Clustering Arabic Documents Using Frequent Itemset-based Hierarchical Clustering with an N-Grams

Dr. Haytham S. Al-sarrayrih
Mutah University, Jordan
haytham@mutah.edu.jo

Dr. Riyad Al-Shalabi
Arab Academy for Banking and Financial Sciences, Jordan
rshalabi@aabfs.org

ABSTRACT

The rapid progress of computers and network technologies makes it easy to collect and store a large amount of unstructured or semi-structured texts such as text documents, WebPages, HTML/XML archives, E-mails, and text files. In this paper we used one of the powerful clustering algorithm "Frequent Itemset-based Hierarchical Clustering (**FICH**)" to cluster Arabic. We conducted our experiments on 600 Arabic documents using N-grams based on word level, Trigrams and Quadgrams and we got a promising results.

Keywords
Clustering, Text Classification, N-gram, Hierarchal clustering, F-measure, Recall, and Precision.

1. Introduction

The availability of large amount of information in an electronic format from different sources in different formats and the need of organizations to benefit from these information encourage researchers to develop applications to handle these information, Clustering plays an important role in providing intuitive navigation and browsing techniques by organizing large collection of documents into a small number of meaningful groups. Since Arabic is used by more than 265 millions of Arabs, also it is understood by more than one billion of Muslims worldwide, as the Muslims' holy book (the Koran) is written in Arabic. Arabic documents became very popular on an electronic format, so the need for clustering documents became very necessary.

There is a growing interest in tools that can help finding information included in the text documents because of the availability of huge amount of data in an electronic format. This availability refers to the fact that most organizations keeps its data in a text form and the internet is a large source of information stored in text, so we need to organize these texts to help organizations and researchers to find appropriate information, when encountering a very large number of text documents, the most common technology to use is Information Retrieval technology.

The traditional information retrieval became inadequate for the increasingly vast amount of text data. Without knowing what could be done with text documents, it is difficult to form effective queries or

analysis or extraction of useful information from these texts.

Text Classification (TC_f) is a text mining method, that predicts a class for each text document. Syiam et al. [15] defined classification as a process of classifying documents into a predefined set of classes based on their content, an example to label each incoming news story with a topic like "sports" or "politics".

There are two main approaches of TC_f; manual and automatic classification. Manual classification classifies documents manually, and it is accurate but time consuming. Automatic classification uses tools or techniques to classify the documents automatically and there are many classification techniques such as: Naive Bayesian Classification and k-Nearest Neighbor.

Text Clustering (TC_t) is opposed to document classification, it does not need predefined classes. TC_t has been investigated for use in a number of different areas of TM and IR. Initially, TC_t was investigated for improving the precision or recall in IR systems. More recently, clustering has been proposed for use in browsing a collection of documents or in organizing the results returned by a search engine in response to a user's query [9]. So clustering used to group similar documents (similar contents) and the output of clustering is a set of clusters each of them consists of a number of documents. According to Hotho et al. [5] the quality of clustering is considered better if the contents of the documents within one cluster are more similar and between the clusters more dissimilar.

Partitional clustering algorithms group the data into un-nested (one level) partitions that usually optimize a clustering criterion locally. Popular partitional clustering algorithm applied to the domain of text documents are *k*-means [8]. While Hierarchical Clustering produces nested partitions of data by splitting (divisive) or merging (agglomerative) clusters based on the similarity among them. Divisive (Top Down) starts with one cluster of all data points and at each iteration split the most appropriate cluster until a stopping criterion such as a requested number *k* of clusters is achieved. In agglomerative (Bottom up) each item starts as an individual cluster and at each step, the most similar pair of clusters are merged [6].

2. Literature Review

Al-Shalabi and Obeidat, (2008) [1], built a classification which includes two classifiers: the first was based on using N-grams as document indexing technique and the second used single terms for document indexing. They trained and tested their system using Arabic corpus from online Arabic newspapers archives. The result of their work showed that using N-gram produced better accuracy than using single terms for indexing.

Beil et. al, (2003) [3], presented a new approach to address the issues of traditional clustering algorithms such as, high dimensionality of the data, very large size of the databases and understandability of the cluster description, in their paper they proposed a novel approach FIHC (Frequent Itemset-based Hierarchical Clustering), the intuition of their approach was that there were some frequent itemsets for each cluster in the document set, and different clusters shared few frequent itemsets, which were sets of words that combined together in some minimum fraction of documents in a cluster. In this paper we applied such algorithm.

