www.ijemr.net ISSN (ONLINE): 2250-0758, ISSN (PRINT): 2394-6962

Volume-7, Issue-2, March-April 2017

International Journal of Engineering and Management Research

Page Number: 70-79

Pre Processing Techniques for Arabic Documents Clustering

Mohammed Alhanjouri
Computer Engineering Department, Islamic University of Gaza, GAZA

ABSTRACT

Clustering of text documents is an important technique for documents retrieval. It aims to organize documents into meaningful groups or clusters. Preprocessing text plays a main role in enhancing clustering process of Arabic documents. This research examines and compares text preprocessing techniques in Arabic document clustering. It also studies effectiveness of text preprocessing techniques: term pruning, term weighting using (TF-IDF), morphological analysis techniques using (root-based stemming, light stemming, and raw text), and normalization. Experimental work examined the effect of clustering algorithms using a most widely used partitional algorithm, K-means, compared with other clustering partitional algorithm, Expectation Maximization (EM) algorithm. Comparison between the effect of both Euclidean Distance and Manhattan similarity measurement function was attempted in order to produce best results in document clustering. Results were investigated by measuring evaluation of clustered documents in many cases of preprocessing techniques.

Experimental results show that evaluation of document clustering can be enhanced by implementing term weighting (TF-IDF) and term pruning with small value for minimum term frequency. In morphological analysis, light stemming, is found more appropriate than root-based stemming and raw text. Normalization, also improved clustering process of Arabic documents, and evaluation is enhanced.

Keywords— Arabic Text Mining, Arabic document clustering, Arabic text preprocessing, Term weighting, Arabic morphological analysis, Vector Space Mode (VSM).

I. INTRODUCTION

An electronic text that available on web is increasing rapidly, such as electronic publications, electronic books, news articles and web pages. Beside this huge increasing for online text information, the researcher on the internet has a big challenging to extract the relevant knowledge that needed. The need for tools to help people for finding, filtering, sorting and managing these resources has grown. Thus, automatic organization of text document collections has become an important research issue. A many of machine learning techniques have been proposed to enhance the organization of text data automatically. There are two main categories for machine learning techniques, supervised (document classification) and unsupervised (document clustering) [1].

II. BACKGROUND

A. Arabic Language

The Arabic language is one of the 5th widely used in the world. There are about 300 million people talking the Arabic as the first language, while 250 million use it as the second language. Because the Arabic language has unique morphological principles [2], there are relatively few studies on the retrieval or mining of Arabic electronic text documents in the literature.

Actually, the Arabic language has 3 forms; Classical Arabic (CA), Modern Standard Arabic (MSA), and Dialectal Arabic (DA). CA, MSA, and DA forms include classical historical liturgical text, news media and formal speech, and predominantly spoken vernaculars and have no written standards, respectively. Arabic alphabet consists of 28 letters, and the Hamza (ء) as an additional letter. There is no upper or lower case for Arabic letters. The letters (ي ,و ,ا) are vowels, and the rest letters are constants. Unlike Latin-based alphabets, the orientation of writing in Arabic is from right to left.

The Arabic script has numerous diacritics, including I'jam (إعجام), consonant pointing, and tashkil (تشكيل), supplementary diacritics. The latter include the ḥarakat (حركات , singular haraka حركة), vowel marks. The literal meaning of tashkil is "forming". As the normal Arabic text does not provide enough information about the correct pronunciation, the main purpose of tashkil (and ḥarakat) is to provide a phonetic guide or a phonetic aid; i.e. show the correct pronunciation (double the word in pronunciation or to act as short vowels). The ḥarakat, which literally means "motions", are the short vowel

Copyright © 2016. Vandana Publications. All Rights Reserved.

marks[18]. Arabic diacritics include Fatha, Kasra, Damma, Sukūn, Shadda, and Tanwin. Arabic words have two genders, masculine (مذكر) and feminine (مؤنث); three numbers, singular (مفرد), dual (مثنى), and plural (جمع); and three grammatical cases, nominative (رفع), accusative (نصب), and genitive (جر). A noun has the nominative case when it is subject (فاعل); accusative when it is the object of a verb (مفعول); and the genitive when it is the object of a preposition (مجرور). Words are classified into three main parts of speech, nouns (أسماء) (including adjectives (صفات) and adverbs (ظروف)), verbs (أفعال), and particles (أدوات) [3].manuscript.

B. Arabic Language Challenges

Arabic is a challenging language for a number of reasons [2]:

1. Orthographic with diacritics is less ambiguous and more phonetic in Arabic, certain combinations of characters can be written in different ways. For example, sometimes in glyphs combining HAMZA with ALEF (أ) the HAMZA is dropped (ا). This makes the glyph ambiguous as to whether the HAMZA is present.
2. Arabic has a very complex morphology recording as compared to English language. For example, to convey the possessive, a word shall have the letter (ي) attached to it as a suffix. There is no disjoint Arabic-equivalent of "my".
3. Arabic words are usually derived from a root (a simple bare verb form) that usually contains three letters. In some derivations, one or more of the root letters may be dropped. In such cases tracing the root of the derived word would be a much more difficult problem.
4. Broken plurals are common. Broken plurals are somewhat like irregular English plurals except that they often do not resemble the singular form as closely as irregular plurals resemble the singular in English. Because broken plurals do not obey normal morphological rules, they are not handled by existing stemmers.
5. In Arabic we have short vowels which give different pronunciation. Grammatically they are required but omitted in written Arabic texts.
6. Arabic synonyms are widespread. Arabic is considered as one of the richest languages in the world. This makes exact keyword match is inadequate for Arabic retrieval and classification.

