Scientia Iranica D (2020) 27(3), 1301–1315

SCIENTIA
IRANICA

Sharif University of Technology
Scientia Iranica
Transactions D: Computer Science & Engineering and Electrical Engineering
http://scientiairanica.sharif.edu

Variance-based features for keyword extraction in Persian and English text documents

H. Veisi[a,*], **N. Aflaki**[b,1], **and P. Parsafard**[b]

a. *Faculty of New Sciences and Technologies (FNST), University of Tehran, Tehran, Iran.*
b. *Kish International Campus, University of Tehran, Kish, Iran.*

Received 10 February 2018; received in revised form 30 December 2018; accepted 28 May 2019

KEYWORDS
Keyword extraction;
Term frequency;
Variance;
Clustering;
Persian text processing.

Abstract. This paper addresses automatic keyword extraction in Persian and English text documents. Generally, to extract keywords from a text, a weight is assigned to each token, and words characterized by higher weights are selected as the keywords. This study proposed four methods for weighting the words and compared these methods with five previous weighting techniques. The previous methods used in this paper include Term Frequency (TF), Term Frequency Inverse Document Frequency (TF-IDF), variance, Discriminative Feature Selection (DFS), and document length normalization based on unit words (LNU). The proposed weighting methods are presented using variance features and include variance to TF-IDF ratio, variance to TF ratio, the intersection of TF and variance, and the intersection of variance and IDF. For evaluation, the documents are clustered using the extracted keywords as feature vectors and by using K-means, Expectation Maximization (EM), and Ward hierarchical clustering methods. The entropy of the clusters and pre-defined classes of the documents are used as the evaluation metrics. For the evaluations, this study collected and labeled Persian documents. Results showed that the proposed weighting method, variance to TF ratio, showed the best performance for Persian texts. Moreover, the best entropy was found by variance to TD-IDF ratio for English texts.

© 2020 Sharif University of Technology. All rights reserved.

1. Introduction

Text mining as a subfield of data mining focuses on extracting useful data and knowledge from textual data. One way to extract knowledge from a document is to access the corresponding keywords. Keywords are important elements that facilitate searching for and obtaining information. They can be considered as a collection of words that describe the document during the search and information retrieval operations. In other words, each important word that describes the content of a document is called a keyword. These words are often used to define and display the information retrieval systems, because they are short and likely to stick in one's mind. Consequently, keywords of a text document, which are the most relevant words concerning the substance of the documents, can be great candidates to be chosen as features in document processing tasks such as classification and clustering [1,2].

Keywords can be assigned to or extracted from a document [3]. In keyword assignment, a set of conceivable keywords is chosen from a controlled vocabulary of words, while keyword extraction distinguishes the pertinent words accessible in the examined docu-

[1] *Present address: Geoinformatics Collaboratory and School of Natural and Computational Sciences, Massey University, Auckland, New Zealand.*

[*] *Corresponding author.*
E-mail addresses: h.veisi@ut.ac.ir (H. Veisi); n.aflaki@ut.ac.ir (N. Aflaki); pooyanparsafard@ut.ac.ir (P. Parsafard)

doi: 10.24200/sci.2019.50426.1685

ment [4]. Moreover, keywords can be categorized into two groups: functional and informative [5]. Informative keywords have a strong connection with the contents of a text and introduce the main content. For example, in the case of sport news about the Barcelona football team, football can be defined as an informative keyword in this news item. On the other hand, functional keywords such as prefixes and conjunctions have a weaker degree of connection with the text [6].

Although there are several methods for keyword extraction [4,7–12] and, also, methods for various languages such as English [13] and Persian [14], finding the appropriate keywords of a document is still a challenging problem. The main focus of the methods is on assigning reliable weights to the keyword candidates and, then, selecting the best ones. This paper proposes new methods for weighting the words and evaluating out methods in Persian and English languages.

In this paper, Section 2 provides a summary of the related works. Section 3 presents a keyword extraction framework that includes the known weighting methods such as variance, Discriminative Feature Selection (DFS), document Length Normalization based on Unit words (LNU), and clustering methods. Section 4 presents the proposed weighting methods, and Section 5 contains experiments and evaluations. The discussions and comparative results are also given in this section. At the end, the conclusion and future works are presented in Section 6.

2. Related works

There are several methods for keyword extraction from a text that can be classified into five general approaches.

2.1. Statistical methods based on Term Frequency (TF) analysis

The first commonly used statistical method is Term Frequency (TF), which calculates the occurrence of a word in a document. Another common method is the Term Frequency Inverse Document Frequency (TF-IDF) that measures the occurrence of a word in a document and all other documents. These approaches, which have been used in this paper, are reviewed in Section 3.1. Statistical methods are well known and are reliable for keyword extraction, because when a word occurs in a document several times, it can be considered a keyword candidate. The statistical keyword extraction techniques can be domain independent and do not require training data [4].

In [8], TF, TF-IDF, and variance were used as weighting methods for document clustering using K-means. They calculated the entropy value to compare the quality of clustered documents with their prior classes. In addition to TF and TF-IDF, there are other statistical methods such as variance and word co-occurrence [13].

2.2. Linguistic methods based on language parsing

The linguistic features of words are used in linguistic methods. Synthetic analysis [9] and lexical analysis [10] are examples of these methods that express the semantic content of a part of the text. Lexical chains, too, are used in text summarization. The recall of synthetic analysis is 66% with 64% precision for lexical chains.

2.3. Machine learning methods such as supervised learning

Keyword extraction is done through a supervised action in this method. In this case, models are trained and keywords extracted by these trained models. Examples of this approach include Naïve Bayes [15] and SVM methods [11]. An ongoing report on keyword extraction presented a model based on fractal patterns [16]. The outcomes demonstrate that the most relevant terms about the topic of a text document have fractal dimensions not quite the same as one, while insignificant terms have a fractal dimension value of one.

2.4. Conceptual methods based on the use of knowledge database to interpret the meaning and concept

This method uses semantic analysis and dictionary [17,18]. In semantic analysis and dictionary as a keyword extraction method, documents are divided into a set of sentences. Later, a model finds the best concept of each term. Then, Lesk algorithm calculates the word-to-word similarity for each pair. Thus, it computes the best concept of each of two words in a pair with their similarity score. Each pair will have clusters based on its similarity score. This is followed by the calculation of the average similarity score for every cluster. In the next step, inverse similarity score is computed to evaluate the importance of a word in a cluster's similarity score. The average of term similarity weights is calculated after removing a term. This step is repeated for all clusters to determine the coherence of clusters.

2.5. Combination of the above approaches

These techniques are a combination of some or all of the approaches to creating a heuristic method, e.g., by using html tags [19]. The heuristic method uses phrase rates, which can be an interactive aid for keyword extraction for human classifiers in informine projects (http://infomine.ucr.edu). This method is a heuristic key phrase extraction for web pages that require no training. Instead, it is based on the hypothesis that most of the well-written web pages offer key phrases based on their inner structure.

3. Keyword extraction framework

To extract keywords, after the stop word removal to do away with all unimportant and meaningless words, the weights are calculated for other terms using the known weighting methods mentioned in Section 3.1 and the proposed methods in Section 4. The terms with higher weights are selected after the extraction of the document features. Then, the quality of the selected terms is evaluated using clustering techniques, as given in Section 3.2. To perform the clustering, document-feature matrices are formed for the selected terms. Three types of such a matrix are used in this paper:

1. *Frequency matrix:* A two-dimensional matrix in which documents are rows and terms are columns and the frequency of every term in every document forms the values of the related cell;
2. *Normal matrix:* This matrix is similar to the frequency matrix in which the values of each cell are the normalized frequency of the related term. The normalization is done by dividing the TF in each document by the maximum TF in that document;
3. Boolean matrix: As the name suggests, the value of each cell in this case is 1 or 0, denoting the presence or absence of that term in the document.

The matrices are formed for a number of features (i.e., terms) and are given to Weka tool [20] for clustering. For clustering, three methods such as K-means, EM, and Ward, as reviewed in Section 3.2, are used. After clustering, the entropy matrix is prepared to evaluate the results. The entropy value is between 0 and 1, with smaller values (i.e., closer to 0) being appropriate. The flowchart of the keyword extraction steps is shown in Figure 1, each of which is described in the following sections.

