EVALUATING THE EFFECT OF STEMMING IN CLUSTERING OF ARABIC DOCUMENTS

Omaia M. Al-Omari
Department of Computer Science
University of Hail
KINGDOM OF SAUDI ARABIA
omaiaomari@yahoo.com

ABSTRACT

In text mining, the concept of clustering is common and important to retrieve and categorize documents. Clustering techniques divers and many of them are applied on different languages but not on Arabic. K-means algorithm is a widely used clustering technique that seeks to minimize the average squared distance between points in the same cluster. This paper aimed to implement and evaluate the K-means algorithm on clustering of Arabic documents and estimate the effect of stemming on such clustering algorithm. The experimented work showed that the accuracy of clustering Arabic documents using the K-means algorithm varies from low to very good. The best achieved result was 69% of successful documents without stemming. Furthermore, the effect of stemming resulted in decreasing the accuracy of retrieving documents because the stemming is an abstract of a word which leads to miss-discriminating of documents. The best result scored with stemming was 55% of successful documents, when applying the same thresholds.

Keywords: clustering Arabic documents, Stemming, K-means.

INTRODUCTION

The amount of electronic text available such as electronic publications, electronic books, news articles, and web pages is increasing rapidly. As the volume of online text information increases, the challenge of extracting relevant knowledge increases as well. The need for tools that help people find, filter, and manage these resources has grown. Thus, automatic organization of text document collections has become an important research issue. A number of machine learning techniques have been proposed to enhance automatic organization of text data. These techniques can be grouped in two main categories as supervised (document classification) and unsupervised (document clustering) (Laskov et al., 2005).

Text mining is the process of deriving interesting information from text (Ikonomakis et al., 2005), usually involves the process of structuring the input text, deriving patterns within the structured data, and finally evaluating and interpreting of the output. Typical text mining tasks include text categorization, text clustering, production of granular taxonomies, document summarization, and entity relation modeling (Cohen, and Hunter, 2008).

According to (Kassab and Lamirel, 2008), clustering is the partitioning of a data set into subsets, or groups similar documents according to dominant features (Tan, 1999) in text mining and information retrieval, a weighted feature vector is frequently used to describe a document. This feature vector contains a list of the main themes or keywords along with a numeric weight indicating the relative importance of the theme or term to the document as a whole (ElWakil, 2002). Unlike data mining applications which use a fixed set of features for all analyzed items (e.g. age, income, gender, etc.), documents are described with a small number of terms or themes chosen from potentially thousands of possible dimensions.

Clustering methods, such as K-means and hierarchical clustering tend to assume that an object belongs to a particular cluster only if it is closer to at least some object in that cluster than to another object in other clusters; an approach based on how many nearest neighbours a document shares used is described in (Jarvis and Patrick, 1973). For documents, this is somewhat indirect measure of similarity turns out to be more accurate than a direct similarity measure based, like on the cosine measure. More recently, as document collections have grown larger, K-means clustering algorithm has emerged as a more efficient approach for producing clusters of documents (Karypis and Han, 2000) (Steinbach et al, 2002) K-means clustering produces a set of un-nested clusters.

Statement of problem

There are many existing classification techniques applied on Arabic texts (documents) but there are only a few for clustering. This paper aims to apply K-means algorithm to find the efficiency of clustering on Arabic documents, Furthermore a few of stemming functions are used during clustering in order to measure the effect of stemming on such clustering algorithm. The main objectives of this research are to: evaluate the result of K-mean for Arabic documents, and estimate the effect of stemming on the documents for better result.

CONTEXT AND REVIEW OF LITERATURE

Unlike document classification, document clustering is an unsupervised task, which does not require predefined categories (Kyriakopoulou and Kalamboukis, 2006). Document clustering has many application areas. In Information Retrieval, it has been used to improve precision and recall, and as an efficient method to find similar documents. Also document clustering has been used to automatically generate hierarchical grouping of documents (Koller and Sahami, 1997).

Algorithms for clustering can be categorized into two main groups as hierarchical and partitional clustering algorithms (Jain et al., 1999).

