel function has been used for data training. The results show that the accuracy of KNN model is more than SVM model. The accuracy of KNN in Accuracy criterion has been more than SVM model by training 80% of the data.

KARABULUT has used Particle Swarm Optimization (PSO) algorithm to reduce the dimensions of the data [12]. He has utilized PSO algorithm with fuzzy combination plus SVM and the NB models. The results have been assessed on Reuters and OHSUMED datasets. Fuzzy model has higher Precision accuracy in comparison with NB and SVM models.

The hybrid model of NB-K-Means models for the classification of text documents has been tested based on Reuters, WEBKB and CADE 12 databases [13]. The results show that the hybrid NB-K-Means model is more accurate compared with K-Means. Moreover, the highest accuracy in the hybrid model belongs to K=3 which equals 93.30. KNN-K-Means hybrid model has been proposed for clustering text documents [14]. In this model, KNN is used to identify similar clusters. Results on Reuters-21578 dataset show that the hybrid model is more accurate in comparison with K-Means model.

3. The Proposed Model

One of the main stages in classifying text documents that is composed of a collection of different texts is the FS. The purpose of the FS is that a subset of features is selected to increase the accuracy of prediction. In this paper, using FPA the features on the complex text documents are chosen, and then the selected properties are given to Ada-Boost to calculate classification accuracy. In the proposed model, we assign a weight to each of the features selected. With the help of obtained features, we form the feature vectors in the FPA. FPA does the act of updating to achieve the most optimal feature vector. The fitness function of each vector is calculated, and the feature vectors with the highest average are chosen. In Figure (1), the flowchart of the proposed model is shown.