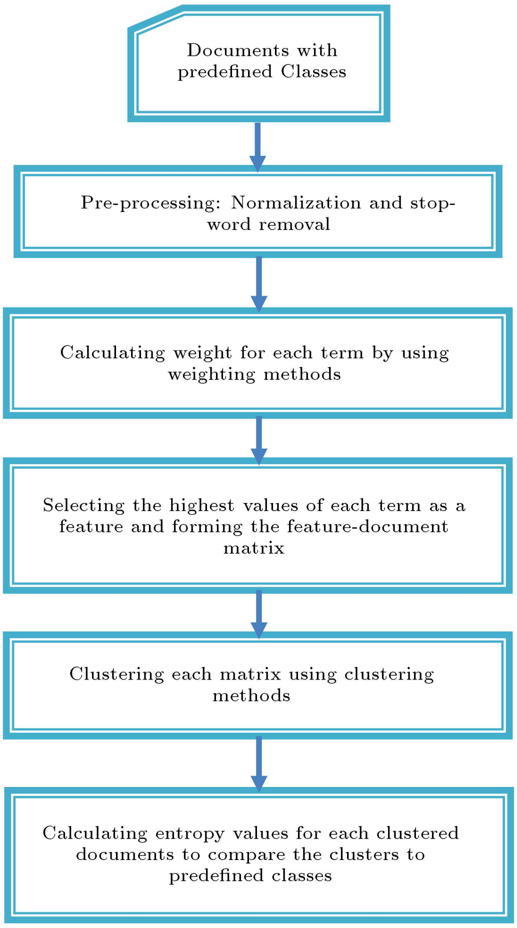

Documents with predefined Classes
Pre-processing: Normalization and stop-word removal
Calculating weight for each term by using weighting methods
Selecting the highest values of each term as a feature and forming the feature-document matrix
Clustering each matrix using clustering methods
Calculating entropy values for each clustered documents to compare the clusters to predefined classes

Figure 1. Keyword extraction and evaluation steps.

3.1. Weighting methods

Term extraction is the first step to extract keywords from a text. The next step is to select features for assigning weights. Weighting of features can be done in various ways with different weighting approaches, yielding different values. This section describes the current weighting methods such as Variance, LNU, and DFS.

3.1.1. TF and TF-Inverse Document Frequency (TF-IDF)

The first statistical method for extracting keywords from a document is TF, which represents the occurrence of a term in a document. If a term occurs more than other terms, it is most likely to be a keyword. Another recent method for keyword extraction is TF-IDF. In this approach, the frequency of using a term is measured in the document concerned and all other documents. The term that occurs in most of the documents has a smaller TF-IDF value because of its lower power of discrimination. Therefore, if a term occurs in fewer documents, its TF-IDF value will be high, indicating that it can identify documents. TF and TF-IDF formulas are shown in Eqs. (1), (2), and (3), respectively.

$$TF(Term_i, Doc_j) = \frac{Number\ of\ times\ Term_i\ appears\ in\ document\ Doc_j}{Total\ number\ of\ terms\ in\ document\ Doc_j}, \tag{1}$$

$$IDF(Term_i) = Log\left(\frac{Total\ number\ of\ documents}{Number\ of\ documents\ with\ Term_i\ in\ it}\right), \tag{2}$$

$$TF-IDF(Term_i, Doc_j) = TF(Term_i, Doc_j) \times IDF(Term_i). \tag{3}$$

3.1.2. Variance

This method calculates the usage variance of each word in a document. *Variance* is equal to the difference between mean square and the average and is calculated through Eq. (4):

$$Variance(Term_j) = \frac{1}{N-1}\sum_{i=1}^{N}(x_i-\mu)^2, \tag{4}$$

where N is the total number of documents containing $Term_j$, μ is the TF average in all documents, and x_i is the occurrence of $Term_i$ in the document.

3.1.3. Document Length Normalization based on Unit words (LNU)

The LNU weighting method is based on the frequency factor (known as the L factor) and the normalization unit (U factor), and word weighting is based on unique words in documents. This relation is expressed in Eq. (5) [21]:

$$LNU(Term_i, Doc_j) =$$

$$\left(\frac{\frac{1+\log(TF(Term_i,Doc_j))}{1+\log(average(TF(:,Doc_j)))}}{(1-slope)\times pivote+(slope\times num\ unique\ terms)}\right), \tag{5}$$

in which *average* $(TF(:,Doc_j))$ is the average of TF for all words in document Doc_j, and slope is the experimental slope of the curve and is often considered as a constant value 0.25. *Pivote* is the ratio of the total number of unique words in all documents to the total number of documents, and *num unique terms* represent the number of unique terms in document Doc_j.

3.1.4. Discriminative Feature Selection (DFS)

Discriminative Feature Selection (DFS) is another feature extraction method proposed for document classification [22]. Here, discriminative features are those with a higher weight in their categories than the others. For extracting these features, the parameters of Table 1 are calculated first.

The DFS will select features having the highest average TF in the mentioned category, picking up those with the highest occurrence rate in most of the documents in the mentioned category. The DFS also does not take those features that occur in most of the documents belonging to both c_j and $\bar{c}_j$ under consideration. Given these descriptions, Eq. (6) is presented for estimating DFS features:

$$DFS(t_i,c_j) = \frac{\frac{TF(t_i,c_j)}{df(t_i,c_j)}}{\frac{TF(t_i,\overline{c_J})}{df(t_i,\overline{c_J})}} \times \frac{a_{ij}}{(a_{ij}+b_{ij})} \times \frac{a_{ij}}{(a_{ij}+c_{ij})}$$

$$\times\left|\frac{a_{ij}}{(a_{ij}+b_{ij})} - \frac{c_{ij}}{(c_{ij}+d_{ij})}\right|, \tag{6}$$

where $TF(t_i,c_j)$ and $TF(t_i,\bar{c}_J)$ show TF of feature t_i in categories c_j and $\bar{c}_J$. $df(t_i,\bar{c}_j)$ and $df(t_i,\bar{c}_J)$ show the number of documents consisting of feature t_i in categories c_j and $\bar{c}_J$. Then, feature t_i has the DFS value in each category, and the final DFS score value for t_i is calculated through the following Eq. (7), which means the features with the highest values.

$$DFS(t_i) = \max_{1\langle j\langle c}\{DFS(t_i,c_j)\}. \tag{7}$$

3.2. Clustering methods

To evaluate the quality of the extracted keywords, the documents are clustered using the selected terms by the weighting techniques. After that, the clusters are compared with the predefined classes using the entropy of the clusters. The clustering methods categorize documents in an unsupervised manner. Herein, three clustering approaches including centroid-based (i.e., K-means) [23,24], distribution-based (i.e., EM: Expectation Maximization) [25], and hierarchical (i.e., Ward) [26] are applied. These methods are selected among a variety of clustering methods due to their popularity in text processing [23,26]. In the following sections, these methods are reviewed briefly.

3.2.1. K-means

K-means is a classic and well-known unsupervised clustering algorithm [23,27]. This method is an easy way to categorize information into a certain number of clusters, i.e., K clusters. The main idea is that K centers are determined for the clusters. The centers must be chosen carefully because each center will produce different results. Therefore, it is better to put them as far apart as possible. To begin with, K-cluster centers are chosen randomly and, in the next step, all points are assigned to the nearest center. Then, all the centers are re-calculated as the mean value of the assigned data to each cluster. The process of assigning the points to the clusters and updating the cluster centers is repeated interactively until the centers show no change. This algorithm aims to minimize the

Table 1. Occurrence number of feature t_i and category c_j [22].

	Document does not have feature $t_i(t_i)$	**Document has feature $t_i(\bar{t}_\imath)$**
Document is in category $c_j(c_j)$	a_{ij}	b_{ij}
Document is not in category $c_j(\overline{c_J})$	c_{ij}	d_{ij}

objective function as a square function of error, shown in the following Eq. (9) [28,29].

$$J=\sum_{j=1}^{k}\sum_{i=1}^{n}||x_i-\mu_j||, \tag{8}$$

where $||x_i-\mu_j||$ is the Euclidean distance between data point x_i and cluster centre μ, n is the number of points in cluster i, and k is the number of cluster centres.

3.2.2. Expectation Maximization (EM)

In conditions without a specific number of clusters, one of the clustering algorithms used is expectation maximization (EM) [25]. It is a computational method for estimating data, particularly hidden data. This algorithm would be suitable for lost data; it can also be an efficient method to calculate the maximum likelihood estimates in repeated computations. The algorithm is related to specific methods of hidden data approximation, which, in these approach parameters, reestimate and continue the process until they converge on a particular value. The name is chosen because, in each repetitive step of the algorithm, there is a phase of expectation and maximization [30].

The probability distribution used in the algorithm is mostly normal distribution, as shown in Eq. (9), because the assumption is that the data could be transformed as a linear sequence from the multivariate normal distribution. EM is an iterative method to calculate parameters.

$$p(x)=\sum_{z}p(z)p(X\,|z\,)=\sum_{k=1}^{K}\pi_k N(X\,|\mu_k,\Sigma_k\,). \tag{9}$$

The connected distribution $p(x,z)$ in terms of a marginal distribution $p(z)$ and an eventual distribution $p(x|z)$, and the marginal distribution over z is specified in terms of the mixing factor π_k, where $p(z_k=1)=\pi_k$. $\sum_k$ is the covariance and $p(x)=\sum_z p(x,z)$ shows that, for every data point X_n, there is a comparable hidden variable z_n.