Wang, et. al, (1999) [11], introduced a new criterion for clustering transactions using frequent itemsets based on the notion of large items (items contained in some minimum fraction of transactions in a cluster) without using any measure of pairwise similarity. In their paper, term "transaction" referred to a set of items in general such as a set of terms in an article. They compared their algorithm to two algorithms, the traditional hierarchical clustering, and the link-based hierarchical clustering. The result of their comparison experiments showed that their algorithm made only 2 or 3 scans of the database and

the execution time scaled up linearly with the size of the database and the processing order of transactions did not have a major impact on the clustering, and they concluded that their algorithm was effective.

3. The Proposed System

Our proposed system consists of the following steps: Text preprocessing, morphological Analysis, Vector Space Model, and the final step is applying FIHC for Arabic Documents, flowchart of the proposed system is shown in Figure 1 [17].

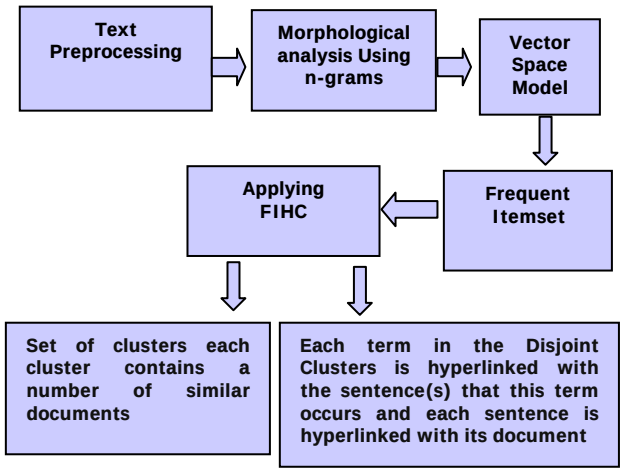

Figure 1: System Flowchart

3.1 Text Preprocessing

The text preprocessing phase include the following steps: Normalization of Arabic, Tokenization, and Stop Words Removal

3.1.1 Normalization of Arabic

Before the representation of data is performed, a number of normalization steps are usually performed to reduce the number of terms used. In this work we will employ the following normalization steps :

- Remove non letters
- Replace initial آ , إ or أ with bare alif ا
- Replace final ة with ه
- Remove ال from the beginning of the word
- Remove ين ، ون ، ات from the end of the word.

3.1.2 Tokenization

This step is the first and one of the most critical tools used to analyze text linguistically. It breaks strings of characters, words, and punctuation into tokens during the indexing process.

The goal of the tokenization process is defined by Bennett et. al in [5] as to determine sentence boundaries, and to separate the text into a stream of individual tokens (words) by removing extraneous punctuation.

3.1.3 Stop Words Removal

Every language has its own stop words. In English, stop words include articles such as “the, a, and an” and demonstratives like “this,” “that” and “those.” Removing these commonly occurring words from indices reduces the number of words each search term must be compared against, significantly improving query response time without affecting accuracy. Likewise, in Arabic Stop Words includes any word that is not considered part of speech, i.e. noun or verb (including prepositions (الى، عن، في...), demonstratives (هذا، هذه، هذان،...), special characters ($,%,&,…), adverbs (فوق، تحت،...), … etc).

3.2 Morphological analysis using N-grams

N-grams is an N-character slice of a string, Cavnar and Trinkle [4] defined N-gram as "the term that can include the notion of any co-occurring set of characters in a string". Al-Shalabi and Obeidat [1] defined N-grams as "a subsequence of N items from given sequence, it could be thought as window of length N moves over the text, the contents of this window is the N-gram". Mayfield and McNamee [16], indicated that the N-grams method provides better retrieval precision and recall performance than affix-removal. The N-grams is useful in a wide variety of natural language processing applications, including text compression, error detection and correction, language identification, text categorization, text searching and retrieval [7]. The work of Xu et al. [12] used N-grams with and without stemming

for text searching. Their results indicated that the use of Tri-grams combined with stemming improved the performance of search retrieval. According to Al-Shalabi and Obeidat [1] N-grams could be character level, word level, or even statement level depending on the application. 1-Gram means that N-gram with length 1 and is called Unigram, 2-gram is called Digram (or Bigram), 3-Gram is called Trigram, 4-Gram is called Quadgram. In our work we will use N-grams at word level and at character level with Trigram and Quadgram. The Trigram for the word "يستطيع" are (يست،ستط،،تطي،طيع), in general a word of length *l* has *l*-2 Trigram, and *l*-3 Quadgram. After this step we will insert all N-grams (frequent itemset) with its weight using TF.IDF weighting technique into a Vector Space Model (VSM).