The authors in (Yoo and Hu, 2006) performed a comprehensive comparison study of various document-clustering approaches such as K-means and Suffix Tree Clustering in terms of the efficiency, the effectiveness, and the scalability. They found that the partitional clustering algorithms are the most widely used algorithms in document clustering.

The work in (Kanungo and Mount, 2002), presented an implementation of a filtering K-means clustering algorithm. It established the practical efficiency of the filtering algorithm by presenting a data-sensitive analysis of the algorithm's running time. For the running time experiments, they used two algorithms, simple *brute-force* algorithm which computes the distance from every data point to every center. The second algorithm, called *kd-center*, operates by building a *kd-tree* with respect to the center points and then uses the *kd-tree* to compute the nearest neighbor for each data point. The results showed that the filtering *K-means* clustering algorithm runs faster as the separation between clusters increases.

The authors in (Zhong and Ghosh, 2002) focused on model-based partitional clustering algorithms because, according to the authors, many advantages provided. First, the complexity is O(n), where n is the number of data documents. In similarity-based approaches, calculating the pair wise similarities requires O(n2) time. Second, each cluster is described by a representative model, which provides a richer interpretation of the cluster.

For document clustering, measures are more commonly used, since typically the documents' category labels are actually known. Examples of these measures include the confusion matrix, classification accuracy, F-measure, average purity, average entropy, and mutual information.

The research in (El Kourdi et al., 2002); deals with automatic classification of Arabic web documents. Such classification is very useful for directory search functionality, which has been used by many web portals and search engines to cope with an increasing number of documents on the web. In his paper, Naive Bayes, which is a statistical machine-learning algorithm, is used to classify non-vocalized Arabic web documents to one of five predefined categories. Cross validation experiments are used to evaluate the Naive Bayes categorizer. The data set used during these experiments consists of 300 web documents per category. The results of cross validation in the leave-one-out experiment showed that, using 2,000 roots, the categorization accuracy varies from one category to another with an average accuracy over all categories of 69 %.

The authors in (Zeng et al., 2004) performed an experiment with a training of a cluster label selection procedure. First, a set of fixed-length word sequences are created from the input. Each label is then scored with an aggregative formula combining several factors: phrase frequency, length, intra-cluster similarity, entropy, and phrase independence. The specific weights and scores for each of these factors are learnt from examples of manually prioritized cluster labels.

The authors in (Wang and Ngai, 2006) proposed, an unsupervised (clustering) approach to Chinese co-reference resolution. This approach performed resolution by clustering, with the advantage that no annotated training data is needed. The authors evaluated this approach using a corpus, which developed using standard annotation schemes; found the system achieves an error reduction rate of almost 30% over the baseline. In addition, they analyzed the performance of the system by investigating the contribution of individual features to the system; results illustrated the contribution of the new language specific features.

(Schenker et al., 2003) proposed the clustering of web document collections using two variants of the popular K-means clustering algorithm, the global K-means method which computes "good" initial cluster centers deterministically rather than relying on random initialization the graph-based and the vector-based representations. They found that allowing web documents to be represented by graphs is more robust than vectors.

(Sawaf et al., 2001) showed that statistical methods for document clustering and text classification are very promising approaches for Arabic, even without any morphological analysis. For document clustering they used a criterion based on the mutual information criterion; the algorithm for clustering documents has a predefined number of classes, where words are assigned to word classes. With this algorithm, an optimal solution is not necessary found, and the quality of classes of more complex algorithm is not better.

(Ghwanmeh, 2001) implemented a K-means -Like with hierarchical initial set on Arabic documents. This method is a combination between hierarchical and K-means-Like method. The advantage of using K-means-Like is that it allows texts to be classified quickly. The algorithm starts with an initial partition and then apply the other documents and relocate them iterately until have the final partition that have a property that it is not allowed to transfer any document from one cluster to another. The initial partition is built using hierarchical classification. He proved that clustering documents enhance precision on information retrieval systems.

METHOD

In this section, the K-means methodology is introduced as a commonly used clustering algorithm.We also introduce the functions used to enhance the methodology for applying it on Arabic. The overall methodology for clustering Arabic text is illustrated in Figure 1.