EM is an iterative method to calculate parameters in two steps: expectation and maximization. In the first step, primary values are given to parameters. Then, in the Expectation step, weighting factor for data point X_n is noted by the posterior probability of $\gamma(z_{nk})$, where parameter k generates X_n, as shown in Eq. (10). In this equation, z_{nk} represents the value of responsibilities, X_n is the data point n, and μ_k is the mean. The following formulas represent EM Gaussian mixture model.

$$\gamma(z_{nk})=\frac{\pi_k N(X_n\,|\mu_k,\Sigma_k)}{\sum_{j=1}^{k}\pi_j N(X_n\,|\mu_j,\Sigma_k)}. \tag{10}$$

The second step is maximization, which calculates the values of the parameters based on the estimated value of $\gamma(z_{nk})$, as shown in Eq. (11), where N_k is the effective number of points assigned to the cluster k.

$$\mu_k^{new}=\frac{1}{N_k}\sum_{n=1}^{N}\gamma(z_{nk})X_n, \quad N_k=\sum_{n=1}^{N}\gamma(z_{nk}), \tag{11}$$

$$\Sigma_k^{new}=\frac{1}{N_k}\sum_{n=1}^{N}\gamma(z_{nk})(X_n-\mu_k^{new})(X_n-\mu_k^{new})^T, \tag{12}$$

$$\pi_k^{new}=\frac{N_k}{N}. \tag{13}$$

The expectation and maximization steps are repeated until the optimal parameter values converge (Eqs. (12) and (13)). Since EM uses the maximum likelihood estimation in each iteration, the following likelihood (Eq. (14)) increases [31]:

$$\ln p(X\,|\mu\,,\Sigma,\pi)=\sum_{n=1}^{N}\ln\left\{\sum_{k=1}^{K}\pi_k N(X_n\,|\mu_k,\Sigma_k\,)\right\}. \tag{14}$$

For simplicity, consider a Gaussian mixture, whose components have covariance Matrices given by $\sum_k=\sigma_k^2 I$, where I is the unit matrix, although the conclusions will hold for general covariance matrices.

Other distributions such as Poisson and log-normal will also be used to improve results. K-means clustering is a special case of EM clustering, too [32].

3.2.3. Hierarchical clustering

In the hierarchical clustering method [33], the tree structure is assigned to the final clusters in accordance with their popularity. This hierarchical tree is called a 'dendrogram'. Hierarchical trees are usually divided into two categories [34,26]:

1. Top-down or divisive: In this approach, initially, all data are considered as a cluster. Then, in every step of the repetition process, the data that are less similar to each other are put in separate clusters. This step continues until the clusters have only one member. Examples of this clustering are bisecting K-means [34];
2. Bottom-up or agglomerative: In this method, each data is considered a separate cluster and, in every step of the repetition process, data that are similar to each other are combined to produce a cluster or a certain number of clusters. The examples of this type of clustering include average link, complete link, and single link [34].

In order to decide which clusters must be merged (for agglomerative), or where a cluster should split (for divisive), it is required to evaluate the dissimilarity between sets. In most hierarchical clustering methods, this is done by the use of an appropriate metric (a measure of the distance between pairs), and a linkage criterion that determines the dissimilarity of sets as a function of the pair-wise distances of observations in the sets. Choosing a proper evaluation metric has a direct impact on the final result. Based on different metrics such as Manhattan, Euclidean, etc., points have different distances from each other, resulting in different forms of clusters and different clustering results [35,36]. Linkage metrics are the same as those that denote distances between sets of points. In hierarchical clustering, there are some linkage metrics such as complete-linkage, minimum-linkage, average-linkage, Ward, etc.

Ward clustering is a method to reduce the loss of remote data [37]. This method uses new criteria to calculate the dissimilarity between clusters. In this process, the difference of square's summation between each data from a cluster to the cluster's mean vector is calculated with the aim of evaluating the cluster. The following algorithm could be considered for Ward [38]:

1. Each data is considered a cluster;
2. For all pairs from a set of clusters, those two clusters whose sum of the squares of differences between the clusters' data to the obtained mean vector is less than the others are going to be selected;
3. Two selected clusters are combined, and a new cluster center is calculated;
4. As long as the number of clusters is not the target number, steps 2 and 3 are repeated.

In Ward's method, the distance between clusters A and B is calculated through Eq. (15):

$$\Delta(A,B)=\sum_{i\in A}||x_i-\mu_{A\cup B}||^2-\sum_{i\in A}||x_i-\mu_A||^2 -\sum_{i\in B}||x_i-\mu_B||^2=\frac{n_A n_B}{n_A+n_B}||\mu_A-\mu_B||^2, \quad (15)$$

where μ_j is the center of cluster j, n_j is the number of points that exists in cluster j, and Δ is the merging cost of two clusters A and B.

4. Proposed weighting methods

This section describes the proposed weighting methods that represent a combination of three known weightings: TF, TF-IDF, and variance. The proposed methods include the variance to TF-IDF ratio, the intersection of TF-IDF and variance, and the intersection of TF and variance.

4.1. The Ratio of Variance to TF-IDF (Var2TF-IDF)

In this method, the variance of TF to TF-IDF ratio of a term is computed. For normalization, the numerator is multiplied by 10^{-5} and denominator by 10^3. This operation is done for all of the terms, and the terms with the highest values are selected as the keywords. This weighting method is calculated through Eq. (16), in which t_i denotes the ith term.

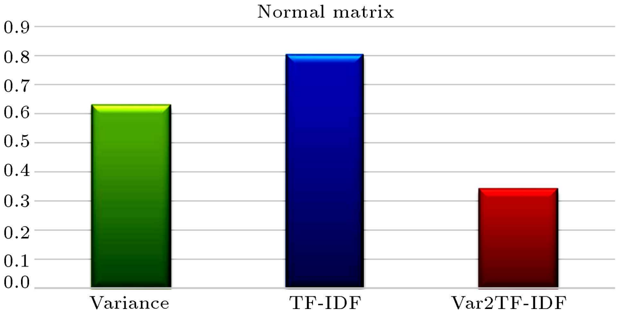


Figure 2. Entropy results of Term Frequency Inverse Document Frequency (TFIDF), variance and proposed variance/TFIDF methods for Ward clustering (5 clusters).

$$Var2TFIDF(Term_i)=\frac{var(TF(Term_i))\times 10^{-5}}{TFIDF(Term_i)\times 10^3}. \quad (16)$$

Figure 2 represents the entropy of document clustering (5 clusters) using variance, TFIDF, and variance to TFIDF ratio feature selection methods. It shows the effectiveness of the proposed method in comparison with the reference methods. Detailed results are given in Section 5.

4.2. The ratio of variance to TF (Var2TF)

In the second proposed method of this paper, after term extraction, the values of variance of TF and TF of a word are calculated in addition to other terms. Then, the ratio of variance to TF is computed. For normalization, the numerator is multiplied by 10^{-5} and the denominator by 10^3. This is done for all the terms, and the terms with the highest values are selected as the keywords. This weighting method is calculated through Eq. (17), in which t_i denotes the ith term:

$$Var2TF(Term_i)=\frac{var(TF(Term_i))\times 10^{-5}}{TF(Term_i)\times 10^3}. \quad (17)$$

Figure 3 represents the entropy of document clustering (5 clusters) using variance, TF, and variance to TF ratio feature selection methods. It shows the effectiveness of the proposed method in comparison with the reference methods. The detailed results are given in Section 5.

4.3. The intersection of variance and TF-IDF (Var ∩ TF-IDF)

In the third proposed method of this paper, the values of features are computed using the variance weighting method and TF-IDF method, as an intersection is

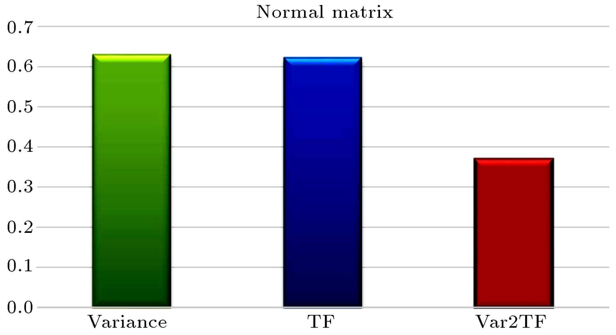


Figure 3. Entropy results of Term Frequency (TF), variance and proposed variance/TF methods for Ward clustering (5 clusters).