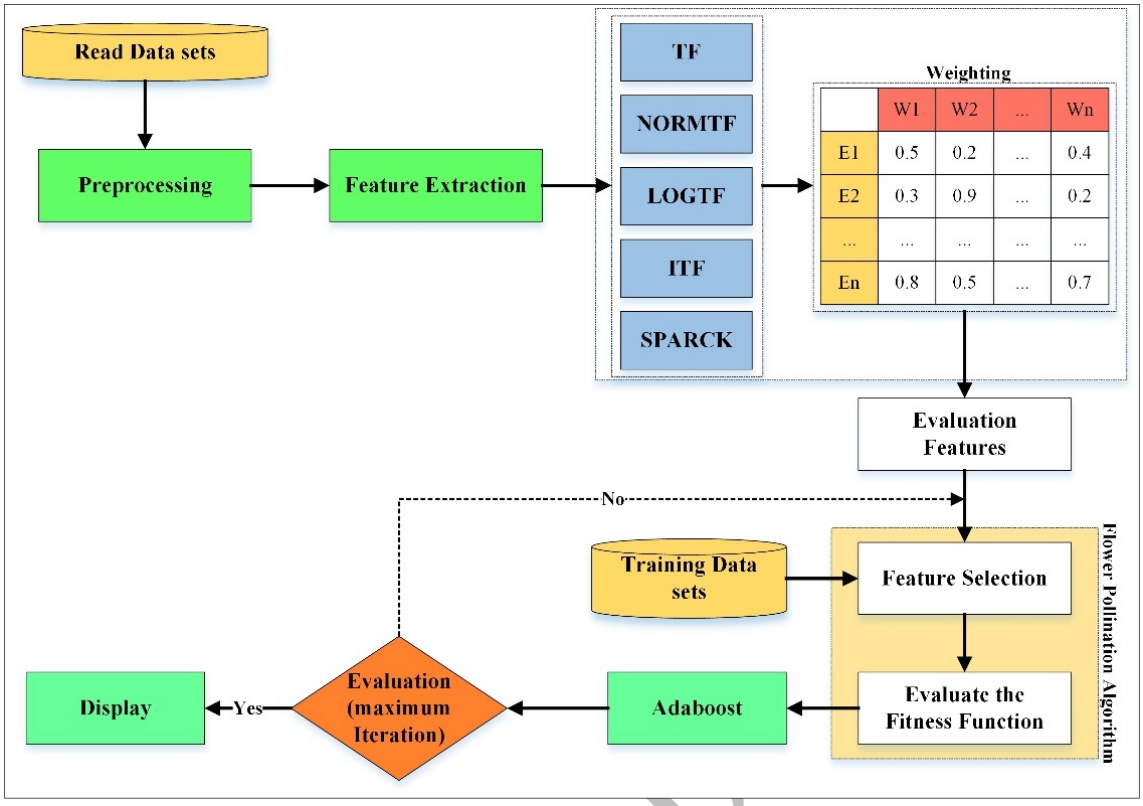

Figure 1: The Flowchart of the Proposed Model

In the proposed model, first of all, pre-processing operation which involves the removal of signs and irrelevant words carried out. Afterwards, the act of indexing which includes keyword extraction is done. Words weight is one of the most important factors in indexing which determines the role of words in terms of their impact as the text keywords. At this point, using different weighting patterns, weight is assigned to each extracted word. This weight reflects the impact of the word in the subject matter of documents as opposed to the other words used in the text.

In indexing phase, based on TF method the keywords are initially extracted and then weighed. In the proposed model, we have made use of different TF methods such as TF [15], NORMTF [16], LOGTF [16], ITF [17] and SPARCK [18]. Each of the TF methods is evaluated in the proposed model. In the next stage, FS is carried out based on FPA. FS from among weights is done based on the number of features.

3.1 Vector Creation in Flower Pollination Algorithm

To create a vector in the FPA, we use weight values which have been set for features. In the proposed model, the first values of the vectors are filled with word weights. Then, using cross-pollination operators on the basis of Equation (1) and self-pollination operators based on the Equation (2), we do the acts of updating and position-changing on weights.

$$x_i^{t+1} = x_i^t + \gamma L(g^* - x_k^t) \quad (1)$$

$$x_i^{t+1} = x_i^t + \varepsilon(x_j^t - x_k^t) \quad (2)$$

Self-pollination operation makes the range of weights diverse and incorporates the weak neighbors in the classification too. These operators cause the weight limits to have

neighborhoods and proximity features for the classification of Ada-Boost and to increase classification accuracy. In the proposed model, in order to improve FPA, the hybrid of cross-pollination and self- pollination is used according to the Equation (3).

The aim of the Equation (3) is improving the FPA's solutions. Because some of the solutions in the problem space are close to local optimization and they are not able to escape from it, then using Equation (3), it can accompany all of the solutions in the fitness function and chose the best solution for classification.

$$x_i^{t+1}(new) = x_i^t + (x_j^t - x_k^t) + (g^* - x_k^t) \quad (3)$$

In the proposed model, firstly a population of vectors as shown in Figure (2) is made on the basis of weight values; secondly on each vector, updating and position-changing operators are conducted.

	W_1	W_2	W_3	W_4		W_n
X_1	W_{11}	W_{12}	W_{13}	W_{14}		W_{1n}
X_2	W_{21}	W_{22}	W_{23}	W_{24}		W_{2n}
X_3	W_{31}	W_{32}	W_{33}	W_{34}		W_{3n}
.	.	.	.	.	.	.
X_m	W_{m1}	W_{m2}	W_{m3}	W_{m4}		W_{mn}

Figure 2: The Formation of Vectors in the Pollinate Algorithm

By starting the primary population in the problem space, and by using cross-pollination and self-pollinating operators, FPA moves toward a totally optimal space. Then, the value of each of the products is calculated, and objective function is assessed for each vector. If there were no better response than the initial response produced in between sets of vectors formed, particles are updated, and the operation continues till the most optimal vector is formed. In the vector model, we consider each document as a vector of words and create a multi-dimensional space whose dimensions are made up of words. Then, each document is displayed in this space in the form of a vector. Values of vectors in fact are weights that show who each of the words play roles in distinguishing documents. Thus, each document with a set of words becomes a weight vector. In Figure (3), an example of the production of cycle vectors is shown that is based on the weight values.

