COMPARISON OF HIERARCHICAL AGGLOMERATIVE ALGORITHMS FOR CLUSTERING MEDICAL DOCUMENTS

Fathi H. Saad[1], Omer I. E. Mohamed[2], and Rafa E. Al-Qutaish[2]

[1] National Health Services (NHS), London, UK
f_miligi@yahoo.com
[2] Al Ain University of Science and Technology, Abu Dhabi, UAE
omar.mohamed@aau.ac.ae, rafa.alqutaish@aau.ac.ae

ABSTRACT

Extensive amount of data stored in medical documents require developing methods that help users to find what they are looking for effectively by organizing large amounts of information into a small number of meaningful clusters. The produced clusters contain groups of objects which are more similar to each other than to the members of any other group. Thus, the aim of high-quality document clustering algorithms is to determine a set of clusters in which the inter-cluster similarity is minimized and intra-cluster similarity is maximized. The most important feature in many clustering algorithms is treating the clustering problem as an optimization process, that is, maximizing or minimizing a particular clustering criterion function defined over the whole clustering solution. The only real difference between agglomerative algorithms is how they choose which clusters to merge. The main purpose of this paper is to compare different agglomerative algorithms based on the evaluation of the clusters quality produced by different hierarchical agglomerative clustering algorithms using different criterion functions for the problem of clustering medical documents. Our experimental results showed that the agglomerative algorithm that uses I1 as its criterion function for choosing which clusters to merge produced better clusters quality than the other criterion functions in term of entropy and purity as external measures.

KEYWORDS

Medical Documents, Clustering, Hierarchical Agglomerative Algorithms

1. INTRODUCTION

Large quantities of information about patients and their medical conditions are available within the clinical documents. Therefore, to enhance the understanding of disease progression and management, an evaluation of stored clinical data, when performed, may lead to the discovery of trends and patterns hidden within the data. Methods are needed to facilitate searching such large quantities of clinical documents [1]. Clustering the medical documents into small number of meaningful clusters is one of the methods that facilitate discovering trends and patterns hidden within these documents, because dealing with only the cluster that will contain relevant documents should improve effectiveness and efficiency. Therefore, the clusters must be of high-quality since further processing will be done based on the produced clusters.

Document clustering is used in various areas to perform different tasks, such as, improving the precision and recall for information retrieval systems, documents collection browsing [2] and

DOI : 10.5121/ijsea.2012.3301

automatic generation of documents' hierarchical clusters [3]. The hierarchical clustering produces nested sequence of partitions between one cluster at the top and clusters of individual points at the bottom [4]. The output of hierarchical clustering is a tree that graphically displays the merging process and the intermediate clusters called '*dendrogram*'.

One of the agglomerative algorithm features is that it is a bottom-up approach since it begins with the objects as individual clusters and then repeatedly merges any two most similar clusters until a single all-inclusive cluster is acquired [1, 2, 4, 5, 6, 16, 19, 21]. The pair of clusters to be merged at each step are determined by:

1. Different clustering criterion functions such as *I1*, *I2*, *ε1*, *G1*, *H1* and *H2* which can be converted into cluster selection scheme for agglomerative clustering algorithm [16].
2. Traditional selection schemes such as *single-link*, *complete-link* and *UPGMA*.
3.

This paper focuses on comparing different agglomerative algorithms that use criterion functions to choose which clusters to be merged for the problem of clustering medical documents. The comparison will be based on evaluating the quality of the produced clusters using two external similarity measurements which are entropy and purity. Six criterion functions will be included in this comparison *I1*, *I2*, *ε1*, *G1*, *H1* and *H2* in addition to the three traditional selection schemes *single-link*, *complete-link* and *UPGMA*.