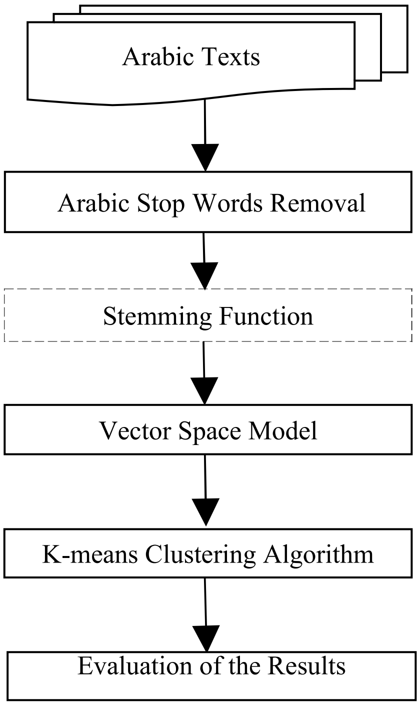
Arabic Texts
Arabic Stop Words Removal
Stemming Function
Vector Space Model
K-means Clustering Algorithm
Evaluation of the Results

Figure 1. Arabic text clustering method.

Tokenization

Given a character sequence and a defined document unit, tokenization is the task of chopping it up into pieces, called tokens, perhaps at the same time throwing away certain characters (Sebesta, 2006).

Stop Words Removal Function

Stop words are words appear frequently within texts that without conveying any particular meaning and they lose their usefulness as search terms. Therefore, removing them is better.

Stemming Function

Stemming is the process of reducing derived words to their stem, base, or root form, The stem need not be identical to the morphological root of the word; it is usually sufficient that related words map to the same stem, even if this stem is not in itself a valid root (Larkey et al., 2002).

The stemming process starts by assigning order and weight values to each letter in a given word, then generate new values by multiplying the order by the weight values for each letter in a given word, and the root will simply be the concatenation of the first three letters that have minimum product values.

Some letters in Arabic language must be given a weight equal to zero, while other letters, which can be combined and formed in the Arabic word "سألتمونيها", will be assigned weight values greater than zero. The weight values assigned to each letter are shown in (Al-Shalabi et al., 2003).

K-means Clustering Algorithm

It assigns each point to the cluster whose center is nearest. The center is the average of all the points in the cluster; so, its computes the arithmetic mean for each dimension separately over all the points in the cluster (Al-Harbi and Al-Thubaity, 2004). This step starts after removing stops words and apply stemming, with a prepared collection of documents and attempts to group them into k number of clusters.
The most usually used partitional strategy is based on the square error criterion which is the minimization of the variations within clusters, as measured by the sum of the distances from each document in one cluster to its cluster center. The cluster *k* has documents $(x_1, x_2, x_3, ..., x_{n_k})$, x_c is the center of cluster *k* and $e_k = \sum_{i=1}^{n_k} (x_i - x_c)^2$ is the variance of the cluster *k*. The square error with *k* clusters is the sum of the entire cluster variations: $E = \sum_{j=1}^{K} e_j$.

Vector Space Model

Vector space model is an algebric model for representing text documents as vectors of identifiers, such as index terms. It maps text documents to vectors and compares their angles (Chuang and Seamons, 1997).

In order to reduce the complexity of texts and make them easier to handle, the texts have to be transformed from the full text version to a document vector, which describes the contents of the text. According to (Chuang and Seamons, 1997), the text is defined as is that it is made of a joint membership of terms, which have various patterns of occurrence. Term weight is very important and critical step in building a vector space model, the parameters in calculating a weight for a document term are:

- Term Frequency (*TF*): Term Frequency is the number of times a term *i* appear in document *j* (TF_{ij}).
- Document Frequency (*DF*): Number of documents a term *i* appears in, (DF_i).
- Inverse Document Frequency (*IDF*): A discriminating measures for a term *i* in collection, i.e., how discriminating term *i* is. (The terms that appear in many documents are not very useful for distinguishing a relevant document from irrelevant one).

