Text Classification based on Discriminative-Semantic Features and Variance of Fuzzy Similarity

Pouyan Parsafard
Kish International Campus, University ofTehran, Kish, Iran
E-mail: pooyanparsafard@ut.ac.ir

Hadi Veisi
Faculty of New Sciencesand Technologies (FNST), University of Tehran, Tehran, Iran
E-mail: h.veisi@ut.ac.ir

Niloofar Aflaki
Geoinformatics Collaboratory and School of Natural and Computational Sciences, Massey University, Auckland, New Zealand
E-mail: n.aflaki@massey.ac.nz

Siamak Mirzaei*
College of Science and Engineering, Flinders University, South Australia
E-mail: siamak.mirzaei@flinders.edu.au

Abstract: Due to the rapid growth of the Internet, large amounts of unlabelled textual data are producing daily. Clearly, finding the subject of a text document is a primary source of information in the text processing applications. In this paper, a text classification method is presented and evaluated for Persian and English. The proposed technique utilizes variance of fuzzy similarity besides discriminative and semantic feature selection methods. Discriminative features are those that distinguish categories with higher power and the concept of semantic feature takes into the calculations the similarity between features and documents by using only available documents. In the proposed method, incorporating fuzzy weighting as a measure of similarity is presented. The fuzzy weights are derived from the concept of fuzzy similarity which is defined as the variance of membership values of a document to all categories in the way that with some membership value at the same time, the sum of these membership values should be equal to 1. The proposed document classification method is evaluated on three datasets (one Persian and two English datasets) and two classification methods, support vector machine (SVM) and artificial neural network (ANN), are used. Comparing the results with other text classification methods, demonstrate the consistent superiority of the proposed technique in all cases. The weighted average F-measure of our method are %82 and %97.8 in the classification of Persian and English documents, respectively.

1. Introduction

In the age of information and technologies, the rapid growth of data production has faced the research community with the challenge of big data analysis. A large portion of the daily generated data is textual which is mainly unstructured and unlabelled. It is clear that manual reviewing and handling huge amount of unlabelled data is a highly time-consuming and costly process and could be replaced by automatic categorization techniques. Thus, document classification is one of the important subjects in text mining which plays a special role in controlling and managing the growing volume of contextual information. Document classification that is used to recognize a text document label or title, is a subject in the natural language processing field which is also known as topic identification. A document classification method manages the documents to make future processes more efficient and easier. As an application, topic identification makes the documents retrieval simpler and faster.

At the beginning of a text classification process, documents should be represented in a representation method. One of the most common document representation methods is the Vector Space Model (VSM). Almost in most cases cited, a document is modelled as a vector (Harish et al., 2010). Feature extraction is the next step to extract all the helpful features from a text document. Later, feature selection is the action that can be performed optionally, in a text classification process depends on its architecture. Feature selection will pick up the best candidates from all features set (Basu and Murthy, 2012; Saraee and Bagheri, 2013). When a document is represented as a feature vector, every component (feature weight) shows the information which is selected from a document. After all, a text classification could be done using an appropriate classifier.

In this paper, we have proposed a new classification method based on VSM document representation that uses Discriminative Feature Selection (DFS) (Zong et al., 2015) method to select the efficient features in a text document. DFS determines how much a feature has the discriminative power between the categories. Later on, after calculating the similarity between each feature and the document that belongs to it, we will calculate the measure of fuzzy similarity of each document to all categories (Widyantoro and Yen, 2000). Then, the variance of fuzzy similarity values adds to the final weight values. According to the mentioned method, documents are converted to features based on the selected features and their weights. Finally, to do the classification, support vector machine (SVM) and artificial neural network (ANN) classifiers are used. Therefore, the main contributions of the paper are summarized as bellow:

- Using similarity variance along with the discriminative and semantic calculations
- The integration of discriminative features and fuzzy similarity for Persian topic identification.

The structure of the paper is as follows. Section 2 will briefly review the related research studies. In Section 3, the proposed classification method will be explained in detail. Algorithms and method architecture are presented in this section, too. Section 4 provides evaluations results in which datasets, evaluation metrics, evaluations of classification results and comparisons are given. Finally, in Section 5, conclusion and suggestions for the future works are expressed.

2. Related Works

There are various research studies in numerous languages for text classification that used several feature extraction and classification methods. A neural network as a classifier is studied in (Chen et al., 2012) for English, also (Pilevar et al., 2009) for Persian. The best result in the mentioned Persian study is obtained using Optimized Learning Vector Quantization 3 (OLVQ3) classifier that is 89.9% for F-measure on Hamshahri 2 dataset[1]. K-nearest neighbor (KNN) classifier is used in (Basu and Murthy, 2012; Ko et. al., 2004) for English, and in (Elahimanesh et al., 2012; Farhoodi and Yari, 2010) for Persian. The best result report in the forenamed Persian studies using KNN is 93.9% of F-measure on Hamshahri dataset (AleAhmad et al., 2009). Some classification works, incorporating fuzzy similarity are done in English (Miyamoto, 2001; Widyantoro and Yen, 2000) and using the fuzzy model as a classifier in Persian (Yari et al., 2010). The best result reported for Persian using fuzzy relation is 91% of F-measure on a hybrid dataset including Hamshahri website, Persian Wikipedia website and some other Persian websites. Support vector machine (SVM) that is more popular than the other classifiers, is used in (Lan et al., 2009; Hao et al., 2006) for English, and in (Maghsoodi and Homayounpoor, 2011; Farhoodi and Yari, 2010) for Persian. For the mentioned Persian researches using SVM, the best obtained result is 94% of F-measure on Bijankhan dataset (Bijankhan, 2008). As a statistical classifier, Naïve Bayes is applied in test classification tasks in English (Qian et al., 2007; Ko et al., 2004) and in Persian (Jafari et al., 2011). For the mentioned Persian classification research based on Naïve Bayes classifier, the best result is 78.4% of F-measure on Hamshahri dataset. Using hidden Markov model (HMM) for topic identification is proposed in (Gharavi and Veisi, 2014) for Persian and they have achieved 79% of F-measure on 8 classes Bijankhan dataset.