3.2.1 Term Frequency × Inverse Document Frequency Weighting (TF×IDF)

TF×IDF weighting is the most common technique used for term weighting, It's defined as the logarithm of the ratio of number of documents in a collection to the number of documents containing the given word [14].

In this technique, the weight of term *i* in document **d** is assigned proportionally to the number of times the term appears in the document, and in inverse proportion to the number of documents in the collection in which the term appears.

$$W_i = tf_i \cdot \log (N/n_i)$$

Where N is the total number of documents in the document corpus and n_i as the number of documents in the collection where term *i* appears.

3.3 Vector Space Model (VSM)

The vector space model is a widely used method for document representation in information retrieval. In this model, each document is represented by a feature vector ***d*** as follows: d = $(w_1,w_2,\ldots,w_i)$ where w_i is the weight of i-th term of document *d*, the weight is the measure that indicates the statistical importance of corresponding words [17].

3.4 Applying (FIHC) Technique to Arabic Documents

The result of this step will be a set of clusters, each cluster contains a number of similar documents, and each cluster label is hyperlinked with its sentences that occur in the collection.

3.4.1 Frequent Itemset-based Hierarchical Clustering (FIHC)

FIHC algorithm works in five steps: constructing initial clusters, making clusters disjoint, build a cluster tree, prune the cluster tree if needed, and sibling merging. For more details please see [3].

4. Experimental Evaluation

The experimental evaluation of the FIHC algorithm applied on Arabic Documents. We developed a C # code and run it on core 2 due machine with 2 GB RAM [17].

4.1 Data Sets

We used a collection of 600 Arabic documents built in house because of the lack of availability of publicly Arabic corpus. This collection consisted of six classes as described in table 1. In this collection, documents are single labeled [17].

Classes	Number of documents
Art	**90**
Economy	**100**
Science	**100**
Agriculture	**100**
Politics	**100**
Health	**110**

Table 1: Data Set

4.2 Evaluation

We employed the F-measure to evaluate the accuracy of the clusters resulted, because F-measure is a standard method for evaluating Hierarchical Clustering. Each cluster is treated as if it were the

result of a query and each class as if it were the relevant set of documents for a query [17].

Recall is defined as the fraction of relevant documents that are retrieved. *Precision* is the portion of the retrieved documents that are actually relevant, and *F-measure* for natural class K_i and cluster C_j are calculated as follows :

$$Recall(K_i, C_j) = \frac{n_{ij}}{|K_i|}$$

$$Precision(K_i, C_j) = \frac{n_{ij}}{|C_i|}$$

where n_{ij} is the number of members of class *Ki* in cluster *Cj* .

$$F(K_i, C_j) = \frac{2 * Recall(K_i, C_j) * Prescision(K_i, C_j)}{Recall(K_i, C_j) + Prescision(K_i, C_j)}$$

$F(K_i;C_j)$ represents the quality of cluster C_j in describing class K_i. While computing $F(Ki;C_j)$ in a hierarchical structure, all the documents in the subtree of C_j are considered as the documents in C_j . The *overall F-measure*, $F(C)$, is the weighted sum of the maximum F-measure of all the classes as defined below:

$$F(C) = \sum_{K_i \in K} \frac{|K_i|}{|D|} \max c_j \in C \{F(K_i, C_j)\}$$

where K denotes the set of natural classes; C denotes all clusters at all levels; $|Ki|$ denotes the number of documents in class K_i; and $|D|$ denotes the total number of documents in the data set.

Taking the maximum of $F(K_i;C_j)$ can be viewed as selecting the cluster that can best describe a given class, and $F(C)$ is the weighted sum of the F-measure of these best clusters. The range of $F(C)$ is [0,1]. A larger $F(C)$ value indicates a higher accuracy of clustering.

We conducted our experiments on the 600 Arabic documents with six natural classes as in table 1, and we experimented different values for GS and CS, and the best result we got when we used GS = 3 and CS = 8 as shown in table 2.