The main objective of the research is to cluster Arabic documents using partition-based algorithm, to give best performance for evaluation, by selecting best combinations of text preprocessing, best clustering algorithm, and best similarity measurement function.

To achieve the main objective, we accomplished some steps such as:

- Study impact of text preprocessing in clustering evaluation.
- Evaluate clustering process in Arabic document using K-means algorithm, according to recall, precision, F-measure evaluation to build model.
- Study if K-means algorithm is appropriate for Arabic text.
- Use machine learning tool at home for clustering experiments, (WEKA) which is an excellent open-source of data mining tool in abroad, but it is rarely used at home.
- Provide comprehensive guide for using best text preprocessing combination for best clustering evaluation.
- Applying several Arabic morphological analysis tools.

C. Related works

Ghwanmeh [4] implemented clustering technique which is K-Means like with hierarchical initial set (Hierarchical K-Means Like clustering HKM). He proved that clustering document sets do enhancement precision on information retrieval systems, since it was proved by Bellot & El-Beze on French language. He made comparison between the traditional information retrieval system and the clustered one. Also the effect of increasing number of clusters on precision is studied. The indexing technique is Term Frequency * Inverse Document Frequency (TF-IDF). It has been found that the effect of Hierarchical K-Means Like clustering (HKM) with 3 clusters over 242 Arabic abstract documents from the Saudi Arabian National Computer Conference has significant results compared with traditional information retrieval system without clustering. Additionally it has been found that it is not necessary to increase the number of clusters to improve precision more. He applied 59 queries on 242 Arabic abstract documents, which are clustered into several sets of clusters (2, 3 and 5), then he compared the results with the traditional IR system. To determine the appropriate number of clusters; a series of tests have been made at several number of clusters (2, 3, and 5), and was found that the best results is at 3 clusters which means that this corpora talks mainly about three topics. In his results the best precision was obtained is 0.49 which enhances results without using clustering by 13%.

A. Abdelfatah, and A. Yahya, [5] used "Frequent Itemset-based Hierarchical Clustering (FICH)" clustering algorithm to cluster Arabic. They conducted their experiments on 600 Arabic documents using N-grams based on word level, Trigrams and Quadgrams and they got promising results. They conducted their experiments using N-grams based on word level and character level Trigrams and Quadgrams. For the accuracy of clusters, word level outperforms both Quadgrams and Ttrigrams for both 4 and 6 natural classes, and Quadgrams gave better accuracy than Trigrams for both 4 and 6 natural classes. For the word level they got accuracy of 0.75 for four natural classes for 4 clusters, and they got accuracy of 0.70 for Quadgrams for four natural classes for 4 clusters, and 0.63 for Trigrams for four natural classes for 8 clusters.

Rafi et al. [6] compared and contrast two approaches to document clustering based on suffix tree data model. The first is an Efficient Phrase based document clustering, which extracts phrases from documents to form compact document representation and uses a similarity measure

Copyright © 2016. Vandana Publications. All Rights Reserved.

based on common suffix tree to cluster the documents. The second approach is a frequent word/word meaning sequence based document clustering, it similarly extracts the common word sequence from the document and uses the common sequence/ common word meaning sequence to perform the compact representation, and finally, it uses document clustering approach to cluster the compact documents. These algorithms are using agglomerative hierarchical document clustering to perform the actual clustering step, the difference in these approaches are mainly based on extraction of phrases, model representation as a compact document, and the similarity measures used for clustering. They investigated the computational aspect of the two algorithms, and the quality of results they produced. The result of experiment shows that the F-score obtained from the test data sets clearly exhibits the superiority of algorithm "Efficient Phrase based clustering algorithm" over algorithm "Text document clustering based on frequent word meaning sequences", on variety of situations. They clearly concluded from the results obtained that Efficient Phrase based clustering algorithm is superior.

Froud et al. [7] evaluated the impact of the stemming on the Arabic Text Document Clustering. Their experiments show that the use of the stemming will not yield good results, but makes the representation of the document smaller and the clustering faster. The representation of the documents and the use of the stemming affect the final results. The stemming makes the representation of the document smaller and the clustering faster.