Table 1. Some of the research-based Persian document classifications

Reference	Feature Extraction	Classifier	Dataset	Best Result (F-Measure)
(Pilevar et al., 2009)	Codebook	Neural Network (LVQ3)	Hamshahri 2	89.9%
(Elahimanesh et al., 2012)	N-Gram	KNN	Hamshahri	93.9%
(Yari et al., 2010)	Selected Terms	Fuzzy Relation (Fuzzy Model)	Hybrid Dataset	91%
(Maghsoodi and Homayounpoor, 2011)	DF	SVM	Bijankhan	94%
(Jafari et al., 2011)	MI	Naïve Bayes	Hamshahri	78.4%
(Gharavi and Veisi, 2014)	Selected Terms	HMM	Bijankhan	79%

The researches for text classification have used various kind of features and feature selection methods, as well. Several researches used classic feature selection method like, Term Frequency-Inverse Document Frequency (TF-IDF)

(Farhoodi and Yari, 2010), Document Frequency (DF) (Basu and Murthy, 2012; Maghsoodi and Homayounpoor, 2011), chi-square statistics (Basu and Murthy, 2012), Mutual Information (MI) (Saraee and Bagheri, 2013), information gain (Parchami et al., 2012). In addition, there are other feature selection methods such as DFS (Zong et al., 2015) that will be described in this paper.

A summary of some Persian document classification works with their best results is shown in Table 1.

3. Proposed Text Classification Method

In this section, our proposed text classification method[2] is described. The method starts with the pre- processing module including stop words removal. Afterward, the discriminative feature selection method will be introduced alongside features semantic similarity to documents. The motivation for using discriminative feature selection and semantic similarity is that the mentioned combination reports superior performance for English document classification (Zong et al., 2015). In addition, we propose incorporating variance of fuzzy similarities for a document to available categories in feature calculation. At the end of this section, the architecture of the method will be presented.

3.1. Pre-processing

It is shown (Aggarwal and Zhaei, 2012) that pre-processing improves the classification performance in most evaluations. Before the dataset is entered into the proposed model, pre-processing data is first performed, this process is needed to prepare the data to be able to further processed by the algorithm and to increase accuracy by minimizing bias and noise caused by non-basic words, unimportant terms (Rismanto et al., 2020). This phase includes text normalization, tokenization, and stop words removal. Normalization and tokenization in Persian require several considerations including incorrect encoding of some characters (especially for letters 'ك' /k/ and 'ى' /i/ or /y/ that are encoded in Arabic), word boundary problem and using space and pseudo-space (i.e., zero-width non-joiner) optionally instead of each other (or even sometime omitting the space) (Bijankhan et.al., 2011). Stop words, refer to a set of frequent words such as determiners (e.g., the), prepositions (e.g., of) and conjunctions (e.g., and) that taking them into the process of classifying the documents is unimportant and worthless. Because of the high frequency of such words in any natural language documents, there is a motivation to remove this type of words. Stop words, are not informative tokens in text classification, therefore, ignoring them doesn't harm the classification process results.

3.2. Discriminative Feature Selection

After pre-processing, feature extraction is performed in which numbers of terms (i.e., tokens) are extracted from the document. The most common features of a text document that used for text processing algorithms are the term (i.e., words). However, the initial extracted features from text documents are mostly in high dimension and mainly sparse. Putting these features into the calculations of classification will make the future proceedings heavier and harder. To reduce the size of feature space, feature selection methods are used which save time and space. Discriminative feature selection (DFS) (Zong et al., 2015) has been selected to be used in our research as a part of feature selection. This method uses the distribution of features to determine how much every feature can distinguish the categories from each other. There are some features that are frequently repeated in a small number of documents belong to a specific category. These features have a low value in chi-square (x^2) statistic feature selection method, but in DFS they are weighted as valuable features. In the DFS method, the features which have higher weights in the category they belong to than other categories that they do not belong to, are called the discriminative features. For more clarity, the contingency table of feature t_i and category c_j is created that is shown in Table 2 (Zong et al., 2015).

Table 2. Contingency table of feature t_i and category c_j (Zong et al., 2015)

	Containing feature t_i	Not containing feature t_i
	(t_i)	$(\bar{t_i})$
In category c_j (c_j)	a_{ij}	b_{ij}
Not in category c_j ($\bar{c_j}$)	c_{ij}	d_{ij}

In this table, a_{ij} is the number of documents that contain feature (i.e., term) t_i in category c_j, b_{ij} is the number of documents that do not contain feature t_i in category c_j, c_{ij} defines the number of documents that contain feature t_i but are not belong to the category c_j and d_{ij} denotes the number of documents that do not contain feature t_i and are not belong to the category c_j. The goals of the DFS method are 1) choosing the features that have higher average term frequency in category c_j, these features are better in representing the category c_j owing to their high representing probability; 2) selecting the features that their occurrence rate in the majority of documents in the category c_j is higher than the others, these features based on previously mentioned reason are better than the others, too; and 3) disregarding the A number which occurred in most of the documents in categories c_j and $\bar{c_j}$, because these features can't distinguish

the categories properly. Based on these purposes, the DFS method formula is expressed below in equation 1 (Zong et al., 2015).

$$DFS(t_i, c_j) = \frac{tf(t_i, c_j)/df(t_i, c_j)}{tf(t_i, \overline{c}_j)/df(t_i, \overline{c}_j)} \times \frac{a_{ij}}{(a_{ij}+b_{ij})} \times \frac{a_{ij}}{(a_{ij}+c_{ij})} \times \left| \frac{a_{ij}}{(a_{ij}+b_{ij})} \times \frac{c_{ij}}{(c_{ij}+d_{ij})} \right| \tag{1}$$

Where $tf(t_i, c_j)$ is the frequency of the term t_i in category c_j, $tf(t_i, \overline{c}_j)$ presents the term frequency of feature t_i in other categories except c_j, $df(t_i, c_j)$ is the number of documents that contains feature t_i in category c_j and $df(t_i, \overline{c}_j)$ denotes the number of documents that contains feature t_i in other categories except c_j. After computation of the DFS values of each term to the available categories, the feature value for each term is calculated as the maximum of obtained feature DFS values over all categories as shown in equation 2.

$$DFS(t_i) = \max_{1 \le j \le C}\{DFS(t_i, c_j)\} \tag{2}$$

So far, all terms that are extracted from documents, have a DFS value. Next, the terms are sorted from the highest value to the lowest value based on their DFS value to select terms with the best ranking. Now, with the selected terms from the DFS method, the initial feature vectors are formed. The weight of each feature (term) in a feature vector would be then computed from Term Frequency-Inverse Document Frequency (TF- IDF) (Jones, 2004; Lan et. al., 2009) weighting method as presented in equation 3.

$$w_{m,k} = \frac{tf(t_m, d_k) \times log(\frac{N}{n_m}+0.01)}{\sqrt{\sum_{m=1}^{n}(tf(t_m, d_k) \times log(\frac{N}{n_m}+0.01))^2}} \tag{3}$$

In this equation, N is the number of train documents, t_m means mth feature, n_m represents the number of documents that include feature t_m, n is the number of features that are selected from the DFS method, d_k denotes the document that is attended and $w_{m,k}$ is the TF-IDF weight of the feature t_m in document d_k.