$(IDF_i) = log_{10} (N/ n_i)$, where:
N= number of documents in the collection.
n_i = number of document that contain the term *i*.
Then the weights are computed as follows $w_{i,j}$ *(TF-IDF):*
One of the most popular methods is based on combining two factors:

1. The importance of each index term in the document *(TF)*.
2. The importance of the index term in the collection of documents *(IDF)*.

Combining these two factors we can obtain the weight of an index term *i* as:
$w_{ij} = (TF_{ij}) (IDF_i) = (TF_{ij}) (log10) (N/ n_i)$
Where:
N= number of documents in the collection.
ni = number of document that contain the term *i*.

A fixed collection of text is clustered into groups or clusters that have similar contents. The similarity between documents is usually measured with the associative coefficients from the vector space model; we will use the cosine coefficient equals to

$$\frac{\sum_{k=1}^{t}(d_{ik} \bullet q_k)}{\sqrt{\sum_{k=1}^{t} d_{ik}^{2} \bullet \sum_{k=1}^{t} q_k^{2}}}$$

Where, $\sum_{k=1}^{t}(d_{ik} \bullet q_k)$ is the inner product, and $\sqrt{\sum_{k=1}^{t} d_{ik}^{2} \bullet \sum_{k=1}^{t} q_k^{2}}$ is the length between documents. Here d, q, i, k, t ..

FINDINGS: EXPERIMENTS AND EVALUATION

The training and testing datasets:

We had two data sets; one for training set in which we build the source system and the other is the test dataset in which we examined the *K-means* algorithm. The training data set includes 1445-Arabic document in different and separate nine categories. Categories include, computer, economics, education, engineering, law, medicine, politics, religion, and sport.

Preparing the data for evaluation

To be able to test the implemented *K-means* algorithm we generated a database that includes a table for documents from the data set and assigned each document to its category.

Experiments and evaluation environment

For the experiment we used PC Pentium 4 with Intel Mobile technology CPU 1.5 MHz, 512 Mb of RAM, Microsoft Windows XP Professional, Microsoft SQL Server 2005 database management systems, and programming language Microsoft Visual VB .NET 2005.

Evaluation

We use the Accuracy to measure the efficiency by measuring the percent of retrieved documents in an accurate category.

In Table 1 the best result in category Computer scored 0.97 of successful documents without applying stemming, otherwise with stemming the best scored result was 0.99 in category Economics. The overall percent of successful documents without stemming equals to 0.69 while with stemming equals to 0.55.

We can observe that more iterations with minimizing the difference between old and new cluster's center result in increase the accuracy of clustering.

DISCUSSION AND CONCLUSION

The experimental results showed that the clustering solution produced by the *K-means* algorithm is not stable; because we changed the initial *k* points every time we ran the system. In addition, the produced clusters facilitate examining each cluster for a clustering task. The task involves discriminating between successful and unsuccessful procedures.

Furthermore, experiments showed that *K-means* generally performed better if it selects several new centers during each iteration. Applying stemming on such clustering is not efficient because the documents must discriminate from each other to relate to the exact category; because the stemming is an abstract of word which leads to miss discriminating of documents.

Table 1. Clustering result at, cluster threshold= 1, and loop count = infinite

Category Name	number of documents	Number of successful documents without stemming	Percent	Number of successful documents with stemming	Percent
Computer	70	68	0.97	9	0.13
Economics	220	121	0.55	218	0.99
Education	68	34	0.5	39	0.58
Engineering	115	101	0.88	43	0.37
Law	97	78	0.8	61	0.63
Medicine	232	155	0.67	37	0.16
Politics	184	121	0.66	110	0.6
Religion	227	184	0.81	150	0.66
Sport	232	90	0.39	186	0.8
Overall Total	1445	953	**0.69**	853	**0.55**

ACKNOWLEDGMENTS

Researcher would like to express sincere gratitude to Dr. Qasem Al-Radaideh for his guidance, encouragement and effort, throughout this research.