Alkoffash [8] implemented the K-means and K-mediods algorithms in order to make a practical comparison between them. The system was tested using a manual set of clusters that consists from 242 predefined clustering documents. The results showed a good indication about using them especially for K-mediods. The average precision and recall for K-means compared with K-mediods are 0.56, 0.52, 0.69 and 0.60 respectively. He also extracted feature set of keywords in order to improve the performance, the result illustrated that two algorithms can be applied to Arabic text, a sufficient number of examples for each category, the selection of the feature space, the training data set used and the value of K can enormously affect the accuracy of clustering. Recall and precision measurers are used for evaluation. Results show K-mediods is better than K-means due to the chance that is given for several files in K--mediods to become a center for a given cluster. Evaluation for K-mediods: 0.60, 0.69 for Average Recall, and Average Precision respectively, despite evaluation for K-means: 0.525, 0.565 for Average Recall, and Average Precision respectively. He concluded that manipulating large corpus may give results that are more nearby to the manual one. Clustering environment is more unbiased than manual due to its dependability on the system rather than user opinion. Most of the errors or weakness that appear in Arabic retrieval systems, due to the strength of language itself that contains several features not existed in any other one. The problem of K-means and K-mediods are represented by selecting initial points, problems of differing sizes, densities, and shapes and outliers data.

Froud et al. [9] proposed to compare the clustering results based on summarization with the full-text baseline on the Arabic Documents Clustering for five similarity/distance measures for three times: without stemming, and with stemming using Khoja‟s stemmer, and the Larkey‟s stemmer. They found that the Euclidean Distance, the Cosine Similarity and the Jaccard measures have comparable effectiveness for the partitional Arabic Documents Clustering task. They used the K-means algorithm as document clustering method. Results for Khoja‟s stemmer, the overall purity values for the Euclidean Distance, the Cosine Similarity and the averaged Kullback-Leibler divergence (KL divergence) are quite similar and performs bad relatively to the other measures. Meanwhile, the Jaccard measure is the better in generating more coherent clusters with a considerable purity score. In this context, using the Larkey‟s stemmer, the purity value of the averaged KL Divergence measure is the best one with only 1% difference relatively to the other four measures. In the other hand, results without stemming shows the higher purity scores (0.77) than those shown for the Euclidean Distance, the Cosine Similarity and the Jaccard measures. In the other hand the Pearson Correlation and averaged KL Divergence are quite similar but still better than purity values for these measures KHOJA‟S stemmer, and LARKEY‟S stemmer. Other best results show the better and similar entropy values for the Euclidean Distance, the Cosine Similarity and the Jaccard measures. In overall results shows that the use of stemming affects negatively the clustering, this is mainly due to the ambiguity created when we applied the stemming (for example, two roots are obtained that made of the same letters but semantically different).

Ahmed and Tiun[10] evaluated the efficiency and accuracy of Arabic Islamic document clustering based on K-means algorithm with three similarity/distance measures; Cosine, Jaccard similarity and Euclidean distance. Additionally, research investigated the effect of using stemming and without stemming words on the accuracy of Arabic Islamic text clustering. They used Islamic dataset (in-house). Based on the results, the K-means algorithm has the best results with Cosine similarity compared to Jaccard similarity and Euclidean distance. The results with Euclidean distance are better than the results with Jaccard similarity. In addition, they concluded that the results with stemming method are better than without stemming. They also depicted that the results depend on number of categories and size of dataset.

III. METHODOLOGY

Copyright © 2016. Vandana Publications. All Rights Reserved.

ISSN (ONLINE): 2250-0758, ISSN (PRINT): 2394-6962

A. Collect Arabic Text Documents

Collection of Data includes the processes like crawling, indexing, filtering, etc., These processes are used to collect documents to be clustered, indexed to store and retrieve in a better way, and filtered to remove extra data; for example, stopwords [11]. Large Arabic corpus of text documents as well as two freely public datasets was used for experiments. The dataset was collected from CNN Arabic website because it is free, public, contains suitable number of documents for clustering process and suitable to for the hardware used in the experiments. CNN Arabic dataset used in the experiments is related to various categories, such as Business, Entertainments, Middle East News, Science and Technology, Sports, and World News.

Table 1
Number of documents in category of CNN testing dataset

id	Text Categories	Number of documents	% from corpus
1	Business	836	16.49
2	Entertainments	474	9.35
3	Middle East News	1462	28.84
4	Science & Technology	526	10.37
5	Sports	762	15.03
6	World News	1010	19.92
Total		5070	100%

The second dataset used in the experiments was BBC Arabic corpus, which has been collected from BBC Arabic website bbcarabic.com. As shown in Table 2, the corpus includes 4,763 text documents; each text document belongs to 1 of 7 categories: Middle East News, World News, Business & Economy, Sports, International Press, Science & Technology and Art & Culture. The corpus contains 1,860,786 (1.8M) words and 106,733 district keywords after stopwords removal. The corpus was converted to utf-8 encoding and html tags were stripped.

Table 2
Number of documents in category of BBC testing dataset

id	Text Categories	Number of documents	% from corpus
1	Middle East News	2356	49.46
2	World News	1489	31.26
3	Business & Economy	296	6.21
4	Sports	219	4.59
5	International press	49	1.028
6	Science & Technology	232	4.87
7	Art & Culture	122	2.56
Total		4763	100%

B. Arabic Text Preprocessing Techniques

Text preprocessing consists of text input, word segment and stop-word filters, which require as much as 80 percent of the total effort. After the segment and filter, the dimensionality of the text feature vector can be significantly reduced, and hence, the processing effort needed in the discovery phase can be decreased greatly [12].