After DFS feature selection, the next step is semantic similarity calculation which is a kind of similarity between the feature and document.

3.3. Semantic Similarity

The semantic similarity used in this paper is a type of similarity that has been used in information retrieval (Zong et al., 2015) and only uses available documents without the need to use other resources such as corpora. Thus, it has a different meaning with the other semantic similarity analysis definitions that use external resources such as dictionaries and WordNets. In this sense, features are related or semantically similar if they are in a common document, subsequently, the documents that include common features are similar in the same way (Carpineto and Romano, 2012). In the beginning, for finding a similarity value of each feature (term) to its document, we have to compute the similarity values between the features. For this aim, the similarity value between mth feature and nth feature is calculated by equation 4 (Zong et al., 2015) in which $w_{m,k}$ and $w_{n,k}$ are the TF-IDF weights defined in equation 3.

$$Sim(t_m, t_n) = \frac{\sum_{k=1}^{N} w_{m,k} \times w_{n,k}}{\sqrt{\sum_{k=1}^{N} w_{m,k}^2 \times \sum_{k=1}^{N} w_{v,k}^2}} \tag{4}$$

Now, the similarity of feature t_i to the document d_k can be obtained from equation 5.

$$Sim(t_i, d_k) = \sum_{\substack{j=1 \\ t_j \in d_k}}^{n} w_{j,k} \times Sim(t_i, t_j) \tag{5}$$

Then, the initial feature vectors should be modified by adding new semantic weights. The modified weight of each feature in the vector calculated from equation 6.

$$Modified(w_{m,k}) = TFIDF(w_{m,k}) + Sim(t_m, d_k)/\sum_{j=1}^{n} w_{j,k} \tag{6}$$

$Modified(w_{m,k})$ is the modified weight of the feature t_m in document d_k. The fuzzy similarity is the second concept of the similarity which exists between the document and category, it will be under consideration for the next step.

3.4. Variance of Fuzzy Similarity

Fuzzy similarity calculates the relationship of a document to all available categories based on the presence amount

(i.e., a membership value between 0 and 1) of the query document features among the train documents categories. Thus, with the aim of computing the features membership amounts, we have to create the feature-category matrix (Widyantoro and Yen, 2000). The collection of train documents is already labeled corresponding to their predetermined categories. Similar to Fig 1, the number of occurrences of each feature (term) in the available categories is specified. In this Fig (left hand), the occurrences are calculated for four features (terms) f_i using statistics of four documents d_i in a four categories task. The membership values are then calculated as shown in Fig 1.b.

Docs	Features				Categories
	f_1	f_2	f_3	f_4	
d_1	1	0	3	4	c_1
d_2	0	2	0	0	c_2
d_3	1	0	1	7	c_3
d_4	2	0	3	0	c_4

a) Statistics of features in documents

Features	Categories			
	c_1	c_2	c_3	c_4
f_1	0.12	0	0.38	0.5
f_2	0	1	0	0
f_3	0.11	0	0.11	0.78
f_4	0.4	0	0.6	0

b) Membership value of features for each category

Fig.1. The feature-category matrix for calculating fuzzy similarity

The terms here are the same features that are selected by the DFS method. After the creation of the feature-category matrix, it is quite clear that every feature belongs to which category and how is its degree of membership. This membership value is calculated by counting the occurrence number of the feature in a specific category divided by the frequency of the same feature in all categories. Thus, all the membership values are between 0 and 1 (Widyantoro and Yen, 2000; Saracoglu et al., 2007).

After constructing the feature-category matrix, we need a method to calculate the similarity of each feature to each category. To do this, the similarity of the query document to the existing categories must be calculated from the fuzzy similarity equation that is shown in equation 7 (Widyantoro and Yen, 2000).

$$FuzzySim(d, c_j) = \frac{\sum_{t \in FeatureSet} \mu_R(t, c_j) \otimes \mu d(t)}{\sum_{t \in FeatureSet} \mu_R(t, c_j) \oplus \mu d(t)} \tag{7}$$

In this equation, d is the query document, c_j represents the intended category, t is every feature that selected by DFS method, $\mu_R(t, c_j)$ expresses the membership value of feature t in category c_j and $\mu_d(t)$ explains a kind of membership value for feature t in document d which is defined as equation 8. In this equation, w_t represents the frequency of feature t in document d and $\max_{t \in Features} Set\{w\}$ is the maximum of frequencies of the available features in the same document (Widyantoro and Yen, 2000).

$$\mu_d(t) = \frac{w_t}{\max_{t \in FeatureSet}\{w\}} \tag{8}$$

Also, $\otimes$ and $\oplus$ are algebraic fuzzy t-norm and t-conorm that are shown in equations 9 and 10, respectively.

$$x \otimes y = x \cdot y \tag{9}$$

$$x \oplus y = (x + y) - (x \cdot y) \tag{10}$$

So far, we have the obtained fuzzy similarities of each document to all existing categories. Then we propose to use the variance of these similarities to be used in the feature extraction. The variance is computed as inequation 11.

$$VarianceWeigh(d) = Variance(FuzzySim(d, c_1), \ldots, FuzzySim(d, c_c)) * R \tag{11}$$

In this equation, R is a kind of scaling for a variance measure that its typical values are 10^2 or 10 depending on the values intervals. After all, by combining equations 6 and 11, the final proposed weight of the feature t_m in document d_k will be formed as equation 12.

$$Proposed(w_{m,k}) = Modified(w_{m,k}) + VarianceWeight(d_k) \tag{12}$$

This means that we are adding the variance weight of document d to all of feature's values of this document. Now, the final feature vectors are getting ready to be classified with an appropriate classifier.

A. Effect of Variance on Vector Space

For better understanding that what would variance weight do with feature vectors, we have done some evaluations in a designed experiment. To do this, a two-dimensional feature vector is considered to realize the feature space graphically. In these evaluations, 20 documents are selected randomly from two categories, 10 documents from category A and 10 documents from category B from ISNA Persian Dataset which is introduced in Section 4. For each document two features (i.e., two terms) are selected based on the DFS values. It means that in each experiment, two terms with higher DFS values are selected as the feature.