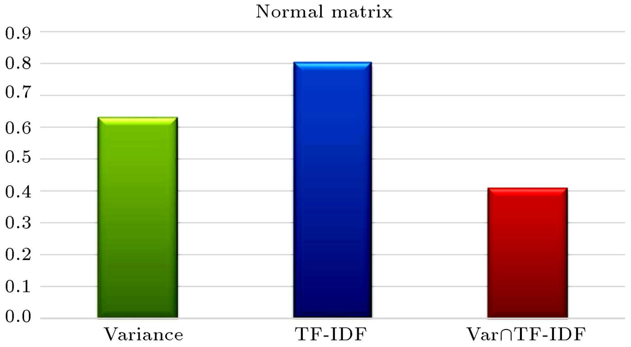


Figure 4. Entropy results of Term Frequency Inverse Document Frequency (TF-IDF), variance, and proposed variance ∩ TF-IDF method with Ward clustering (5 clusters).

made between 150 terms of both methods. Therefore, for common words, a matrix will form.

Figure 4 represents the difference between variance and TF-IDF values and variance ∩ TF-IDF with Ward clustering (5 clusters). The effectiveness of the proposed method is shown in comparison to the reference methods. More detailed results are given in Section 5.

4.4. *The intersection Variance and TF (Var∩TF)*

In the fourth proposed method of this paper, the values of features are calculated using the variance weighting method and TF. Subsequently, an intersection is made between 150 terms of both methods. For common words, a matrix would form.

Figure 5 represents the difference of performance between variance, TF, and Var∩TF with Ward clustering (5 clusters). The effectiveness of the proposed method is shown in comparison to the reference methods. More detailed results are given in Section 5.

5. Experiments and evaluations

In this section, first, the data are explained. Then, the evaluation methods and the results are given in entirety. The comparison between the results is shown at the end of the section. All the evaluations and results are calculated for both English and Persian text documents.

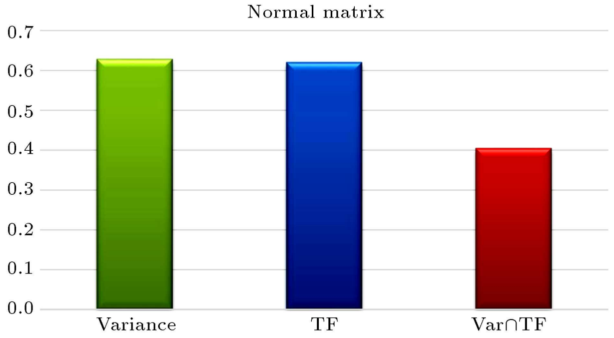


Figure 5. Entropy results of Term Frequency (TF), variance, and proposed variance ∩ TF-IDF method with Ward clustering (5 clusters).

5.1. *Data*

The keyword extraction methods for both Persian and English languages have been applied. For Persian, evaluations are done on 500 documents collected for this research. The Persian data include news (collected from the ISNA website - https://www.isna.ir/) and scientific articles (articles of the Iran Computer Association 2014 - http://csicc2014.sbu.ac.ir/). This dataset has five classes including cultural, medical, sport, information technology, and political categories. The data of classes are balanced, and each class consists of 100 documents.

The English data consist of 500 patent documents [8]. This dataset also has five classes including five patent categories in various subjects: gasification, genetically engineered organisms, solar cells, passive space heating, and wind. Each class has 100 documents.

The documents of these datasets did not have any pre-defined keywords. They were extracted by weighting methods; then, document clustering, based on the keywords, was done to evaluate the weighting and clustering methods.

5.2. *Evaluation method*

As each document in our datasets has a label of its class, the evaluation metric used here is cluster entropy. To calculate entropy, a cluster matrix is calculated in which rows represent the classes and columns represent the clusters. Figure 6 shows the matrix in which X_{ij} is the number of elements in class i and cluster j, N is number of classes, and M is number of clusters.

The purpose of forming this matrix is to calculate entropy. Scattering is reduced as the amount of entropy moves closer to zero, indicating an improvement in the chosen words and clustering. In contrast, if the entropy value is close to 1, the method will be less

$$\begin{array}{c|cccc} & 1 & 2 & \dots & M \\ 1 & X_{11} & X_{12} & \dots & X_{1M} \\ 2 & X_{21} & X_{22} & \dots & X_{2M} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ N & X_{N1} & X_{N2} & \dots & X_{NM} \end{array}$$

Figure 6. Entropy matrix for evaluation.

accurate. Through the mentioned matrix, the entropy is calculated through Eqs. (18) and (19).

$$e(c_j)=\sum\nolimits_i^N\left(-\frac{X_{ij}}{N}\log_N\frac{X_{ij}}{N}\right)$$

$$i=1,2,...,N, \qquad j=1,2,...,M, \tag{18}$$

$$e=\sum\nolimits_j^M\left(\frac{1}{n}\sum\nolimits_i^N X_{ij}\right)e(c_j). \tag{19}$$

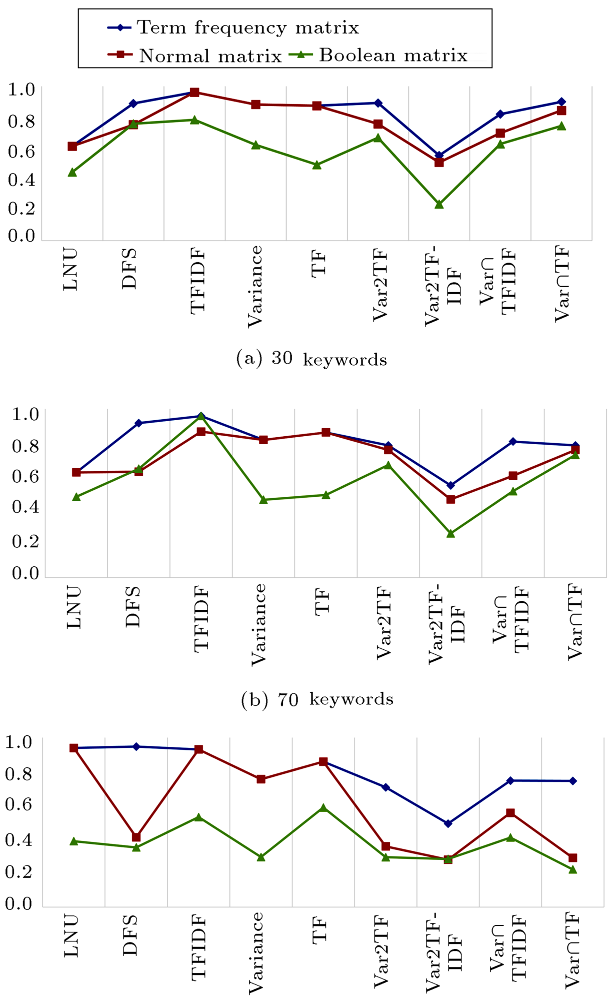
In these equations, n is the total number of documents, $e(c_j)$ is the entropy of cluster j, $w(c_j)$ is the weight of cluster j, and e is the total entropy.

5.3. Keyword extraction for English

5.3.1. Evaluation of K-means clustering

In this section, the results of all the weighting methods by K-means clustering for English documents are presented (5 clusters). After extracting the keywords by the weighting methods mentioned in Sections 3.1 and 4, the most important terms (i.e., keywords with higher weights) are selected. Our evaluations are performed for a different number of terms–30, 70, and 150. For each of these numbers of keywords, three mentioned features, i.e., frequency, normalized frequency, and Boolean are used for clustering. After performing K-means clustering on each feature, the entropy value is calculated to compare the obtained clusters with predefined classes. Figure 7 represents the results.

It is shown that, for 30 features, the best response is the one that relates to the Var2TF-IDF with the Boolean matrix. The worst response is for TF-IDF with frequency and normal matrices. For 70 features, the best response is the same as in 30 features, and the worst response is for the TF-IDF method with Boolean and frequency matrices. In 150 features, the best response is for the proposed method, Var∩TF, and the worst is for TF-IDF with normal and frequency matrices. The experiments in this section show that the proposed methods, Var2TF-IDF and Var∩TF, are better methods for keyword extraction than the reference methods.