Vectors	Word1	Word2	Word3	Word4	Word5	Word6	Word7	Word8	Word9	Word10
V1	0.2	0.3	0.4	0.6	0.8	0.3	0.4	0.9	1.0	1.2
V2	0.6	0.9	0.6	1.8	0.3	1.0	1.6	1.0	1.7	0.9
V3	0.3	0.3	0.5	0.8	1.8	0.9	1.5	1.9	1.2	0.3
V4	0.8	0.8	1.2	0.9	0.4	1.9	0.8	0.4	0.2	2.5
V5	1.3	1.7	1.5	0.1	0.9	0.6	0.3	0.9	0.4	2.3

Figure 3: Sample of the Formation of Vectors with Weight Values

The proximity of vector values criterion with cosine distance method is calculated according to Equation (4) [19]. Educational documents are ranked based on their similarity to X document. Then k is the document that has the closest resemblance, and d is keywords based on the values weight.

$$sim(x,d_i)=\frac{\sum_{k=1}^{m} x_k \times d_{ik}}{\sqrt{\sum_{k=1}^{m} x_k^2 \sum_{k=1}^{m} d_{ik}^2}} \tag{4}$$

$$p(x,C)=\sum_{d_i} sim(x,d_i)\,y(d_i,C_j) \tag{5}$$

$$y(d_i,c_j)=\begin{cases}1, d_i & \in C_j \\ 0, d_i & \in C_j\end{cases} \tag{6}$$

The degree of X document belonging to each C_j category is calculated according to the Equation (5). In Equation (4), $y(d_i,c_j)$ calculates X document belonging to the category based on the Equation (6).

3.2 Ada-Boost Algorithm

In the proposed model, Ada-Boost algorithm is used of classification to obtain any document belonging to a particular class according to its properties. This algorithm finds categories based on weight values. And it repeats the operation of category-formation based on continuous updates of values. In the proposed model, the weights will increase in each iteration, and the weight of samples which are mistakenly categorized will increase while the weights of those which have been rightly categorized will decrease. Ada-Boost algorithm makes use of the total set of data in order to train each classifier, but after each training, it mainly focuses on data with more weight so that they can be correctly classified. With each iteration, the distribution of training data is on the samples which are accurately categorized.

4. Evaluation and Results

In this section, the evaluation and the results of the proposed model have been carried out in VC#.NET 2016 programming. The evaluation has been done on Reuters-21578,

WEBKB and CADE 12 datasets. Various criteria have been regarded for the investigation of word frequencies because of the efficiency of detection accuracy of the data sets. The results of the proposed model should be analyzed at the evaluation stage in order to determine its value, and subsequently determine its effectiveness. These criteria can be calculated both for training data set at learning stage and for experimental record set at evaluation stage. There are various criteria such as Precision, Recall, F-Measure and Accuracy for assessment which uses accuracy criterion for the evaluation of the proposed model [20].

$$\Pr ecision = \frac{TP}{TP + FP} \tag{7}$$

$$\mathrm{Re}call = \frac{TP}{TP + FN} \tag{8}$$

$$F - Measure = \frac{2 * \Pr ecision * \mathrm{Re}call}{(\Pr ecision + \mathrm{Re}call)} \tag{9}$$

$$Accuracy = \frac{(TP + TN)}{(TP + TN + FP + FN)} \tag{10}$$

TN stands for the number of records whose actual category has been negative, and classification algorithm has rightly detected their categories to be negative. TP indicates the number of records whose real category has been detected to be positive, and their classification algorithm has rightly detected them to be positive.

FP stands for the number of records whose real categories have been negative, but the classification algorithm has wrongly detected them to be positive. FN stands for the number of records whose real categories have been positive, but the classification algorithm has wrongly detected them to be negative.

4.1 Ada-Boost Model

In Table (1), the datasets have been evaluated using Ada-Boost model based on TF models. The impact of each TF is different on accuracy criteria. In Reuters-21578 dataset, ITF model is the most accurate compared with other models. And in WEBKB and CADE 12 datasets, NORMTF and ITF models are respectively the most accurate.