In the first experiment, sample documents are selected from two categories, Political (category A) and Industry (category B). In Fig 2, the values of feature vectors are demonstrated before and after adding the variance weight, i.e., equations 6 and 12, respectively. As it is shown, this proposed modification has resulted in higher discrimination between two classes. The variance weights in category A, on average have higher values in comparison with the variance weights of category B. This fact has resulted in separating the categories and therefore improving the classification performance.
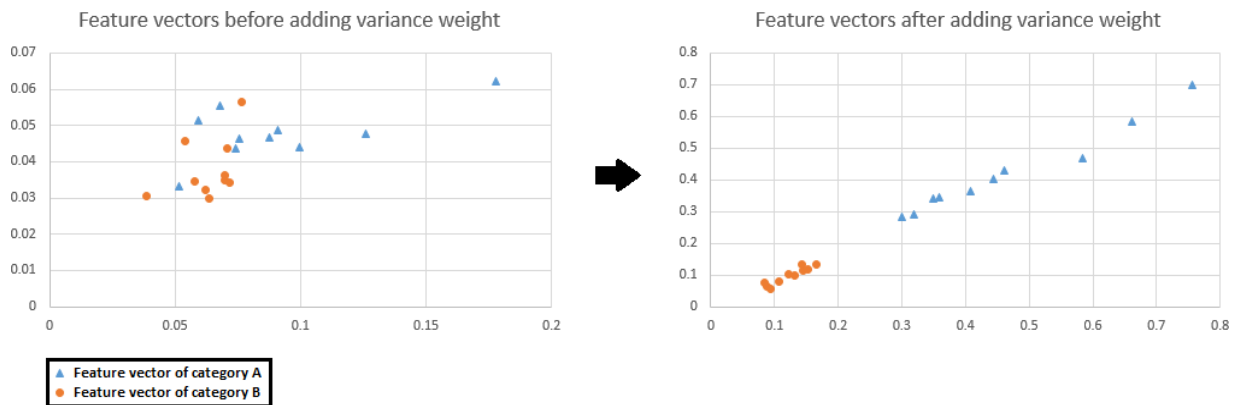

Fig.2. Class discrimination effect of the proposed method in feature vector space for categories Political and Industry

The second experiment is repeated in a similar condition like the first experiment but for categories Technology (denoted as A) and Economy (shown as B). The space vector diagram is given in Fig 3. As it is shown, the proposed features have effectively separated the samples of two classes.
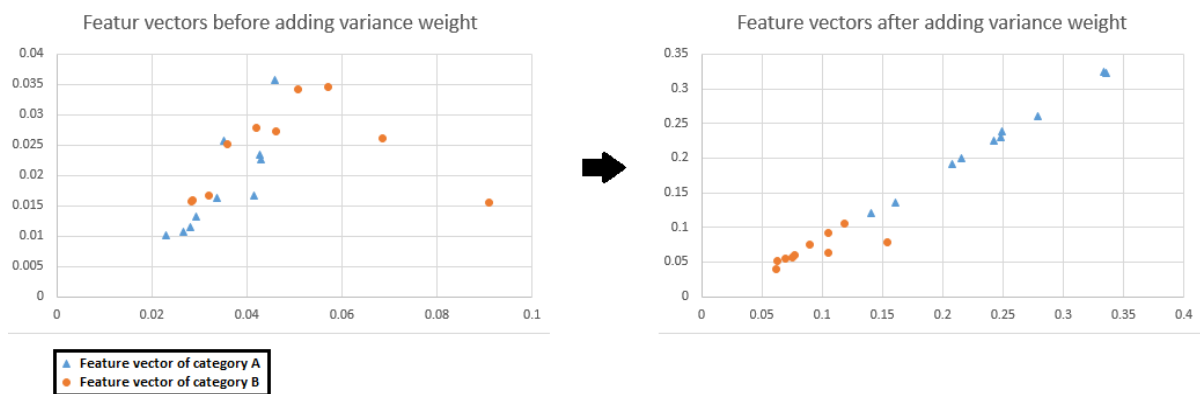

Fig.3. Class discrimination effect of the proposed method in feature vector space for categories Technology and Economy

In the third experiment, sample data are taken for categories Social (A) and Cultural (B). In this case, the variance weights of category A almost are similar to variance weights of category B as shown in Fig 4.

Although the samples of two classes are separated in a way to be classified better than the original features, the discrimination power of the proposed features in this experiment are shown less than the previous experiments.

The effectiveness of the proposed method is highly dependent on the variance values. If the variance values for the two classes are similar, the variance will not result in class discrimination. In the fourth experiment, we have demonstrated this fact. In this case, we take samples of categories Religious (A) and Cultural (B) as shown in Fig 5.

Due to the similarity of variance values for category A and category B, the effect of applying proposed method is the similar movement of the samples in the feature space which it means no significant improvement in finding better decision boundary.

Also, there are cases in which the feature space of the two classes is discriminant before using the proposed method.

The following experiment shows an example of such cases in which sample are taken from categories Political (A) and Sport (B). A sample is shown in Fig 6 in which using the proposed method has not corrupted the class discrimination.
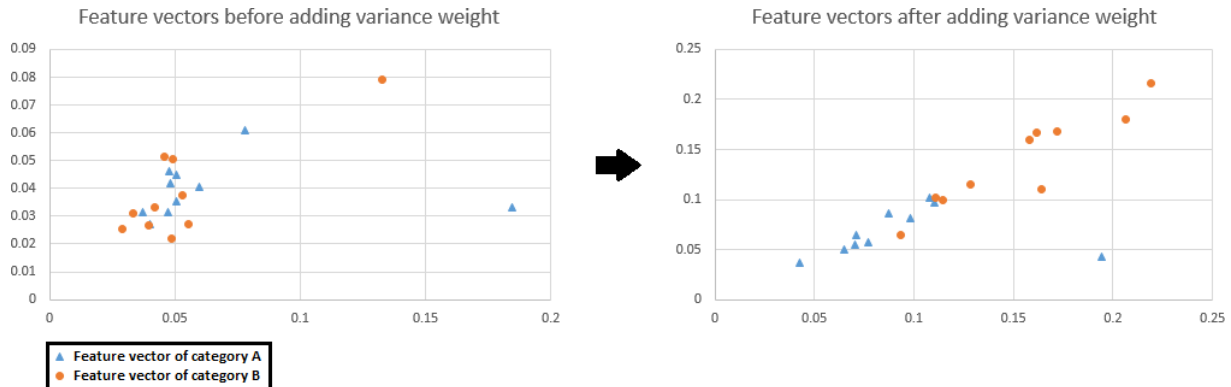