There are six techniques for Arabic text preprocessing:

1. String Tokenization
2. Dropping common terms: stop words
3. Normalization
4. Morphological Analysis Techniques (Stemming and Light Stemming)
5. Term Pruning
6. Vector Space Model (VSM) and Term Weighting Schemes

Tokenization is the task of chopping it up into pieces, called tokens, perhaps at the same time throwing away certain characters, such as punctuation. A token is an instance of a sequence of characters in some particular document that are grouped together as a useful semantic unit for processing. A type is the class of all tokens containing the same character sequence. A term is a (perhaps normalized) type that is included in the IR system‟s dictionary [13].

Stop words are common words which would appear to be of little value in helping select documents matching a user need are excluded from the vocabulary entirely. The general strategy for determining a stop list is to sort the terms by collection frequency (the total number of times each term appears in the document collection), and then to take the most frequent terms, often hand-filtered for their semantic content relative to the domain of the documents being indexed, as a stop list, the members of which are then discarded during indexing

As data variables are of variable size and scales, it is therefore essential that we scale the data variables so that they are comparable. Data scaling can be performed by normalizing or standardizing the data variables, which is typically performed on the independent variables. Normalization scales each data variable into a range of 0 and 1 as shown in the following equation:

$$x_{ij}^{normalization} = \frac{x_{ij} - x_j^{min}}{x_j^{max} - x_j^{min}}$$

Where $x_{ij}^{normalization}$ represents the normalized value, x_{ij} represents the value of interest, x_i^{min} represents the minimum value and x_j^{max} represents the maximum value. After being scaled, the minimum value would become 0 and the maximum value would become 1, while all other values would be in between 0 and 1 [14].

Copyright © 2016. Vandana Publications. All Rights Reserved.

C. Morphological Analysis Techniques (Stemming and Light Stemming)

Applying stemming algorithms as a feature selection method reduces the number of features since lexical forms (of words) are derived from basic building blocks; and hence, many features that are generated from the same stem are represented as one feature (their stem) [15]. These stemmers mostly deal with the removal of suffixes as this is sufficient for most information retrieval purposes.

Arabic Language needs special stemming algorithms for many reasons described by El-Sadany and Hashish in the following points [16]:

i. Arabic is one of Semitic languages which differ in structure of affixes from Indo-European type of languages such as English and French;

ii. Arabic is mainly roots and templates dependent in the formation of words;

iii. Arabic roots consonants might be changed or deleted during the morphological process.

We will view two approaches for stemming: the root-based stemmer, and the light stemmer.

- Root-based Stemming

Stemming using root extractor uses morphological analysis for Arabic words. Several algorithms have been developed for this approach such as: RDI MORPHO3 Algorithm, Sebawai root extractor (SR) Algorithm, and Khoja Stemming Algorithm which will be used in our experiments.

Khoja and Garside developed stemmer algorithms [17]. The algorithm, developed by using both Java and C++ languages, removes the longest suffix and the longest prefix. It then matches the remaining word with verbal and noun patterns, to extract the root. The stemmer makes use of several linguistic data files such as a list of all diacritic characters, punctuation characters, definite articles, and 168 stop words. The algorithm achieves accuracy rates of up to 96%. The algorithm correctly stems most Arabic words that are derived from roots.

- Light Stemming

The main idea for using light stemming is that many word variants do not have similar meanings or semantics. However; these word variants are generated from the same root. Thus, root extraction algorithms affect the meanings of words. Light stemming by comparison aims to enhance the categorization performance while retaining the words' meanings. It removes some defined prefixes and suffixes from the word instead of extracting the original root [18]. Light-stemming keeps the word's meanings unaffected.

D. Term Pruning

Term Pruning, in Machine Learning, refers to an action of removing nonrelevant features from the feature space. In text mining, pruning is a useful preprocessing concept because most words in the text corpus are low-frequency words. According to the Zipf's law, given some corpus of natural language texts, if words are ranked according to their frequencies, the distribution of word frequencies is an inverse power law with the exponent of roughly one [19]. This implies that, in any training corpus, the majorities of the words in the corpus appear only a few times. A word that appears only a few times is usually statistically insignificant - low document frequency, low information gain, etc. Moreover, the probability of seeing word, that occurs only once or twice in the training data, in the future document is very low [20]. In the other hand term pruning can be defined as the process of eliminating the words that its count is less or greater than a specific threshold.

E. Vector Space Model (VSM) and Term Weighting Schemes

The representation of a set of documents as vectors in a common vector space is known as the vector space model and is fundamental to a host of information retrieval operations ranging from scoring documents on a query, document classification and document clustering. A pivotal step in this development is the view of queries as vectors in the same vector space as the document collection [21]. In the Vector Space Model, the contents of a document are represented by a multidimensional space vector. The proper classes of the given vector are determined by comparing the distances between vectors. The procedure of the Vector Space Model can be divided into three stages:

1. The first step is document indexing, when most relevant terms are extracted.
2. The second stage is based on the introduction of weights associated to index terms in order to improve the retrieval relevant to the user.
3. The last stage classifies the document with a certain measure of similarity.

The most common vector space model assumes that the objects are vectors in the high-dimensional feature space. A common example is the bag-of-words model of text documents. In a vector space model, the similarity function is usually based on the distance between the vectors in some metric.