(a) 30 keywords

(b) 70 keywords

(c) 150 keywords

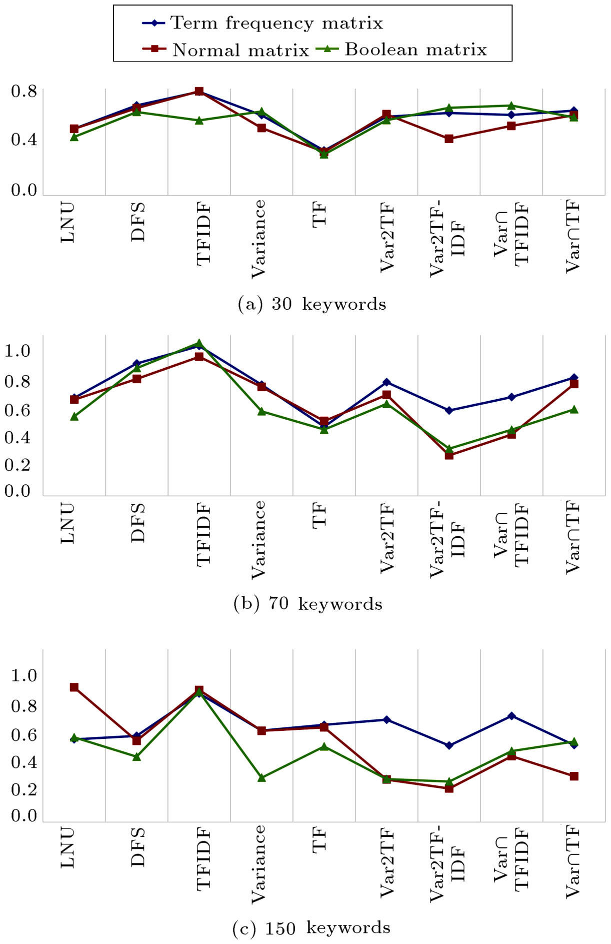
Figure 7. K-means results for different numbers of extracted keywords in English text documents using different methods (5 clusters).

5.3.2. Evaluation of EM clustering

This section represents the results of all weighting methods evaluated by the EM clustering algorithm of English documents (5 clusters). As shown in Section 5.3.1, in this experiment, 30, 70, and 150 features are extracted and EM clustering is applied to them. The results are shown in Figure 8. It is shown that, for 30 features, the best response is the one that relates to TF with all three matrices and the worst response is for TF-IDF with normal and frequency matrices. In 70 and 150 features, the best answers are achieved by the Var2TF-IDF proposed method with the Boolean and normal matrices. In contrast, the worst response in 70 features is for the TF-IDF method with Boolean and frequency matrices and that in 150 features is the one relating to the TF-IDF with all three matrices. As a result, the recommended keyword extraction methods include TF and Var2TF-IDF.

5.3.3. Evaluation of Ward clustering

The results of all weighting methods evaluated by the hierarchical Ward clustering method for English

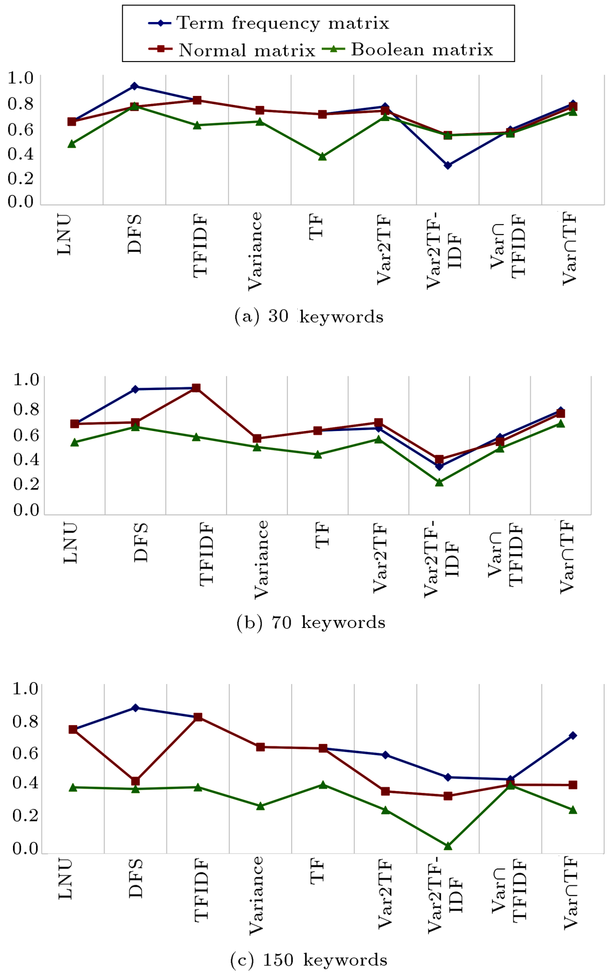


(a) 30 keywords

(b) 70 keywords

(c) 150 keywords

Figure 8. Expectation Maximization (EM) results for different numbers of extracted keywords in English text documents using different methods (5 clusters).



(a) 30 keywords

(b) 70 keywords

(c) 150 keywords

Figure 9. Ward results for different numbers of extracted keywords in English text documents using different methods (5 clusters).

documents are given in Figure 9 (5 clusters). Similar to Sections 5.3.1 and 5.3.2, for this method, too, 30, 70, and 150 features are extracted and Ward clustering is applied to the features. For this number of features, the best response is the one that relates to Var2TF-IDF with frequency and Boolean matrices. In 30 features, the worst performance is the one that relates to DFS with a frequency matrix. In 70 and 150 features, the worst response is the one that relates to DFS and TFIDF with frequency and normal matrices. The experiments in this section define the proposed Var2TF-IDF method for keyword extraction.

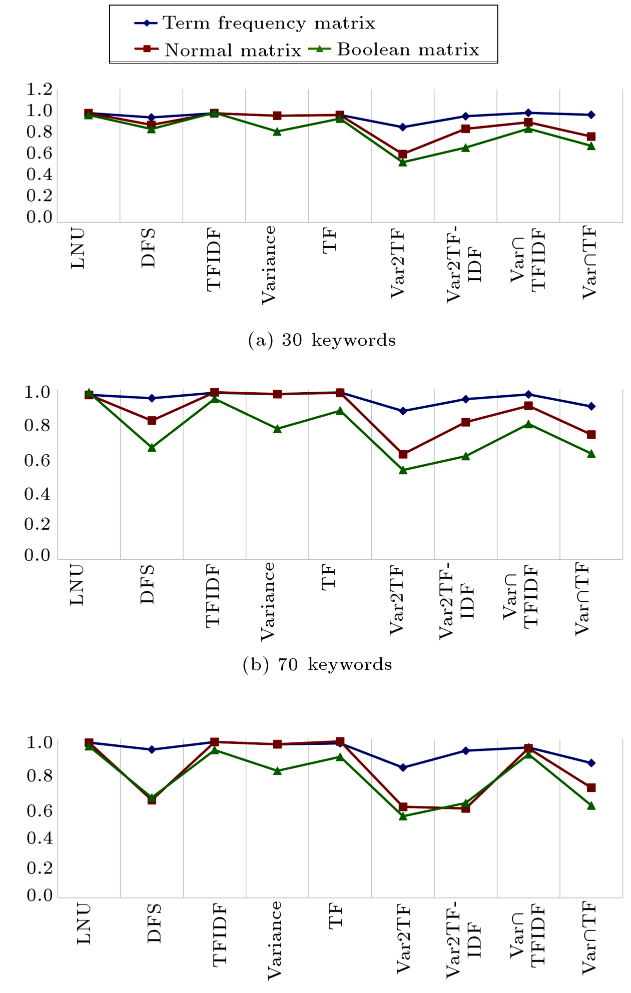
5.4. *Keyword extraction for Persian*

5.4.1. *Evaluation of K-means clustering*

In the current section, the results of all weighting methods evaluated by K-means for Persian documents are presented in Figure 10 (5 clusters). Similar to the English experiments, the matrices are formed for 30, 70, and 150 features after extracting keywords by all the weighting methods. Afterward, the K-means clustering is performed on the matrices to compare the new clusters with predefined classes. This process has the best performance for 30 features by Var2TF with the Boolean matrix and the worst performance by the TF-IDF with all three matrices. For 70 features, the proposed Var2TF method shows the best response with the Boolean matrix, while the TF-IDF method shows the worst response with all three matrixes. In 150 features, the Var2TF shows the best response with the Boolean matrix, while the TF-IDF has the worst response with normal and frequency matrices. It is implied that the proposed Var2TF is the best method for keyword extraction.