Fig.4. Class discrimination effect of the proposed method in feature vector space for categories Social and Cultural
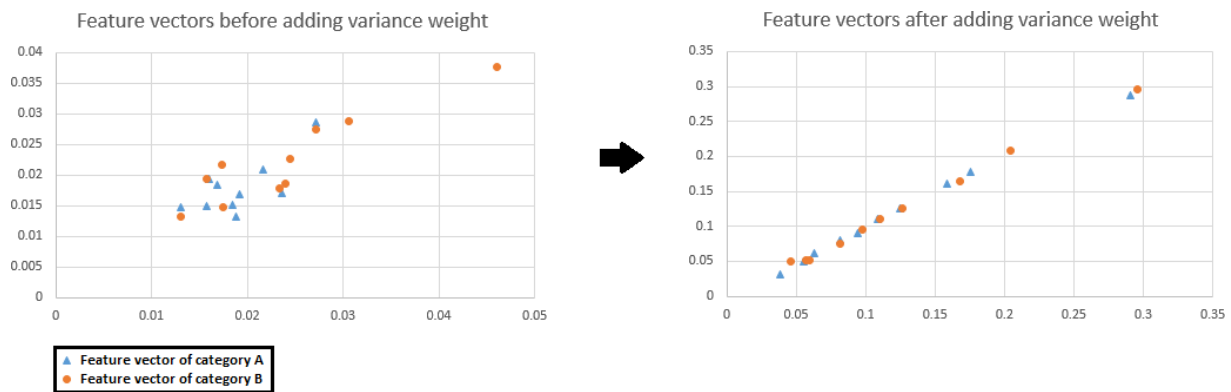

Fig.5. Effect of using the proposed method in feature vector space for categories Religious and Cultural (similar variances for two classes)
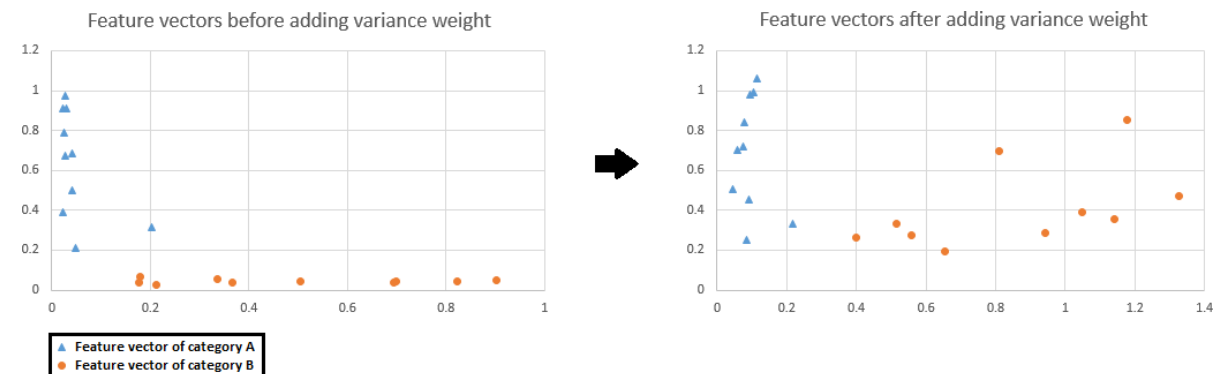

Fig.6. Effect of using the proposed method in feature vector space for categories Political and Sport (classes are well discriminated before using variance)

3.5. Classifier

After selecting the features, a classification method needs to be used in the next step. Although various classification techniques are used in document classification (Pilevar et al., 2009; Aggarwal and Zhaei, 2012; Farhoodi and Yari, 2010), SVM almost has resulted in higher performance among the other methods (Harish et al., 2010). For English and Persian document classification in this paper, SVM and ANN are chosen to be used as the classifiers.

SVM has a capability of independent learning in multi-dimensional feature space. This method is a supervised classification technique that finds the best decision boundary between two classes. The core of this algorithm creates a hyperplane (y = 0) between samples of the positive class $L_1(y = +1)$ and negative class $L_2(y = −1)$. It maximizes the distance between boundary plates of each class. Documents or samples in the maximum distance of classification hyperplane are called support vectors. The main problem with SVM is high training time as well as high memory consumption (Aggarwal and Zhaei, 2012). In this paper, SVM is applied for the classification using Weka tool (Hall et al, 2009). In our evaluations, polynomial kernel function and other default parameters of Weka are used.

The second classifier used in the paper is a neural network. A neural network is a mathematical model that

simulates the structure and processes of the human brain. A common neural network used for classification usually involves an input layer, an output layer and a small number of hidden layers. The input layer is the same features extracted from the sample documents; the output layer represents different categories of a classification (Chen et al., 2012). The model of a neural network that has been used in this study is a 3-layer Multi-Layer Perceptron (MLP) (Aggarwal and Zhaei, 2012). In our simulations, sigmoid and softmax activation functions are used for the hidden and output layer neurons, respectively. A number of hidden neurons are chosen between 100 and 350 which is discussed in detail in Section 4.

3.6. Proposed Classification Method Architecture

In this section, a summary of the mentioned techniques is given as a step-by-step flowchart of the proposed method. The architecture is presented in Fig 7. The steps shown in this Fig, are similar for both train and test phases. It needs to be noted that in both phases, semantic and fuzzy similarities of documents are calculated toward the train documents.
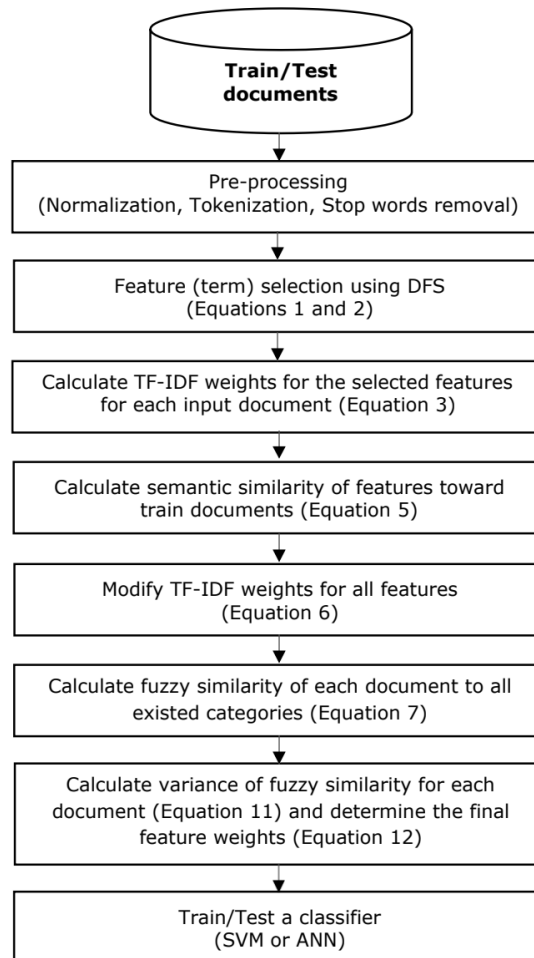

Fig.7. The proposed document classification architecture

4. Evaluation Results

To evaluate the proposed method, in addition to Persian document classification, our evaluations are also done for English. In this section, Persian and English datasets are described and the evaluation metrics are reviewed. Then, evaluation results are given for Persian and English using the proposed method and reference methods.