Table 1: Evaluation of Data Sets Using Ada-Boost

Data sets	TF	Evaluation Criteria's			
		P	R	F-Measure	Accuracy
Reuters-21578	TF	70.23	69.32	69.77	66.35
	NORMTF	72.32	65.10	68.52	69.48
	LOGTF	76.09	66.87	71.18	63.47
	ITF	77.94	68.24	72.77	70.35
	SPARCK	74.29	66.07	69.94	62.15
WEBKB	TF	76.38	68.00	71.95	66.37
	NORMTF	72.24	69.94	71.07	69.48
	LOGTF	76.54	65.37	70.52	65.02
	ITF	75.98	64.87	69.99	62.17
	SPARCK	76.00	67.47	71.48	65.98
CADE 12	TF	70.32	70.05	70.18	63.27
	NORMTF	76.94	68.34	72.39	64.00
	LOGTF	74.00	65.08	69.25	62.44
	ITF	76.57	71.24	73.81	69.89
	SPARCK	74.01	65.46	69.47	63.24

4.2 Proposed Model

In Table (2), datasets have been evaluated using the proposed model and on the basis of the impact of TF models. The impact of each TF is different in accuracy criterion.

Table 2: Evaluation of Data Sets Using the Proposed Model

Data sets	TF	Evaluation Criteria's			
		P	R	F-Measure	Accuracy
Reuters-21578	TF	72.65	65.32	68.79	75.06
	NORMTF	69.35	64.77	66.98	76.34
	LOGTF	73.00	62.00	67.05	77.00
	ITF	76.00	69.35	72.52	80.54
	SPARCK	72.14	68.11	70.07	75.66
WEBKB	TF	75.23	61.02	67.38	80.13
	NORMTF	71.77	66.55	69.06	76.35
	LOGTF	73.68	68.08	70.77	83.99
	ITF	76.04	69.77	72.77	85.47
	SPARCK	79.58	70.36	74.69	73.48
CADE 12	TF	80.10	68.08	73.60	76.38
	NORMTF	73.98	69.11	71.46	72.88
	LOGTF	76.58	65.00	70.32	73.24
	ITF	82.36	80.35	81.34	80.35
	SPARCK	75.09	71.68	73.35	76.00

4.3 Proposed Model with Feature Selection

In Table 3, data sets have been evaluated using the proposed model based on FS. Each feature has different results. The results have shown that FS with 80 is more accurate than that with 40. Because more features participate in the classification and the possibility to reach more accurate answer is high.

Table 3: Evaluation of Data Sets Based on FS

Data sets	TF	FS =40				FS =80			
		P	R	F-Measure	Accuracy	P	R	F-Measure	Accuracy
Reuters-21578	TF	95.36	78.35	86.02	95.35	96.34	83.54	89.48	93.55
	NORMTF	94.25	79.24	86.10	94.27	92.87	82.14	87.18	95.00
	LOGTF	93.27	69.34	79.54	89.32	95.11	80.27	87.06	92.89
	ITF	96.64	70.00	81.19	95.47	93.60	82.81	87.88	97.70
	SPARCK	97.84	72.05	82.99	94.00	97.11	83.02	89.51	91.58
WEBKB	TF	94.30	71.22	81.15	90.14	96.34	67.24	79.20	92.36
	NORMTF	93.67	76.28	84.09	89.67	94.37	82.00	87.75	95.84
	LOGTF	96.84	77.94	86.37	95.34	95.10	79.35	86.51	94.30
	ITF	96.54	76.20	85.17	96.17	98.75	79.48	88.07	96.27
	SPARCK	95.49	80.00	87.06	92.13	96.40	82.97	89.18	93.87
CADE 12	TF	91.36	72.34	80.75	93.17	95.01	76.34	84.66	95.66
	NORMTF	93.64	73.24	82.19	90.25	96.48	80.03	87.49	92.37
	LOGTF	97.20	67.54	79.70	94.88	96.84	83.78	89.84	97.45
	ITF	92.34	75.01	82.78	95.20	98.00	79.01	87.49	96.80
	SPARCK	94.74	75.13	83.80	93.06	90.35	80.69	85.25	96.37

4.4 Comparison and Evaluation

In this part, we will compare the proposed model with different models that have been evaluated by the researchers. In Table 4, a comparison of the proposed model based on 80 features with NB-K-Means [13] model has been carried out on Reuters, WEBKB and CADE 12 datasets.