5.4.2. *Evaluation of EM clustering*

In this section, the results of the weighting methods for Persian documents are evaluated by the EM method (5 clusters). The results are given in Figure 11. As in Section 5.4.1, 30, 70 and 150 features are extracted, to which EM clustering is applied. For 30 and 70 features, the Var2TF has the best response with normal and frequency matrices. For 150 features, the TF



(a) 30 keywords

(b) 70 keywords

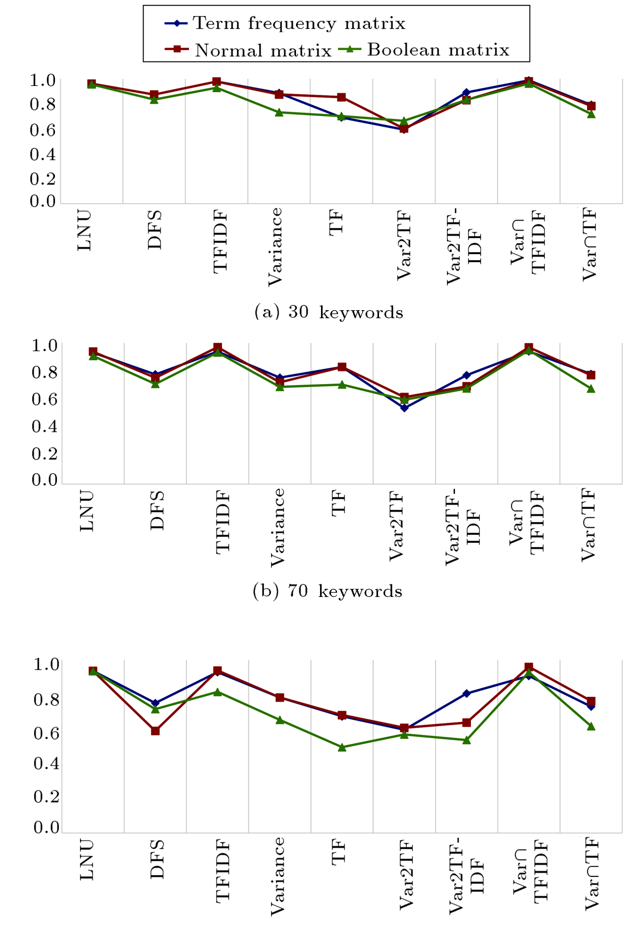
(c) 150 keywords

Figure 10. K-means results for different numbers of extracted keywords in Persian text documents using different methods (5 clusters).

method has the best response with the Boolean matrix, while the TF-IDF method shows the worst response for all features with normal and frequency matrices. The results of this section show that the best keyword extraction methods are TF and Var2TF.

5.4.3. Evaluation of Ward clustering

Now, the results of the weighting methods are evaluated by the hierarchical Ward for Persian documents, and the results are shown in Figure 12 (5 clusters). Similar to Sections 5.4.1 and 5.4.2, 30, 70, and 150 features are extracted in this section before applying the Ward clustering to them. In 30 and 70 features, the Var2TF method shows the best responses with the Boolean matrix. In addition, for the 150 features, the best response is given by the Var2TF with normal and Boolean matrices. In this experiment, the worst response for all features is obtained by the TF-IDF method with normal and frequency matrices. The results of this section also show the effectiveness of



(a) 30 keywords

(b) 70 keywords

(c) 150 keywords

Figure 11. Expectation Maximization (EM) results for different numbers of extracted keywords in Persian text documents using different methods (5 clusters).

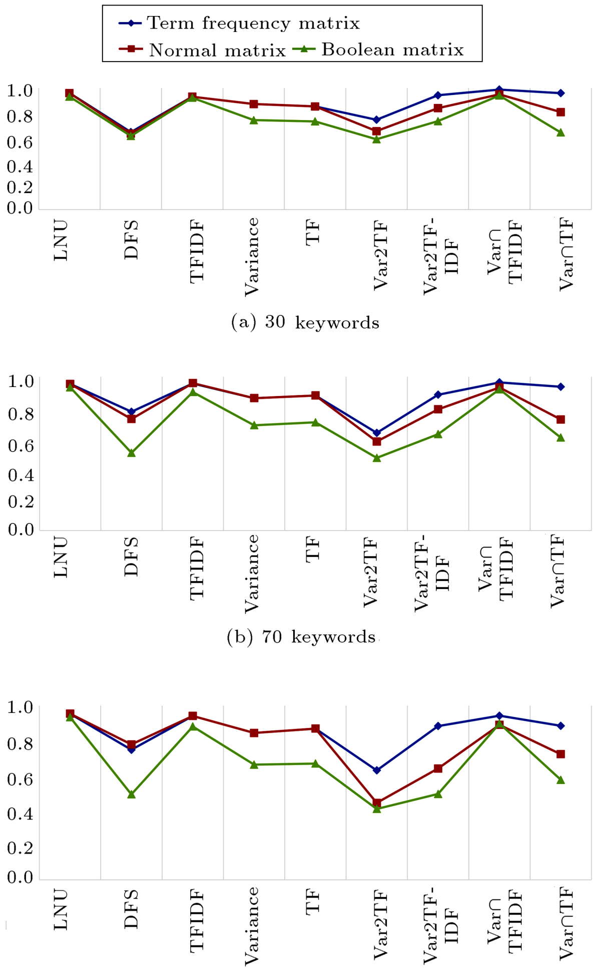
the Var2TF for keyword extraction in Persian in comparison with the other methods.

5.5. Comparative results

This section presents a comparative study of the keyword extraction methods, clustering methods, and the numbers of features. In Figure 13, the average entropy values of the proposed and the reference keyword extraction methods (i.e., weighting) are shown for English and Persian text documents. This figure presents the evaluation results of three clustering methods. The entropy values shown in this figure are the average of the entropy values of three sets of features (i.e., 30, 70, and 150) and three feature types (i.e., Boolean, normal, and TF matrices).

As shown in Figure 13, for keyword extraction in English, the proposed Var2TF-IDF method in Ward clustering has the best entropy value (i.e., the lowest entropy value). Besides, in Persian keyword extraction, the proposed weighting, Var2TF, has the best entropy value in Ward clustering.

Among other clustering methods, the proposed



(a) 30 keywords

(b) 70 keywords

(c) 150 keywords

Figure 12. Ward results for different numbers of extracted keywords in Persian text documents using different methods (5 clusters).

keyword extraction methods achieve good entropy values, too. For English, the Var2TF-IDF and Var∩TF are the best methods for the K-means and EM clustering methods, respectively. In Persian, the Var2TF is the best method for other clustering methods, too. Moreover, it is notable that the average of the entropy values for Persian is higher than that for English. This is probably due to a higher overlap of the documents belonging to different classes for Persian than English.

The average of entropy values obtained from the three clustering methods is shown in Figure 14 (8 clusters). In this figure, the average is calculated with the entropy values of nine weighting methods, three numbers of keywords, and also all three feature extraction techniques. The results show that the Ward method produced lower entropy than others and K-means achieved the highest value. This demonstrates the higher clustering power of Ward than the other methods.

The results shown in Figure 15 are obtained by averaging the outcome of all nine weighting methods, three clustering techniques, and three numbers of keywords to show the entropy values for three types of features used in clustering (i.e., matrices). It is clear that the Boolean matrix in both English and Persian text documents has the best result (i.e., lower entropy values). It is also evident that the average entropy values for Persian are higher than English.

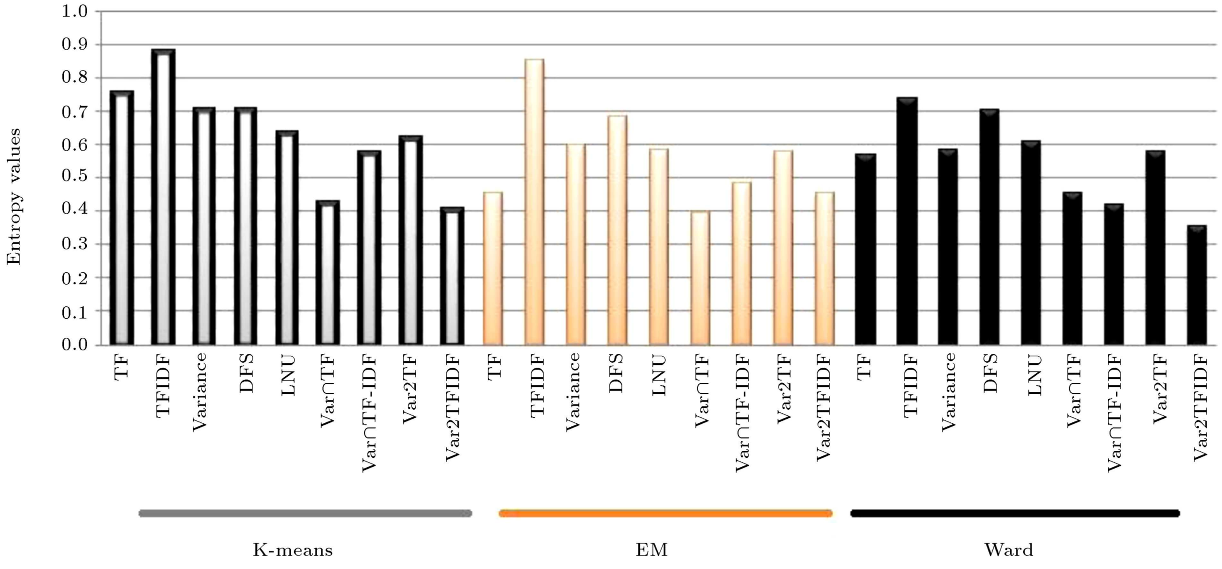
The averaging results of all nine weighting methods, three clustering techniques, and three features extraction methods to show the entropy values for numbers of keywords are shown in Figure 16. As expected, an increase in the number of keywords resulted in better clustering (i.e., lower entropy values). Moreover, it can be seen that the average entropy values for Persian are higher than English.