4.1. Datasets

The datasets used in the evlauations and measuring the performance of the proposed classification method are one dataset in Persian and two datasets in English. The details of each dataset are given in this section.

A. ISNA Persian Dataset

This dataset consists of 4,111 documents belonging to eight newsgroups and 26,073 unique words (terms). This data is collected from the Iranian Students News Agency (ISNA) news agency website[3]. This dataset has an almost

equal distribution of documents in all categories. The details of the training and test subsets of this dataset are shown in Table 3.

Table 3. Documents distribution of Persian ISNA dataset

Category	Number of training documents	Number of test documents
social	355	168
economic	362	171
religious	290	138
political	352	167
technology	362	172
cultural	358	169
sport	354	165
industry	358	170
Totla number of documents	2,791	1,320

B. Reuters English Dataset

The Reuters dataset contains 7,674 documents belonging to eight categories gathering from Reuters news agency (Dobbins et al., 1987). It is the most frequently used dataset in English text processing and document classification. The distribution of documents between categories is unequal. In the training subset, the size of "Earn" category which is the largest category, is more than 70 times larger than the smallest category (i.e., Grain). Details of the distribution of documents between training and test sets for Reuters dataset are shown in Table 4.

Table 4. Documents distribution of English Reuters-21578 dataset

Category	Number of training documents	Number of test documents
Acquisition	1,596	696
Crude	253	121
Earn	2,840	1,083
Grain	41	10
Interest	190	81
Money-fx	206	87
Ship	108	36
Trade	251	75
Totla number of documents	5,485	2,189

C. 20-Newsgroup English Dataset

This dataset consists of the 4,595 news documents in 20 categories (Mitchell, 1997). There are many types of research in the field of English document classification that applied this dataset (Zong et al., 2015; Harishet al., 2010; Lan et al, 2009). This set has a uniform distribution of documents in 20 categories. Details of the training and test sets of 20-newsgroup are given in Table 5.

4.2. Evaluation Metrics

In order to introduce the evaluation metrics, the variables of Table 6 are used. In this paper, the following evaluation metrics are used:

- Recall: It defined as the percentage of correct classified documents among all the documents that must be assigned to the target category. According to the variables of Table 6, $Recall(c_j)$ for category j is determined as in equation 13.
- Precision: It is the percentage of correct classified documents among all the documents that have been assigned to the target category. This metric is shown by $Precision(c_j)$ in equation 14 for category j.
- F-measure: That is a harmonic mean of $Recall(c_j)$ and $Precision(c_j)$, and is presented by $F-Measure(c_j)$ in equation 15.

$$Recall(c_j) = \frac{A_j}{A_j + B_j} \tag{13}$$

$$Precision(c_j) = \frac{A_j}{A_j + C_j} \tag{14}$$

$$F - Measure(c_j) = \frac{2P(c_j) \times R(c_j)}{P(c_j) + R(c_j)} \qquad (15)$$

Table 5. Documents distribution of English 20-newsgroups dataset

Category	Number of training documents	Number of test documents
alt.atheism	119	80
comp.graphics	139	93
comp.os.ms-windows.misc	138	96
comp.sys.ibm.pc.hardware	145	97
comp.sys.mac.hardware	140	94
comp.windows.x	143	96
misc.forsale	140	94
rec.autos	145	97
rec.motorcycles	146	100
rec.sport.baseball	145	95
rec.sport.hockey	146	96
sci.crypt	149	99
sci.electronics	144	95
sci.med	147	97
sci.space	146	96
soc.religion.christian	146	97
talk.politics.guns	136	89
talk.politics.mideast	138	90
talk.politics.misc	115	76
talk.religion.misc	90	61
Totla number of documents	2,757	1,838

Table 6. The possible occurrences of a test document category after the classification process

Is it labeled in the mentionedcategory? Yes		Is it labeled in the mentionedcategory? No
Is it categorized in the mentionedcategory? Yes	A_j	C_j
Is it categorized in the mentionedcategory? No	B_j	D_j

Accordingly, the weighted average of F-measure for all categories is then derived from equation 15 as shown in equation 16, in which N_j is the number of documents that are labelled in c_j category, c is the numberof all categories.

$$F_{weighted} = \frac{1}{\sum_{j=1}^{c} N_j} \times \sum_{j=1}^{c} F(c_j) \times N_j \qquad (16)$$

4.3. Experimental Results

In this section, evaluation results on Persian and English text classification are given. In our experiments, in addition to our proposed method, two classification methods based on discriminative features are used as the reference techniques. The methods are:

- **The proposed method (DFS+SSIM+FSIM):** As described, this classification method is based on a discriminative feature selection method by involving both semantic and variance of fuzzysimilarities.
- **Zong's method (DFS+SSIM):** This classification method is based on a discriminative feature selection method by involving only the semantic similarity (Zong et al., 2015).
- **The DFS method (DFS):** This classification method is only based on a discriminative feature selection method (Zong et al., 2015).

Although the main goal of the research is Persian text classification, two English datasets are also used in the evaluations to show the effectiveness and generality of our method. To evaluate the proposed and the reference methods, SVM and MLP classifiers with a different number of features are employed. Table 7 presents Persian classification results based on SVM classifier. In this table (and other tables of this section), F, R, and P denote F-Measure, Recall, and Precision, respectively. According to the results of this table, the superiority of the proposed method compared to the other classification methods are evident for all metrics.In the best case that is for 3,000 features, the proposed method has 3.3% absolute improvement of weighted average F-Measure against Zong's method. Also, it can be seen that for the reference methods, DFS+SSim has resulted in higher performance than DFS. The higher performance of the proposed remains consistentlyfor all feature sizes.