Table 4: Comparison of the Proposed Model with K-Means-NB Model based on 80 Features

Models	NB-K-Means [13]					Proposed Model
Data sets	K=3	K=4	K=5	K=6	K=7	FS =80
Reuters-21578 R8	91.60	90.10	83.40	77.70	71.60	93.45
Reuters-21578 R52	88.50	84.80	74.30	72.80	70.50	92.14
WEBKB	94.80	93.20	78.20	69.90	62.30	96.27
CADE 12	88.10	85.80	70.30	67.00	55.50	97.45

In Table 5, the comparison of the proposed model with KNN-K-Means model with 80 features [14] is done on the Reuters-21578 datasets.

Table 5: Comparison of the Proposed Model with K-Means-KNN based on 80 Features

Models		K-Means [14]			KNN-K-Means [14]			Proposed Model
Number of Documents		K=3	K=4	K=5	K=3	K=4	K=5	FS =80
Reuters-21578	100	76.85	68.23	55.92	87.32	72.37	67.15	97.23
	250	72.15	65.14	58.16	78.52	69.11	64.92	96.54
	500	65.83	54.20	48.92	71.23	68.88	62.25	96.03

Table (6) shows the comparison of the proposed model with learning models based on 80 features on Reuters-21578 data set.

Table (6): The Comparison of the Proposed Model based on 80 Features with Learner Models

Models[21]	Accuracy [21]					F-Measure [21]				
	CSI	EB	MF	TS-ISF	TR	CSI	EB	MF	TS-ISF	TR
NB [21]	71.08	82.38	83.70	76.19	81.62	68.00	83.00	87.00	76.00	81.00
SVM [21]	64.70	74.30	78.53	70.78	77.98	65.00	79.00	85.00	71.00	76.00
LR [21]	66.70	76.94	76.24	72.71	78.79	65.00	79.00	84.00	73.00	79.00
RF [21]	70.57	79.94	81.73	75.42	79.78	67.00	81.00	87.00	75.00	80.00
Bagging +RF [21]	73.34	87.37	91.49	82.39	88.96	71.00	87.00	92.00	82.00	89.00
Random Subspace +RF [21]	72.67	85.77	91.42	82.19	88.63	70.00	85.00	91.00	81.00	88.00
Majority Voting [21]	72.94	83.18	86.83	76.90	82.64	70.00	83.00	87.00	75.00	82.00
Proposed Model	TF	NORM TF	LOGT F	ITF	SPAR CK	TF	NORM TF	LOGT F	ITF	SPAR CK
	93.55	**95.00**	**92.89**	**97.70**	**91.58**	**89.48**	**87.18**	87.06	**87.88**	**89.51**

As shown in Table (6), the proposed model is more accurate than learner models. In learner models of CSI, EB, MF, TS-ISF and TR are used for word frequencies. Also in the F-Measure criterion, bagging+ RF [21] model based on MF has better detection compared with the proposed model. Since the machine learning models use mostly the decision and rules to classify, then, the possibility of similarity between the categories is less likely. But in proposed model, FS is used to reduce the size of features as well as the similarity between the categories of weight and the distance between the words.

5. Conclusion and Future Works

In this paper, we have made use of FPA and Ada-Boost in order to classify text documents. In Ada-Boost model, with frequent repetitions on the weight of the keywords, educational documents are classified and then tested. In the proposed model, firstly, we recognized the preprocessing of the keywords using the word frequencies. This makes the size of classes in the training data more balanced. For FS, FPA has been used. After selecting the properties, documents were trained by Ada-Boost model. Furthermore, the accuracy of the proposed model is higher compared with KNN-K-Means, NB-K-Means models. Various techniques such as statistical methods and machine learning algorithms have been introduced in this field with their own advantages and disadvantages. In future research, in order to solve the problem of FS, we will use a model that is less dependent on the parameter values and it can more efficient choose feature weights.

References

[1] F.S. Gharehchopogh, Z.A. Khalifelu, Analysis and Evaluation of Unstructured Data: Text Mining versus Natural Language Processing, 5th International Conference on Application of

Information and Communication Technologies (AICT2011), IEEE press, pp. 1-4, Baku, Azerbaijan, 12-14 October 2011.