6. Summary and conclusions

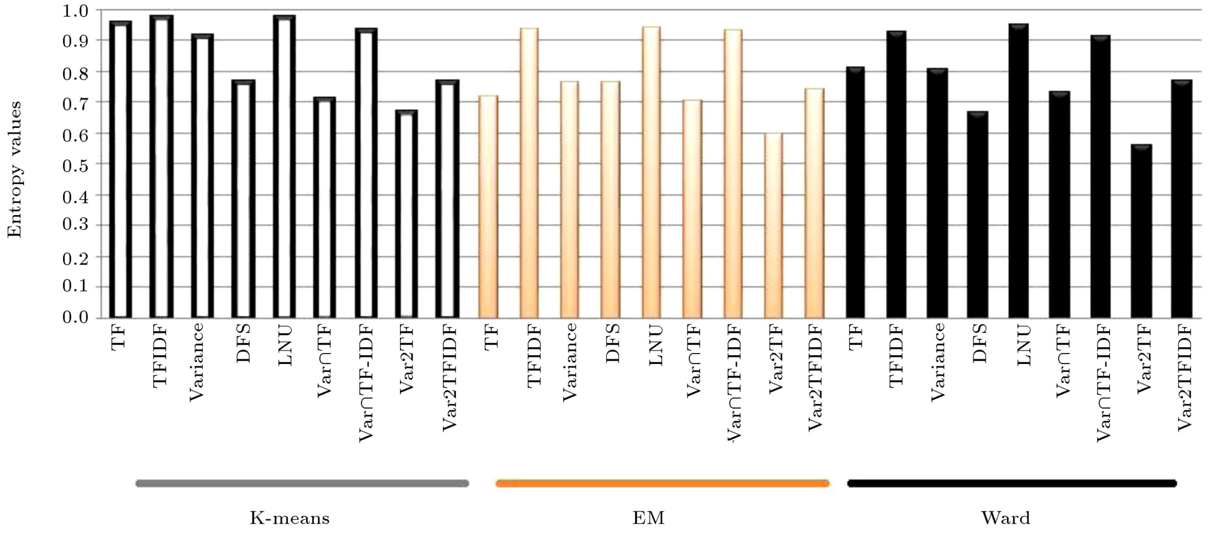
Keyword extraction is often done to realize the overall concept of documents and give readers an overall view. In this paper, five statistical methods of keyword extraction from a text document were presented, and four others were proposed. In a keyword extraction method, first, all terms were weighted by using the reference and proposed methods, and the terms with the highest weights were selected as the keywords. The proposed weighting methods included variance to Term Frequency Inverse Document Frequency (TF-IDF) ratio, variance to Term Frequency (TF) ratio, the intersection of TF and variance, and the intersection of variance and TF-IDF. To evaluate the proposed methods, this study used documents with predefined classes. All documents were clustered using three clustering methods: K-means, Expectation Maximization (EM), and hierarchical Ward. The clusters were then compared with predefined classes using entropy value as the evaluation metric.

This study conducted the necessary evaluations for both English and Persian documents for three sets of keywords. For English documents, the best keyword extraction method was the proposed variance to TF-IDF ratio method and, for Persian documents, the best method was the variance to TF ratio method. Moreover, the methods were evaluated using three different feature extraction techniques during the clustering, with results showing the effectiveness of the Boolean method in comparison with the other methods.

As an extension of this research, the use of a semantic approach to extracting and selecting features will be pursued. In other words, after extracting and counting the features, the features that are semantically similar and, yet, lack identical written forms are considered as a single word (feature). The similarity of the words can be discerned by using a semantic dictionary like WordNet in English and FarsNet in

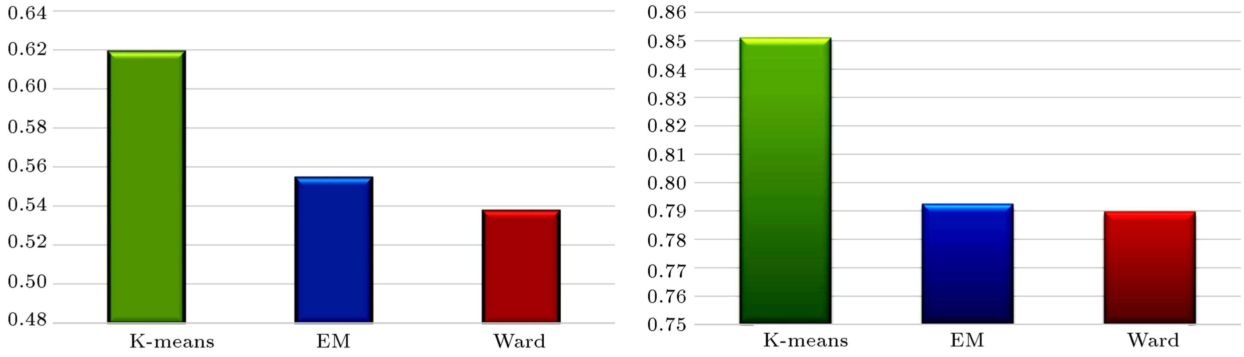


(a) English text documents



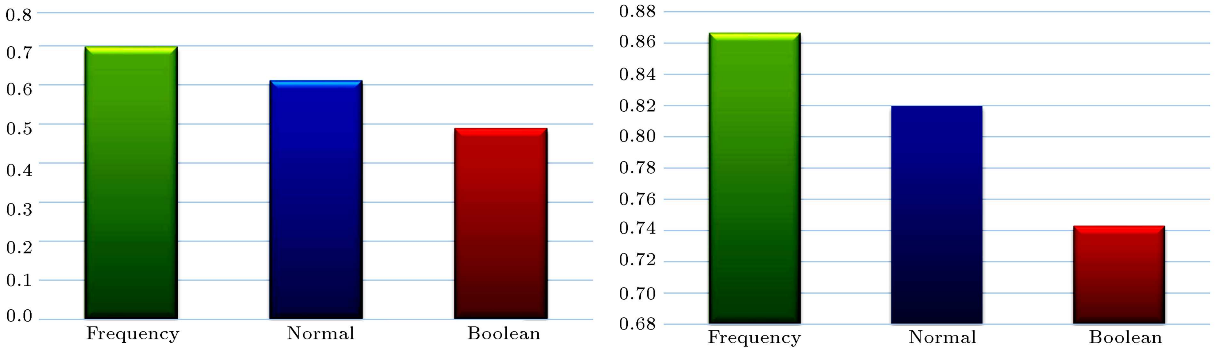
(b) Persian text documents

Figure 13. Average of entropy values for all weighting methods in (a) English and (b) Persian documents for different clustering methods (5 clusters).



(a) English text documents (b) Persian text documents

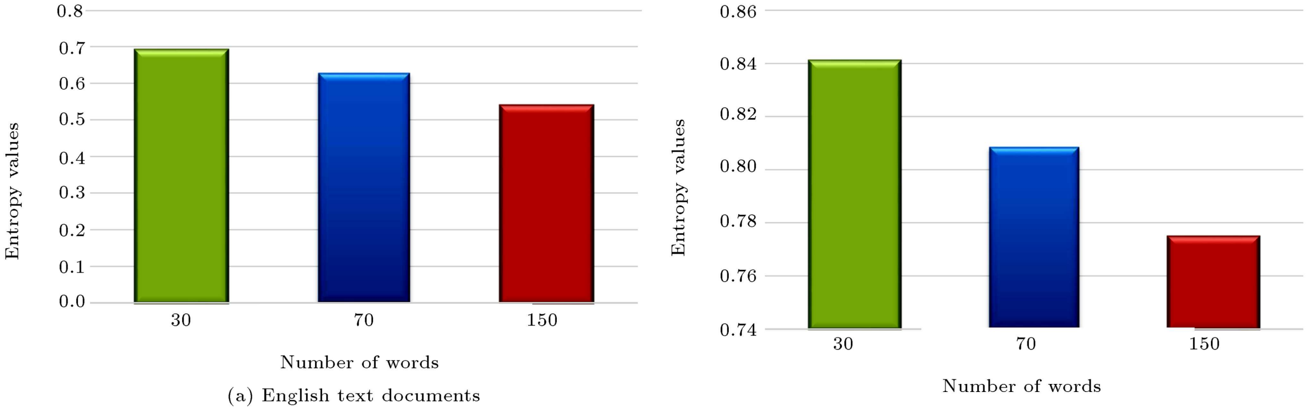
Figure 14. Average of entropy values for clustering methods in (a) English and (b) Persian documents.