Table 7. Weighted average metrics of Persian text classification on ISNA dataset with different number of features based on SVM classifier

Method	Weighted Average Metrics	500 Features	1000 Features	2000 Features	3000 Features	4000 Features	5000 Features
DFS+SSim+FSim (Proposed)	**F**	**78.4**	**80.3**	**81.2**	**82.0**	**81.6**	**81.9**
	R	**78.5**	**80.3**	**81.1**	**82.0**	**81.6**	**81.9**
	P	**78.6**	**80.5**	**81.4**	**82.3**	**81.8**	**82.2**
DFS+SSim (Zong et al., 2015)	**F**	77.3	77.8	77.8	78.7	78.2	78.9
	R	77.3	77.7	77.7	78.7	78.2	78.9
	P	77.4	77.9	78.0	78.9	78.6	79.1
DFS (Zong et al., 2015)	**F**	76.8	77	77.7	77.9	78.3	78.8
	R	76.7	76.9	77.6	77.8	78.3	78.8
	P	77.0	77.2	78.0	78.2	78.6	79.1

Persian classification results in F-measure for MLP neural network classifier are shown in Table 8. Using this classification method also demonstrates the higher performance of the proposed method in comparison with the other method (DFS+SSim) for all feature sizes. As it seems, the best result 80.1% is observed in 2000 features which are 2.8% higher than the reference. Also, the maximum distance between the F- measures of two methods is 5% that occurred for 3000 features.

Table 8. Weighted average F-Measure of Persian text classification on ISNA dataset with different number of features based on neural network MLP classifier

Method	500 Features	1000 Features	2000 Features	3000 Features	4000 Features	5000 Features
DFS+SSim+FSim (Proposed)	**76.9**	**78.6**	**80.1**	**79.2**	**76.1**	**78.2**
DFS+SSim (Zong et al., 2015)	76.4	76.4	77.3	74.2	73.5	75.8

Comparing the results of Tables 9 and 10 shows that SVM is superior to ANN. Therefore, for English document classification, we only have reported SVM results. F-measure values for English using SVM classifier, are reported in Tables 11 and 12. In Table 9, document classification results on Reuters dataset are given. Similar to the Persian language, the results of this table show the superiority of the proposed method in comparison with the other methods. In the best case (97.8%), the performance distance betweenthe proposed method and Zong's method is 1.4%.

Table 9. Weighted average F-Measure of English text classification on Reuters-21578 dataset with different number of features based on SVM classifier

Method	500 Features	1000 Features	2000 Features	3000 Features	4000 Features	5000 Features
DFS+SSim+FSim (Proposed)	**97**	**97.3**	**97.2**	**97.3**	**97.6**	**97.8**
DFS+SSim (Zong et al., 2015)	96.8	96.7	96.3	96.2	96.2	96.4
DFS (Zong et al., 2015)	96.1	95.8	95.9	96.3	96.1	96.4

In Table 10, document classification results on 20-newsgroups are presented. From the results, the proposed method outperformed the reference methods in all features number. The best result is for 5000 features (76.5%) in which the F-measure of the proposed method is 2.5% higher than the Zong's method. It is noticeable that the performance of the proposed method using 2000 features is almost equal to the Zong's method using 5000 features.

Table 10. Weighted average F-Measure of English text classification on the 20-newsgroups dataset with different number of features based on SVM classifier

Method	500 Features	1000 Features	2000 Features	3000 Features	4000 Features	5000 Features
DFS+SSim+FSim (Proposed)	**68**	**71.1**	**74.2**	**75.2**	**76**	**76.5**
DFS+SSim (Zong et al., 2015)	66	68.9	71.1	73.3	72.7	74
DFS (Zong et al., 2015)	64.4	66.6	69	69.6	71.1	71.6

5. Summary and Conclusions

In this paper, we addressed the text document classification (i.e., topic identification) problem in Persian. We proposed to use the variance of fuzzy similarity in combination with discriminative feature selection and semantic similarity methods. The variance of fuzzy similarities takes into account the membership variations of each document to different categories. Applying the variance values of fuzzy similarity move the samples of categories in the feature space in a direction that results in higher discrimination between different classes. The proposed method is evaluated for both Persian and English tasks and is compared with two reference methods. The evaluation results showed the consistent superiority of the proposed method in comparison to the reference methods for all feature sizes. The best results obtained from the classification was the weighted average F-measure of %97.8 in English and the weighted average F-measure of %82 in Persian documents.

There are some ideas for future activities to the continuation of this research. During various and different tests on discriminative semantic features, it was obvious that the DFS method shows high flexibility in combination with other techniques and similarities. It is proposed for future works to studying other similarity methods like functional tree similarity (Ankali and Parthiban, 2021) and similarity methods mentioned in (Verma and Aggarwal, 2019) rather than fuzzy similarity. Also, extending the method to handle multi-label document categorization is another research topic.

6. Compliance with Ethical Standards

Conflict of Interest: The authors declare that they have no conflict of interest.