In VSM, document can be represented as vector space in high dimensions. Each document can be represent as vector space V(d), V(d)=(($t1$,$w1$),($t2$,$w2$),…,(tn,wn)). Where, ti is the feature i in document d, wi is the weight of ti in document d. The value of wi can be 0 or 1, in the other hand tf * idf is a widely used method in term weight (wi) calculation in document representation. For tf, reflects local weight in each document, idf reflects global weight in all documents [22].

Term weighting is one of preprocessing methods; used for enhanced text document presentation as feature vector. Term weighting helps us to locate important terms in a document collection for ranking purposes [23].

- Term Frequency

Copyright © 2016. Vandana Publications. All Rights Reserved.

This approach is to assign each term in a document a weight for that term, which depends on the number of occurrences of the term in the document. To get this, compute a score between a query term t and a document d, based on the weight of t in d. The simplest approach is to assign the weight to be equal to the number of occurrences of term t in document d. This weighting scheme is referred to as term frequency Term Frequency and is denoted TFt,d, with the subscripts denoting the term and the document in order [21].

$$TF(d,t_i) = \frac{n(d,t_i)}{\sum_i n(d,t_i)}$$

Where n(d,ti) is the number of occurrences of ti in a document and Σi n(d, ti) is the total number of tokens in document.

- Inverse Document Frequency

This approach is to reduce the TF weight of a term by a factor that grows with its collection frequency. Instead, it is more commonplace to use for this purpose the document frequency DFT, defined to be the number of documents in the collection that contain a term t. This is because in trying to discriminate between documents for the purpose of scoring it is better to use a document-level statistic (such as the number of documents containing a term) than to use a collection-wide statistic for the term [21].

$$IDF(t_i) = \log(\frac{D}{D_i})$$

Where Di is the number of documents containing ti and D is the total number of documents in the collection.

- Term Frequency-Inverse Document Frequency (TF-IDF)

TF-IDF is a popular method of preprocessing documents in the information retrieval community.

TF-IDF*t,d* assigns to term t a weight in document d that is:

1. highest when t occurs many times within a small number of documents (thus lending high discriminating power to those documents).
2. Lower when the term occurs fewer times in a document, or occurs in many documents (thus offering a less pronounced relevance signal).
3. lowest when the term occurs in virtually all documents [21].

$$W_{ij} = tfidf(t_i,d_j) = \frac{f_{ij}}{\sqrt{\sum_{k=1}^{M} f_{kj}^2}} \times \log(\frac{N}{n_i})$$

Where N is the number of documents in the data set, M is the number of terms used in the feature space, fij is the frequency of a term i in document j, and ni denotes the number of documents that term i occurs in at least once.

F. Evaluation

There are many evaluation standards in information retrieval used in document clustering such as Entropy, Cluster Purity, and F-measure which will be used in this work.

F-measure [24] is widely used in text clustering. It provides a good balance between precision and recall, which is excellent in the context of information retrieval [25].

- Precision shows how many documents are in right cluster with respect to the cluster size.
- Recall shows how many documents are in the right cluster with respect to total documents.

Precision (P) is the fraction of retrieved documents that are relevant [21].

$$\textbf{Precision} = \frac{\#(\text{relevant items retrieved})}{\#(\text{retrieved items})} = P(\text{relevant}|\text{retrieved})$$

Recall (R) is the fraction of relevant documents that are retrieved.

$$\textbf{Precision} = \frac{\#(\text{relevant items retrieved})}{\#(\text{relevant items})} = P(\text{retrieved}|\text{relevant})$$

These notions can be made clear by examining the following contingency table:

	Relevant	Non relevant
Retrieved	True positive (tp)	False positive (fp)
Not retrieved	False negative (fn)	True negative (tn)

Precision (P) = tp/(tp+fp)

Recall (R) = tp/(tp+fn)

On the other hand we can compute precision and recall for class i and cluster j as:

Recall (i,j) = n_{ij} / n_j

Precision (i,j) = n_{ij} / n_i

Where n_{ij} is the number of documents with class label *i* in cluster *j* , *ni* is the number of documents with class label *i,* and *nj* is the number of documents in cluster *j* , and n is the total number of documents.

The F-measure for class i and cluster j is given as:

$$F(i,j) = \frac{2 * \text{Recall}\ (i,j)\ *\ \text{Precision}\ (i,j)}{\text{Recall}\ (i,j) + \ \text{Precision}\ (i,j)}$$

Then total F-measure of clustering process is calculated as:

$$F = \sum \frac{n_i}{n * \max F(i,j)}$$

IV. EXPERIMENTAL RESULTS AND ANALYSIS

Experimental results were investigated by measuring evaluation of clustered documents in many

Copyright © 2016. Vandana Publications. All Rights Reserved.

ISSN (ONLINE): 2250-0758, ISSN (PRINT): 2394-6962

cases of preprocessing techniques. The two most frequent and basic measures for information retrieval effectiveness (measuring precision and recall) were used for accuracy reasons. The other measurement is F-Measure which is a single measure that trades off precision versus recall.
Many symbols were used in experiments setup for preprocessing combinations, as depicted below in Table 3.