(a) English text documents

(b) Persian text documents

Figure 15. Average of entropy values for different features of clustering in (a) English and (b) Persian documents.



(a) English text documents

(b) Persian text documents

Figure 16. Average of entropy values for different numbers of keywords in (a) English and (b) Persian documents.

Persian [18]. By considering every sense of each word, similar words are counted as a single word and the rank of word counting would increase TF, TF-IDF, Variance, Discriminative Feature Selection (DFS), the ratio of variance to TF, and the ratio of variance to TF-IDF methods.

References

1. Liu, J. and Wang, J. "Keyword extraction using language network", In *Proceedings of the International Conference on Natural Language Processing and Knowledge Engineering*, pp. 129–134 (2007).
2. Rossi, R.G., Maracini, R.M., and Rezende, S.O. "Analysis of domain independent statistical keyword extraction methods for incremental clustering", *Learning and Nonlinear Models*, **12**(1), pp. 17–37 (2014).
3. Siddiqi, S. and Sharan, A. "Keyword and keyphrase extraction techniques: a literature review", *International Journal of Computer Applications*, **109**(2), pp. 18–23 (2015).
4. Beliga, S., Mestrovic, A., and Martincic-Ipsic, S. "An overview of graph-based keyword extraction methods and approaches", *Journal of Information and Organizational Sciences*, **39**(1), pp. 1–20 (2015).
5. Taeho, C.J. "Text categorization with the concept of fuzzy set of informative keywords", *Fuzzy Systems Conference Proceedings*, FUZZ-IEEE'99, 1999 IEEE International, **2**, IEEE (1999).
6. Mohammadi, M. and Analouyi, M. "Keyword extraction in Persian documents", *13th Conference of Iran Computer Association*, Kish, Iran (2007).
7. Biswas, S.K., Bordoloi, M., and Shreya, J. "A graph based keyword extraction model using collective node weight", *Expert Systems with Applications*, **97**, pp. 51–59 (2018).
8. Noh, H., Joe, Y., and Lee, S. "Keyword selection and processing strategy for applying text mining to patent analysis", *Expert Systems with Applications*, Elsevier (2015).
9. Haulth, A. "Improved Automatic Keyword Extraction Given More Linguistic Knowledge", In *Proceedings of the 2003 Conference on Emprical Methods in Natural Language Processing*, Sapporo, Japan: pp. 216–223 (2003).
10. Ercan, G. and Cicekli, I. "Using lexical chains for keyword extraction", *Information Processing and Management*, pp. 1705–1714 (2007).

11. Zhang, K., Xu, H., Tang, J., et al. "Keyword extraction using support vector Machine", In *Proceeding of the 7th International Conference on Web-AgemInformation Management*, Hong Kong, China, pp. 85–96 (2006).

12. Onan, A., Korukoğlu, and Bulut, S. "Ensemble of keyword extraction methods and classifiers in text classification", *Expert Systems with Applications*, **57**(C), pp. 232–247 (2016).

13. Wartena, C., Brussee, R., and Slakhorst, W. "Keyword extraction using word co-occurrence", *IEEE 2010 Workshops on Database and Expert Systems Applications*, pp. 54–58 (2010).

14. Arabi, S., Vahidi, M., and Minaei, B. "Keyword extraction for Persian text categorization", *First Iranian Data Mining Conference*, Amirkabir University of Technology (2007).

15. Frank, E. and Paynter, I.H. "Domain-specific keyphrase extraction", In *Proceedings of the 16th International Joint Conference on Artificial Intelligence*, Stockholm, Sweden, Morgan Kaufman, pp. 668–673 (1999).

16. Najafi, E. and Darooneh, A.H. "The fractal patterns of words in a text: a method for automatic keyword extraction", *Plos One*, **10**(6), e0130617 (2015).

17. Haggag, M. "Keyword extraction using semantic analysis", *International Journal of Computer Applications*, **61**(1), pp. 1128–1132 (2013).

18. Shamsfard, M., Hesabi, A., Fadaei, H., et al. "Semi-automatic development of farsNet; the Persian wordNet", *Proceedings of 5th Global WordNet Conference*, Mumbai, India, **29** (2010).

19. Keith, J.B. "Phraserate: an Html keyphrase extractor", Technical Report, University of California, Riverside, pp. 1–16 (2002).

20. Sharma, N., Bajpai, A., and Litoriya, R. "Comparison the various clustering algorithms of Weka tool", *International Journal of Emerging Technology and Advanced Engineering*, **4**(7), pp. 73–80 (2012).

21. Aleahmad, A., Hakimian, P., Mahdikhani, F., et al. "N-gram and local context analysis for Persian text retrieval", *International Symposium on Signal Processing and Its Applications (ISSPA2007)*, Sharjah, United Arab Emirates (UAE), pp. 12–15 (2007).

22. Zong, W., Wu, F., Chu, L., et al. "A discriminative and semantic feature selection method for text categorization", *Int. J. Production Economics Elsevier*, **165**(1), pp. 215–222 (2015).

23. Kaur, M. and Kaur, N. "Web document clustering approaches using K-means algorithm", *International Journal of Advanced Research in Computer Science and Software Engineering*, **3**(5), pp. 861–864 (2013).

24. Zarandi, M.F., Faraji, M.R., and Karbasian, M. "An exponential cluster validity index for fuzzy clustering with crisp and fuzzy data", *Scientia Iranica, Transactions E, Industrial Engineering*, **17**(2), p. 95 (2010).

25. McLaclan, G. and Krishnan, T., *The EM Algorithm and Extensions*, Wiley 2nd Edition, New Jersey (2008).

26. Zhao, Y. and Karypis, G. "Hierarchical clustering algorithms for document datasets", *Journal of Data Mining and Knowledge Discovery Elsevier*, **10**(2), pp. 141–168 (2005).

27. Montazeri-Gh, M. and Fotouhi, A. "Traffic condition recognition using the k-means clustering method", *Scientia Iranica, Transactions B: Mechanical Engineering*, **18**, pp. 930–937 (2011).

28. Hartigan, J.A. and Wong, M.A. "Algorithm as 136: A *k*-means clustering algorithm", *Journal of the Royal Statistical Society, Series C (Applied Statistics)*, **28**(1), pp. 100–108 (1979).

29. Jain, A.K. "Data clustering: 50 years beyond K-means", *International Conference on Pattern Recognition (ICPR) Elsevier*, pp. 651–666 (2010).

30. Jeff, Wu. C.J. "On the convergence properties of the EM algorithm", *The Annals of Statistics*, **11**(1), pp. 95–103 (1983).

31. Bock, R.D. and Aitkin, M. "Marginal maximum likelihood estimation of item parameters: Application of an EM algorithm", *Psychometrika*, **46**(4), pp. 443–459 (1981).

32. Bottou, L. and Bengio, Y. "Convergence properties of the k-means algorithms", *Advances in Neural Information Processing Systems*, pp. 585–592 (1995).

33. Johnson, S.C. "Hierarchical clustering schemes", *Psychometrika*, **32**(3), pp. 241–254 (1967).

34. Web, A.R., *Statistical Pattern Recognition*, John Wiley & Sons. 2nd Edition (2002).

35. Olson, C.F. "Parallel algorithms for hierarchical clustering", *International Journal of Parallel Computing*, Elsevier, **21**(8), pp. 1313–1325 (1995).

36. Mangiameli, P., Chen, S.K., and West, D. "Comparison of SOM neural network and hierarchical clustering method", *European Journal of Operational Research*, **93**(2), pp. 402–417 (1976).

37. He, Q., *A Review of Clustering Algorithms as Applied in IR*, Graduate School of Library and Information Science University of Illinois at Urbana-Champaign (1999).

38. Batagelj, V., *Generalized Ward and Related Clustering Problem. Classification and Related Methods of Data Analysis*, H.H. Bock (editor), North Holland, Amsterdam, pp. 67–74 (1986).

Biographies

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, Kish International Campus 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.

Pouyan Parsafard began his studies in Computer Science at KIAU Azad University of Karaj for 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, Kish International Campus 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, fuzzy systems, and data mining.