# of natural Classes	# of Clusters	Overall F-measure (GS: 3%, CS: 8%)		Overall F-measure (GS: 8%, CS: 15%)		Overall F-measure (GS: 10%, CS: 25%)		Overall F-measure (GS: 15%, CS: 30%)	
		3	4	3	4	3	4	3	4
4 Classes	**4**	**0.52**	**0.70**	**0.52**	**0.54**	**0.49**	**0.60**	**0.51**	**0.54**
6 Classes	**4**	**0.38**	**0.43**	**0.34**	**0.32**	**0.38**	**0.36**	**0.30**	**0.29**

Table 2: Experimental Overall F-measure [17]

After that we did our experiments using 3 for GS and 8 for CS for all available natural classes for different number of clusters and we got the accuracy as shown in table 3.

Data Set Number of natural Classes	# of Clusters	Overall F-measure (GS: 3%, CS: 8%)		
		Tri-grams	Quad-grams	Word
4 Classes	4	0.48	0.70	0.75
	6	0.53	0.68	0.74
	8	0.63	0.64	0.73
	12	0.57	0.58	0.53
	16	0.53	0.54	0.41
6 Classes	4	0.38	0.43	0.43
	6	0.37	0.49	0.50
	8	0.39	0.52	0.53
	12	0.41	0.50	0.51
	16	0.40	0.49	0.46

Table 3: Overall F-measure for FIHC (Arabic Languages) [17]

Table 4 shows the accuracy of FIHC algorithm for European languages with four and six natural classes.

# of Natural Classes	# of Clusters	FIHC
4 Classes	3	0.62
	15	0.52
	30	0.52
	60	0.51
6 Classes	3	0.45
	15	0.42
	30	0.41
	60	0.41

Table 4: Overall F-measure for FIHC (European Languages)

Because of the lack of clustering methods applied to Arabic documents we compared our results with the results that done on European languages, and from table 3 we can conclude that N-grams based on word level gives better accuracy than Quadgrams and Trigrams for both 4 and 6 classes and Quadgrams gives better accuracy than Trigrams for both 4 and 6 classes. Figure 2 shows the overall F-measure for 4 classes with 4, 6, 8, 12, and 16 clusters, from this figure we can conclude that the best result was with 4 clusters using word and Quadgrams, and 8 clusters with trigrams. Figure 3 shows the overall F-measure for 6 classes with 4, 6, 8, 12, and 16 clusters, from this figure we can conclude that the best result was with 8 clusters using word and Quadgrams, and 12 clusters with Trigrams [17].

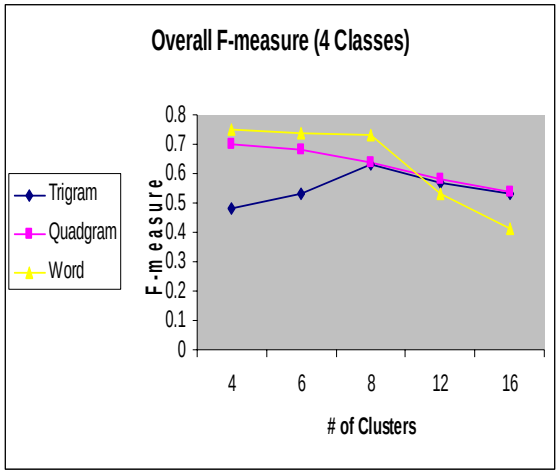

Figure 4.2: Overall F-measure for 4 Natural Classes [17]

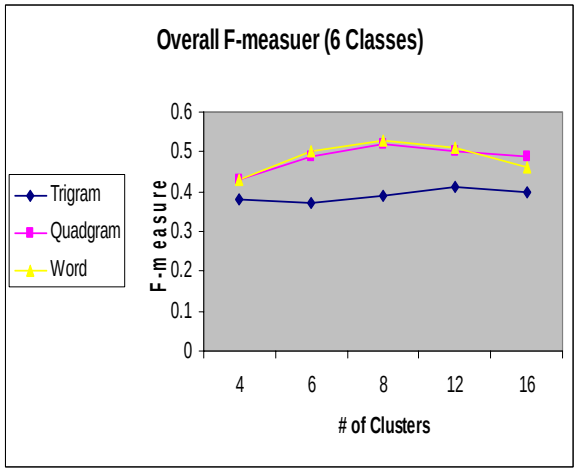

Figure 4.3: Overall F-measure for 6 Natural Classes [17]

5. Conclusion

In this paper we applied Frequent Itemset-based Hierarchical Clustering (FIHC) algorithm using N-grams on Arabic documents which was implemented on

European languages using Porter stemmer, and we built browsing system to navigate collection of Arabic documents. Our experiments conducted on a collection of 600 Arabic documents built in-house because of lack of Arabic corpus that available publicly, our collection is consisted of six natural classes (Agriculture, Art, Economics, Politics, Health, and Science).