TABLE 3
Symbols used in experiments and their description

Symbol	description
Boolean	Indicating presence (1) or absence (0) of a word
wc	Output word counts
wc-tf	Apply TF trans. On word count
wc-tf-idf	Apply TFIDF trans. On word count
wc-norm	Apply document normalization On word count
wc-minFreq3	Apply term pruning on word count that less than3
wc-norm-minFreq3	Apply norm. and term pruning on word count that less than3
wc-tfidf-norm-minFreq3	Apply TFIDF and norm. on word count that less than3
wc-norm-minFreq5	Apply norm. and term pruning on word count that less than5
wc-tfidf-norm-minFreq5	Apply TFIDF and norm. on word count that less than5

Term pruning is applied for preprocessing in String to Word Vector options by setting the minimum term frequency in the document. In default state the minTermFreq =1, that means no term pruning are applied and all words are contained in dataset. We increased term frequency in many counts (3, 5, 7 and 9) to investigate the impact of term pruning in clustering process. The first dataset is CNN dataset; it is used in experiments with three preprocessing techniques which are: System A is term pruning combining with term weighting and light stemming. System B is term pruning combining with term weighting and normalization. System C is term pruning combining with term weighting, normalization, and root-based stemming (Khoja)

Table 4
results of using three preprocessing techniques for several min. Term Frequency by using CNN dataset

Min. Term Frequency	Preprocessing techniques	A	B	C
3	Precision	0.699	0.614	0.545
	Recall	0.491	0.492	0.331
	F-Measure	0.577	0.546	0.412
5	Precision	0.495	0.625	0.635
	Recall	0.443	0.436	0.509
	F-Measure	0.468	0.514	0.565
7	Precision	0.531	0.591	0.527
	Recall	0.531	0.536	0.527
	F-Measure	0.531	0.563	0.527
9	Precision	0.554	0.620	0.531
	Recall	0.554	0.498	0.531
	F-Measure	0.554	0.552	0.531

From tables 4; results of CNN database show that F-measure has the largest value for minimum term frequency at minTermFreq 3: 0.577, the largest measure is for minimum term frequency at minTermFreq 7: 0.563, and the last value is for minimum term frequency at minTermFreq 5: 0.565. From these results as shown the best value from these results is for minimum term frequency at 3. This gives indication that to use a small value for minimum term frequency to enhance results of text preprocessing as shown in Figure 1.

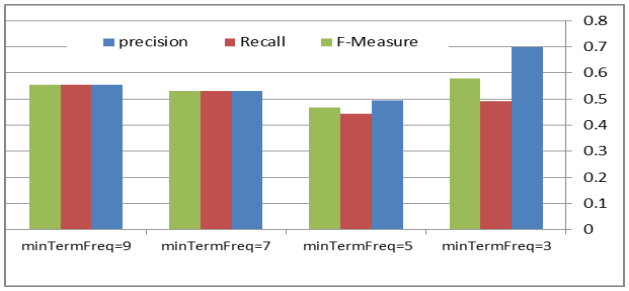

Figure1: results of System A applied on CNN database

Also from figures 1, 2, 3 we can tote that system (A) that applied a term pruning combining with term weighting and light stemming achieved better results than the other two used systems for CNN database

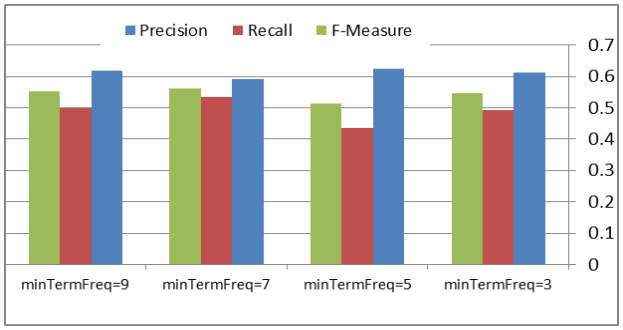

Figure2: results of System B applied on CNN database

Copyright © 2016. Vandana Publications. All Rights Reserved.

ISSN (ONLINE): 2250-0758, ISSN (PRINT): 2394-6962

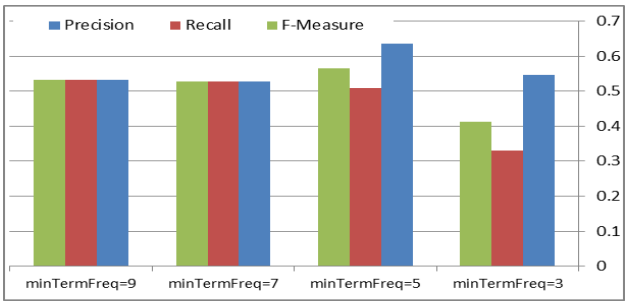

Figure 3: Results of System C applied on CNN database

For confirmation of term pruning impact, and the appropriate value for minimum term frequency, another dataset is used (BBC dataset) to show evaluation of term pruning as shown in table 5.