[2] M. Rostami, S.S. Ayat, I. Attarzadeh, F. Saghari, Proposing a Method to Classify Texts Using Data Mining, Journal of Advances in Computer Research, Vol 6, Issue: 4, pp. 125-137, Autumn 2015.

[3] F.S. Gharehchopogh, Z.A. Khalifelu, "Study on Information Extraction Methods in Unstructured Data: Text Mining versus Natural Language Processing", AWERProcedia Information Technology & Computer Science Journal, Vol. 1, pp.1321-1327, 2012.

[4] A. H. Jadidinejad, V. Marza, Building Semantic Kernel for Persian Text Classification with a Small Amount of Training Data, Journal of Advances in Computer Research, Vol 6, Issue: 1, pp. 125-136, Winter 2015.

[5] 14.M.h Haghiri, H. Hassanpour, (2011), Using Supervised Clustering Technique to Classify Received Messages in 137 Call Center of Tehran City Council, Journal of Advances in Computer Research, Vol 2, Issue: 3, pp. 15-24, Summer 2011.

[6] N. Ebrahimpour, F.S. Gharehchopogh; Z. A.Khalifehlou, (2016), New Approach with Hybrid of Artificial Neural Network and Ant Colony Optimization in Software Cost Estimation, Journal of Advances in Computer Research, Vol 7, Issue: 4, pp. 1-12, Autumn 2016.

[7] Onan, S. Korukoglu, H. Bulut, Ensemble of keyword extraction methods and classifiers in text classification, Expert Systems with Applications, Vol. 57, pp. 232-247, 2016

[8] Trstenjak, S. Mikac, D. Donko, KNN with TF-IDF based Framework for Text Categorization, Procedia Engineering, Vol. 69, pp. 1356-1364, 2014.

[9] Y. Ko, J. Seo, Text classification from unlabeled documents with bootstrapping and feature projection techniques, Information Processing & Management, Vol. 45, Issue 1, pp. 70-83, 2009

[10] Q. Luo, E. Chen, H. Xiong, A semantic term weighting scheme for text categorization, Expert Systems with Applications, Vol. 38, Issue 10, pp. 12708-12716, 2011

[11] D.S. Guru, M. Suhil, A Novel Term_Class Relevance Measure for Text Categorization, Procedia Computer Science, Vol. 45, pp. 13-22, 2015.

[12] M. Karabulut, Fuzzy unordered rule induction algorithm in text categorization on top of geometric particle swarm optimization term selection, Knowledge-Based Systems, Vol. 54, pp. 288-297, 2013

[13] A. Allahvirdipour, F.S.Gharehchopogh, New Approach in Features Selection in Text Documents Classification using the Hybrid Model Algorithms of Naïve Bayes and K-Means, Journal of Advances in Computer Research, (Accepted).

[14] R. Habibpour, K. Khalilpour, A New Hybrid K-means and K-Nearest-Neighbor Algorithms for Text Document Clustering, International Journal of Academic Research, Vol. 6 Issue 3, pp. 79-84, 2014

[15] H.P. Luhn, A Statistical Approach to the Mechanized Encoding and Searching of Literary Information, IBM Journal of Research and Development, Vol. 1, No. 4, pp. 309-317, 1957.

[16] F. Raja, M. keikha, F. Oroumchian, M. Rahgozar, Using rich document representation in XML information retrieval, vol. Initiative on the evaluation of XML retrieval (INEX), 2006.

[17] E. Leopold, and J. Kindermann, Text Categorization with Support Vector Machines. How to Represent Texts in Input Space?, Machine Learning, Vol. 46, No. 1-3, pp. 423-444, 2002.

[18] Sparck Jones, K. Indexing Term Weighting, Information Storage and Retrieval, Vol. 9, pp. 619-633, 1973.

[19] G. Salton. Automatic Text Processing: The Transformation, Analysis, and Retrieval of Information by Computer, Addison-Wesley, 1989.

[20] R.S. Michalski, I. Bratko, M. Kubat, Machine Learning and Data Mining: Methods and Applications, New York: Wiley, 1998

[21] H. Uguz, A two-stage feature selection method for text categorization by using information gain, principal component analysis and genetic algorithm, Knowledge-Based Systems, Vol. 24, pp. 1024-1032, 2011.