We implemented our system using C# to execute FIHC algorithm on Arabic Documents and to build browsing system.

Due to the lack of availability of other clustering methods applied on Arabic documents, a comparison has been made with the work done on European languages, we conducted our experiments using N-grams based on word level and character level Trigrams and Quadgrams, and we experimented different values for GS and CS and the best result we got was when we used 3 for GS and 8 for CS, and for these values of GS and CS we experimented different number of clusters 4, 6, 8, 12, and 16 for four natural classes and six natural classes.

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 we got accuracy of 0.75 for four natural classes for 4 clusters, and we 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.

Comparing to the best result with European languages (0.62), we got (0.70), found that our results are promising using N-grams based on word level for Arabic language [17].

References

[1] Al-Shalabi R., Obeidat R., "***Improving KNN Arabic Text Classification with N-grams Based document indexing***", In Proceedings of the sixth international conference on informatics and systems. Egypt 2008.

[2] Arimura H., Abe J., Fujino R., Sakamoto H., Shimozono S., Arikawa S., ***"Text Data Mining: Discovery of Important Keywords in the Cyberspace"***, In Proc. IEEE Kyoto Int'l Conf. Digital Library, 2001.

[3] Biel F., Wang K., and Ester M., "***Hierarchical Document Clustering Using Frequent Itemsets***" , SIMS International Conference on data mining, SDM'2003 . USA, 2003 .

[4] Cavnar W. and Trenkle J., "***N-gram-based Text Categorization***", Proceedings of SDAIR-94, 3rd Annual Symposium on Document Analysis and information retrieval, 1994 .

[5] Hotho A., Nurnberger A., and Paab G., "***A Brief Survey of Text Mining***" , *GLDV*-Journal for Computational Linguistics and Language Technology, PP. 19-62, 2005.

[6] Jain A., Murty M., and Flynn P., "***Data Clustering: A Review***", ACM Computing Surveys, Vol. 31, No. 3, pp. 264–323, 1999.

[7] Majumder P., Mitra M., Chaudhuri B. "***N-gram: a language independent approach to IR and NLP***", Proc. International Conference on Universal Knowledge and Language (ICUKL- 2002), India, 2002.

[8] Ozgur A., "***Supervised And Unsupervised Machine Learning Techniques For Text Document Categorization***" , Master Thesis, Bogazici University, Istanbul, 2004.

[9] Steinbach M., Karypis G., and Kumar V., "***A Comparison of Document Clustering Algorithms***" In Proceedings of the Text Mining

Workshop for The Sixth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Boston, MA.2000.

[10] Tan A. , ***"Text mining: The state of the art and the challenges***", In Proc of the Pacific Asia Conf on Knowledge Discovery and Data Mining, PAKDD, Workshop on Knowledge Discovery from Advanced Databases, PP 65–70, 1999.

[11] Wang K., Xu C., and Liu B., "***Clustering transactions using large items***" , in proceedings CIKM'99 , 1999.

[12] Xu J., Fraser A., Weischedel R., ***"Empirical studies in strategies for Arabic Retrieval"***, in Proceedings of the 25th annual international ACM SIGIR conference on research and development in information retrieval, SIGIR '02. ACM, NY, PP 269-274, 2002 .

[13] Hearst M., http://www.ischool.berkeley.edu/~hearst/text-mining.html, 2003, accessed 6/2008.

[14] Polettini N., "***The Vector Space Model in Information Retrieval - Term Weighting Problem***", 2004 ,

http://sra.itc.it/people/polettini/PAPERS/Polettini_Information_Retrieval.pdf, accessed 1/2008.

[15] Syiam M., Fayed Z., Habib M., "***An Intelligent System for Arabic Text Categorization***", IJICIS, Vol.6, No. 1, 2006 .

[16] Mayfield M. and McNamee P., "***Indexing using both N-grams and words***", Proceedings of the seventh Text Retrieval Conference (TREC-7), Gaithersburg, MD: National Institute of Standards and Technology, 1998.

[17] A-sarrayrih H. and Al-Shalabi R., "***Clustering Arabic Documents Using Frequent Itemset-based Hierarchical Clustering with an N-Grams***", PHD Thesis, "Arab Academy for Banking and Financial Sciences", 2008.