TABLE 5
results of using three preprocessing techniques for several min. Term Frequency by using BBC dataset

Min. Term Frequency	Preprocessing techniques	A	B	C
3	Precision	0.786	0.366	0.019
	Recall	0.692	0.366	0.019
	F-Measure	0.736	0.366	0.019
5	Precision	0.739	0.370	0.052
	Recall	0.684	0.399	0.052
	F-Measure	0.710	0.384	0.052
7	Precision	0.321	0.309	0.321
	Recall	0.345	0.383	0.330
	F-Measure	0.332	0.342	0.325
9	Precision	0.321	0.056	0.321
	Recall	0.345	0.074	0.420
	F-Measure	0.332	0.064	0.364

From table 5; results depicts that adjustment value of minimum term frequency at minTermFreq 3 gives the best evaluation for precision, recall, and F-measure, in comparison with other results. This also gives indication to use a small value for minimum term frequency to enhance results of text preprocessing. Figure 4 is shows that minimum term frequency at minTermFreq 3 is the best value of evaluation for precision, recall, and F-measure.

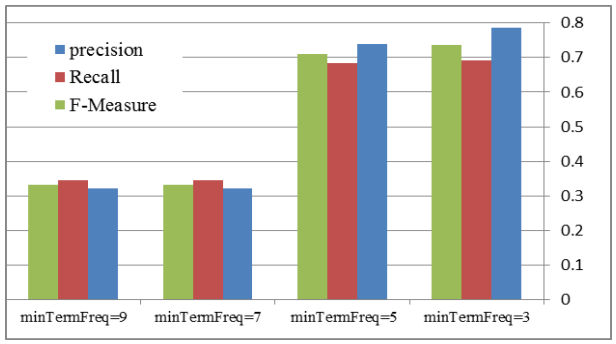

Figure 4: Results of System A applied on BCC database

Also from figures 4, 5, 6 we can tote that system (A) that applied a term pruning combining with term weighting and light stemming gives better results than the other two used systems for BBC database.

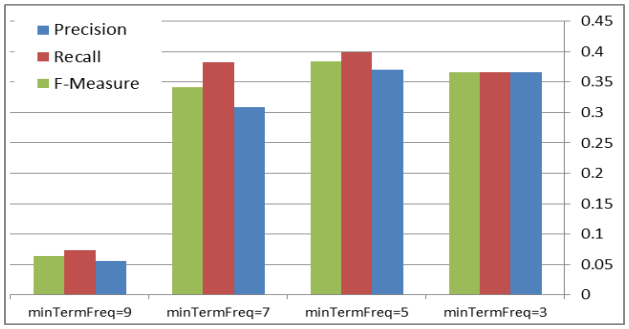

Figure 5: Results of System B applied on BBC database

From the comprehensive results of using the BBC and CNN datasets, the observation of evaluation using precision, recall, and F-measure of applying minimum term frequency at minTermFreq 3 is the best value of evaluation. For applying term weighting using (TF-IDF), it affects in evaluation positively. Light stemming in Arabic text preprocessing can improve clustering process, and this technique of morphological analysis is more appropriate than root based stemming and raw text. Performing Normalization on data, can enhance clustering process of documents and gives better evaluation than without Normalization.

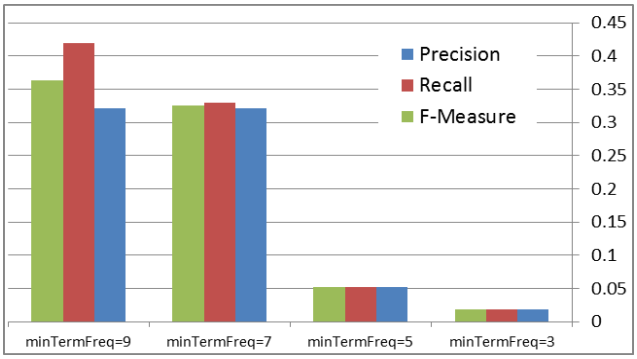

Figure 6: Results of System C applied on BBC database

Copyright © 2016. Vandana Publications. All Rights Reserved.

V. CONCLUSION

In this research we applied text preprocessing techniques to Arabic documents, then we achieve best combinations of these techniques when perform clustering algorithm. Experiments were applied to large corpora includes BBC corpus contains 1,860,786 (1.8M) words and 106,733 district keywords after stopwords removal, and CNN corpus contains 2,241,348 (2.2M) words and 144,460 district keywords after stopwords removal.

From overall experiments, to enhance clustering process of Arabic documents many adjustments should be applied to give best evaluation results: In text preprocessing, applying term pruning with small value for minimum term frequency enhance results of text preprocessing. Results depicted that minimum term frequency at minTermFreq-3 using term pruning combining with term weighting and light stemming is the best value of evaluation that is 0.737 of F-Measure for CNN dataset and 0.577 of F-Measure for BBC dataset.