References

[1]   Aggarwal, C.C., Zhai, C.X., 2012. *A Survey of Text Classification Algorithms*. Mining Text Data, Springer.
[2]   AleAhmad, A., Amiri, H., Darrudi, E., Rahgozar, M., Oroumchian, F., 2009. *Hamshahri: A Standard Persian Text Collection.* Knowledge-Based Systems, 22, 382-387.
[3]   Basu, T., Murthy, C.A., 2012. *Effective Text Classification by a Supervised Feature Selection Approach*. Data Mining Workshops (ICDMW), IEEE 12th International Conference, 918-925.
[4]   Bijankhan, M., 2008. *100 Millions Word Farsi Corpus.* Technical Report, Research Center for Intelligent Signal Processing.
[5]   Bijankhan, M., Sheykhzadegan, J., Bahrani, M., & Ghayoomi, M. 2011. *Lessons from building a Persian written corpus: Peykare.* Language Resources and Evaluation, 45, 143-164.
[6]   Carpineto, C., Romano, G., 2012. *A Survey of Automatic Query Expansion in Information Retrieval.* ACM Computing Surveys (CSUR). 44 (1), 1-50.
[7]   Chen, J., Pan, H., Ao, Q., 2012. *Study a Text Classification Method Based on Neural Network Model*. Proceedings of the MSEC International Conference on Multimedia, Software Engineering and Computing, Springer Berlin Heidelberg, 128, 471-475.
[8]   Dobbins, S., Topliss, M., Weinstein, S., Andersen, P., Cellio, M., Hayes, P., Knecht, L., Nirenburg, I., 1987. *Reuters-21578 Text Categorization Collection*. (Available at http://kdd.ics.uci.edu/databases/reuters21578).
[9]   Elahimanesh, M.H., Minaei-Bidgoli, B., Malekinezhad, H., 2012. *Improving K-Nearest Neighbor Efficacy for FarsiText Classification*. The International Conference on Language Resources and Evaluation (LREC),1618-1621.
[10] Farhoodi, M., Yari, A., 2010. *Applying Machine Learning Algorithms for Automatic Persian Text Classification*. 6th International Conference on Advanced Information Management and Service (IMS), 318-323.
[11] Gharavi E., Veisi H., 2014, *A Hidden Markov Model for Persian Text Classification*. 3rd National Computational Linguistics Conference, Tehran-Iran.
[12] Hall, M., Frank, E., Holmes, G., Pfahringer, B., Reutemann, P., Witten, I., 2009. *The Weka Data Mining Software.* ACM SIGKDD Explorations Newsletter 11(1), 10-18.
[13] Hao, P.Y., Chiang, J.H., Tu, Y.K., 2007. *Hierarchically SVM Classification Based on Support Vector Clustering Method and Its Application to Document Categorization*. An International Journal Expert Systems with Applications. 33(3), 627-635.
[14] Harish, B.S, Guru, D.S, Manjunath, S., 2010. *Representation and Classification of Text Documents*: A Brief Review. IJCA Special Issue on Recent Trends in Image Processing and Pattern Recognition.
[15] Jafari, A., Hosseinejad, M., Amiri, A., 2011. *Improvement in Automatic Classification of Persian Documents by Means of Naïve Bayes and Representative Vector*. 1st International Conference on Computer and Knowledge Engineering (ICCKE), IEEE, 226-229.
[16] Jones, K.S., 2004. *A Statistical Interpretation of Term Specificity and Its Application in Retrieval*. J. Document. 60 (5), 493-502.
[17] Ko, Y.J., Park, J., Seo, J., 2004. *Improving Text Categorization Using the Importance of Sentences*. An International Journal of Information Processing and Management. 40, 65-79.
[18] Lan, M., Tan, C.L., Su. J., 2009. *Supervised and Traditional Term Weighting Methods for Automatic Text Categorization*. IEEE Transactions on Pattern Analysis and Machine Intelligence. 31(4), 721-735.
[19] Maghsoodi, N. and Homayounpoor, M., 2011. *Using Thesaurus to Improve Multiclass Text Classification.* Part II, LNCS 6609, 244-253.
[20] Mitchell, T., *1997. The 20 Newsgroups Dataset*. (Available at http://kdd.ics.uci.edu/databases/20newsgroups).
[21] Miyamoto, S., 2001. *Fuzzy Multisets and Fuzzy Clustering of Documents*. In Proceedings of the IEEE International Conference on Fuzzy Systems, FUZZ-IEEE, 3, 1539-1542.

[22] Pilevar, M.T., Feili, H., Soltani, M., 2009. *Classification of Persian Textual Documents Using Learning Vector Quantization*. In IEEE Proceedings of the International Conference on Natural Language Processingand Knowledge Engineering, 1-6.

[23] Qian, T., Xiong, H., Wang, Y., Chen, E., 2007. *On the Strength of Hyperclique Patterns for Text Categorization.* An International Journal Information Sciences. 177, 4040-4058.

[24] Ridwan Rismanto, Arie Rachmad Syulistyo, Bebby Pramudya Citra Agusta, "Research Supervisor Recommendation System Based on Topic Conformity", International Journal of Modern Education and Computer Science(IJMECS), Vol.12, No.1, pp. 26-34, 2020.DOI: 10.5815/ijmecs.2020.01.04

[25] Sanjay B. Ankali, Latha Parthiban, " A Methodology for Reliable Code Plagiarism Detection Using Complete and Language Agnostic Code Clone Classification", International Journal of Modern Education and Computer Science(IJMECS), Vol.13, No.3, pp. 34-56, 2021.DOI: 10.5815/ijmecs.2021.03.04

[26] Saracoglu, R., Tutuncu, K., Allahverdi, N., 2007. A fuzzy clustering approach for finding similar documents using a novel similarity measure. Expert Systems with Applications, 33(3), 600-605.

[27] Saraee, M., Bagheri, A., 2013. Feature Selection Methods in Persian Sentiment Analysis. Natural Language Processing and Information Systems, Springer-Verlag Berlin Heidelberg, 7934, 303-308.

[28] Vijay Verma, Rajesh Kumar Aggarwal, "Accuracy Assessment of Similarity Measures in Collaborative Recommendations Using CF4J Framework", International Journal of Modern Education and Computer Science(IJMECS), Vol.11, No.5, pp. 41-53, 2019.DOI: 10.5815/ijmecs.2019.05.05

[29] Widyantoro, D.H., Yen, J., 2000. *A Fuzzy Similarity Approach in Text Classification Task*. IEEE, Fuzzy Systems, The Ninth IEEE International Conference on, FUZZ-IEEE, 2, 653-658.

[30] Yari, A., Abbasi, A., MomenBellah, S., 2010. *Presenting a fuzzy relation to classify the Persian Web documents.* IEEE International Conference on Intelligent Computing and Intelligent Systems (ICIS), 2, 220-223.

[31] Zong, W., Wu, F., Chu, L., Sculli, D., 2015. *A Discriminative and Semantic Feature Selection Method for Text Categorization*. International Journal of Production Economics, 215-222.

Endnotes:

[1] - http://ece.ut.ac.ir/DBRG/Hamshahri

[2] -https://github.com/pooyanparsafard/Text-Classification-based-on-Discriminative-Semantic-Features-and-Variance-of-Fuzzy-Similarity

[3] - http://isna.ir

Authors' Profiles

Pouyan Parsafard: began his studies in Computer Science at KIAU his BS degree in 2013 and, after learning various programming languages in the first year, he became acquainted with machine learning and data mining. He finished his MS degree in Software Engineering from University of Tehran in 2016. He carried out his research in MS degree at University of Tehran, Data and Signal Processing lab (DSP) under supervision of Dr. Hadi Veisi. There, he developed a strong interest in machine learning, natural language processing and data science.

Hadi Veisi: received his PhD in Artificial Intelligence from Sharif University of Technology in 2011. He joined University of Tehran, Faculty of New Sciences and Technologies (FNST) in 2012 and established Data and Signal Processing (DSP) lab. The main research interests of Hadi are artificial neural network and deep learning, natural language processing, and speech processing.

Niloofar Aflaki: obtained her BS and MS degree in Software Engineering from Islamic Azad University Central Tehran Branch and University of Tehran in 2013 and 2016, respectively. She is currently a PhD Candidate and a Tutor at Massey University, Auckland. Her recent research focuses on the interpretation of geospatial language.

Siamak Mirzaei: received his BSc degree in Computer Software Engineering from Karaj Azad University in 2011 in Iran. He completed his MS degree in Information Technology at the Flinders University of South Australia in 2016. Followed by his Master's, he completed a Graduate Diploma in Research Methods at Flinders University in 2017. He is currently a PhD candidate of the College of Science and Engineering at Flinders University. His research interests include mobile application development, serious games, technology use in education and vocabulary learning/teaching.