REFERENCES

Laskov et al., (2005). *Learning intrusion detection: supervised or unsupervised*, Internet Measurement Conference.

Ikonomakis et al., (2005). *Text Classification Using Machine Learning Techniques,* WSEAS Transactions on Computers, Issue 8, 4: 966-974.
Cohen, K. & Hunter, L. (2008). *Getting Started in Text Mining*, PLoS Computational Biology Journal.

Kassab R. & Lamirel J. (2008). *Feature-based Cluster Validation for High-Dimensional Data*, in Proceeding 595, Artificial Intelligence and Applications, track 168.

Tan A. H. (1999). *Text Mining: the state of art and the challenges*, proceeding workshop on Knowledge Discovery from Advanced Databases, 71-76.

ElWakil M. M. (2002). *Introducing Text Mining, 9th* scientific Conference for Information Systems and Information Technology (ISIT02).

Jarvis, R. & Patrick, E. (1973). *Clustering Using a Similarity Measure Based on Shared Nearest Neighbors*, IEEE Transactions on Computers, C-22.

Karypis, G. & Han E. (2000). *Concept Indexing: a Fast Dimensionality Reduction Algorithm with Applications to Document Retrieval & Categorization*, CIKM.

Steinbach et al., (2002). *A Comparison of Document Clustering Algorithms*, KDD Text Mining Workshop.

Kyriakopoulou, A. & Kalamboukis, T. (2006). *Text Classification Using Clustering*, 10th European Conference on Principles and Practice of Knowledge Discovery in Databases (PKDD), 28-38.

Koller, D. & Sahami, M. (1997). *Hierarchically Classifying Documents Using Very Few Words,* proceeding of 14th International Conference on Machine Learning, 170–178.

Jain et al., (1999). *Data Clustering: A Review*, ACM, Computing Surveys, 31: 264–323.

Yoo, I. & Hu X. (2006). *A Comprehensive Comparison Study of Document Clustering for a Biomedical Digital Library MEDLINE*, JCDL, ACM.

Kanungo, T. & Mount D. (2002). *An Efficient K-means Clustering Algorithm Analysis and Implementation,* IEEE Transactions on Pattern Analysis and Machine Intelligence, 24.

Zhong, S. & Ghosh J. (2002). *A Comparative Study of Generative Models for Document Clustering,* SIAM Knowledge and Information Systems.

El Kourdi et al., (2002). *Automatic Arabic Document Categorization Based on the Naïve Bayes Algorithm*, COLING Computational Approaches to Arabic Script-based Languages, 51-58.

Zeng et al., (2004). *Learning to Cluster Web Search Results,* proceeding of the 27th ACM International Conference on Research and Development in Information Retrieval, 210-217.

Wang, C. & Ngai, G. (2006). *A Clustering Approach for Unsupervised Chinese Co-reference Resolution,* proceeding of the Fifth SIGHAN Workshop on Chinese Language Processing, 40–47.

Schenker et al., (2003). *A Comparison of Two Novel Algorithms for Clustering Web Documents,* proceeding of the Second International Workshop on Web Document Analysis, 71-74.

Sawaf et al., (2001). *Statistical Classification Methods for Arabic News Articles,* Arabic Natural Language Processing workshop.

Ghwanmeh S. H. (2001). *Applying Clustering of Hierarchical K-means-like Algorithm on Arabic Language*, International Journal of Information Technology, 3.

Sebesta R. W. (2006). *Concepts of programming languages*. Boston: Pearson/Addison-Wesley, 7th edition pp.177.

Larkey et al., (2002). *Improving Stemming for Arabic Information Retrieval: Light Stemming and Co-occurrence Analysis*, in SIGIR, 275-282.

Al-Shalabi et al., (2003). *New Approach for Extracting Arabic Roots,* Paper presented at the *International Arab Conference on Information Technology.*

Al-Harbi, S. & Al-Thubaity, A. (2004). *The Use of Modified K-means Algorithm in Bridging the Digital Gap*.

Chuang, H. & Seamons, K. (1997). *Document ranking and the vector-space model***,** IEEE, 14: 67 – 75.