REFERENCES

[1] O. M. Al-Omari, "Evaluating the effect of stemming in clustering of Arabic documents," Academic Research International, vol. 1, p. 8, 2011.
[2] M. K. Saad and W. Ashour, "Arabic text classification using decision trees," presented at the Workshop on computer science and information technologies CSIT"2010, Moscow - Saint-Petersburg, Russia, 2010.
[3] N. Sandhya, Y. S. Lalitha, V. Sowmya, K. Anuradha, and A. Govardhan, "Analysis of Stemming Algorithm for Text Clustering," International Journal of Computer Science, vol. 8.
[4] S. H. Ghwanmeh, "Applying Clustering of hierarchical K-means-like Algorithm on Arabic Language," International Journal of Information Technology, vol. 3, 2005.
[5] A. A.-D. Abdelfatah A. Yahya, "Clustering Arabic Documents Using Frequent Itemset-based Hierarchical Clustering with an N-Grams," The 4th International Conference on Information Technology. Al-Zaytoonah University, Jordan. June 4th , 2009.
[6] M. Rafi, M. Maujood, M. M. Fazal, and S. M. Ali, "A comparison of two suffix tree-based document clustering algorithms," in Information and Emerging Technologies (ICIET), 2010 International Conference on, 2010, pp. 1-5.Karachi, 14-16 June 2010.
[7] H. Froud, R. Benslimane, A. Lachkar, and S. A. Ouatik, "Stemming and similarity measures for Arabic Documents Clustering," in I/V Communications and Mobile Network (ISVC), 2010 5th International Symposium on, 2010, pp. 1-4. Rabat, Sept. 30 2010-Oct. 2010.
[8] M. S. Alkoffash, "Comparing between Arabic Text Clustering using K Means and K Mediods," International Journal of Computer Applications, vol. 51, 2012.
[9] H. Froud, A. Lachkar, and S. A. Ouatik, "Arabic text summarization based on latent semantic analysis to enhance arabic documents clustering," arXiv preprint arXiv:1302.1612, 2013.
[10] M. H. Ahmed and S. Tiun, " k-means based algorithm for islamic document clustering," 07/2013; In proceeding of: IMAN 2013.
[11] N. Shah and S. Mahajan, "Document Clustering: A Detailed Review," International Journal of Applied Information Systems 4(5):30-38. Published by Foundation of Computer Science, New York, USA, October 2012.
[12] Z. Yao and C. Ze-wen, "Research on the construction and filter method of stop-word list in text Preprocessing," in Intelligent Computation Technology and Automation (ICICTA), 2011 International Conference on, pp. 217-221, 2011.
[13] C. D. Manning and P. Raghavan, "An Introduction to Information Retrieval Draft," Online edition. Cambridge University Press. -544 p, -2009.
[14] F. Alotaiby, I. Alkharashi, and S. Foda, "Processing large Arabic text corpora: Preliminary analysis and results," in Proceedings of the second international conference on Arabic language resources and tools, pp. 78-82, 2009.
[15] M. Syiam, Z. Fayed , and M. Habib , "An Intelligent System for Arabic Text Categorization," presented at the International Journal of Intelligent Computing and Information Systems IJICIS,vol. 6, no. 1, 2006.
[16] A. A. B. Sembok T., and Abu Bakar Z., "A Rule and Template Based Stemming Algorithm for Arabic Language," International Journal of Mathematical Models and Methods in Applied Sciences, Issue 5, Volume 5, pp. 974-981, 2011.
[17] S. Khoja and R. Garside, "Stemming arabic text," Lancaster, UK, Computing Department, Lancaster University, 1999.
[18] M. a. F. Aljlayl, O. "On Arabic Search: Improving the Retrieval Effectiveness via a Light Stemming Approach", ACM Eleventh Conference on Information and Knowledge Management; 2002 November 340-347; Mclean, VA, USA, 2002.
[19] E. W. Weisstein, "Zipf"s law," MathWorld–A Wolfram Web Resource, Available: http://mathworld. wolfram. com/ZipfsLaw. html,(Date Last Accessed on Jul. 22, 2016), 2016.
[20] A. Hoonlor, "Sequential patterns and temporal patterns for text mining," Rensselaer Polytechnic Institute, 2011.
[21] C. D. Manning and P. Raghavan, "An Introduction to Information Retrieval Draft," Online edition. Cambridge University Press. -544 p, -2009.
[22] P. Han, D.-B. Wang, and Q.-G. Zhao, "The research on Chinese document clustering based on WEKA," in

Copyright © 2016. Vandana Publications. All Rights Reserved.

Machine Learning and Cybernetics (ICMLC), 2011 International Conference on. Vol. 4. IEEE, 2011. p. 1953-1957.10-13, Guilin, July 2011.
[23] Z. Qiu, C. Gurrin, A. Doherty, A. Smeaton., "Term Weighting Approaches for Mining Significant Locations from Personal Location Logs," presented at the Proceedingsin CIT(2010), 2010 IEEE 10th International Conference on, pages 20 –25, 2010, Georgia, USA.
[24] X.-B. Xue and Z.-H. Zhou, "Distributional features for text categorization," Knowledge and Data Engineering, IEEE Transactions on, vol. 21, pp. 428-442, 2009.
[25] V. Amala Bai and D. Manimegalai, "An analysis of document clustering algorithms," in Communication Control and Computing Technologies (ICCCCT), 2010 IEEE International Conference on, pp. 402-406, Ramanathapuram, 2010.

Copyright © 2016. Vandana Publications. All Rights Reserved.