The rest of this paper will be organized as follows. Section 2 provides some information on vector-space model which used to represent the documents. Section 3 describes the calculation of the similarity between documents. Section 4 describes the six criterion functions as well as three traditional selection schemes. Section 5 provides detailed information about the medical documents used in the experiment. Section 6 explains the cluster quality measures used for the evaluation of produced clusters. Section 7 provides detailed description of the methodology. Summary of the experimental results obtained are described in section 8. Section 9 describes the comparison and discussion of some observations from the experimental results. Finally, section 10 which provides concluding remarks.

2. THE VECTOR SPACE MODEL

The different clustering algorithms used in the comparison use vector-space model for representation of the documents. In this model, the document d in the term space is considered to be a vector. The *Term-Frequency* (TF) vector represents each document,

$$d_{tf} = (tf_1, tf_2, \ldots, tf_n),$$

where tf_i denotes the frequency of the *i*th term in the document. But some terms appear frequently in many documents have limited discrimination power, so these terms must be de-emphasized [7]. To refine this model, weighting each term based on its *Inverse Document Frequency* (IDF) in the documents. This is calculated by multiplying the frequency of each term *i* by $\log(N/df_i)$, where *N* is the total number of documents in the collection, and df_i is the number of documents that contain the *i*th term, so the *tf-idf* representation of the document is defined as:

$$dtfidf = (tf1 \log(N/df1), tf2 \log(N/df2), \ldots , tfm \log(N/dfm))$$

It normalizing the length of each document vector, so that it is of unit length (i.e. $||dtfidf|| = 1$) to account for different lengths' documents.

3. SIMILARITY MEASURES

When clustering algorithm is used, the similarity between two documents must be measured. There are many similarity measures such as *Tanimoto* [17], *cosine* [1, 4, 7, 18], *correlation coefficient* [15, 32], *Euclidean distance* [6, 7, 15, 32] and *extended Jaccard coefficient* [15, 32]. In the vector-space model, the *cosine* similarity is the most commonly used method to compute the similarity between two documents d_i *and* d_j [1, 4], which is defined as:

$$\cos(d_i, d_j) = \frac{d_i^t d_j}{||d_i|| \, ||d_j||}$$

This equation will become $cosine(d_i, d_j) = d_i^t d_j$ if the documents vectors are of unit length; and in this case this measure becomes one if the documents are identical, and zero if there is nothing in common between them. The composite vector D_S of set of documents S and their corresponding vector representation is the sum of all documents vectors in S [4, 7, 19] and defined as:

$$D_S = \sum_{d \in S} d$$

The centroid vector C_S is the vector obtained by averaging the weights of different terms in the set of documents S [4, 7, 19] and defined as:

$$C_S = \frac{D_S}{|S|}$$

4. CLUSTERING CRITERION FUNCTIONS

As we mentioned earlier, the main objective of this paper is to compare different agglomerate algorithms that use different criterion functions in order to find the best one that produce high-quality clusters for medical documents. Karypis [8] state clearly in his technical report that ”*the choice of the right criterion function depends on the underlying application area, and the user should perform some experimentation before selecting one appropriate for his/her needs*”.

There are six clustering criterion functions that can be classified in to four groups: *internal*, *external*, *graph-based* and *hybrid*. The three traditional selection schemes *single-link*, *complete-link* and *UPGMA* can only be used within the context of agglomerative clustering.

4.1. Internal Criterion Functions

This kind of criterion functions does not take into account the documents assigned to different clusters. Such internal criterion function focuses in creating a clustering solution that optimizes a particular criterion function which is defined over the documents that are part of each cluster [1, 4, 7].

The first internal criterion function maximizes the average pairwise similarities summation between the documents assigned to each cluster, and the size of each cluster determines the weight. Measuring the similarity between two documents using the cosine function then the clustering solution must optimize the following criterion function [4]:

$$\text{maximize} \quad I_1 = \sum_{r=1}^{k} n_r \left(\frac{1}{n_r^2} \sum_{d_i, d_j \in S_r} \cos(d_i, d_j) \right) = \sum_{r=1}^{k} \frac{\| D_r \|^2}{n_r}$$

where n_r is the number of documents in cluster *r*. and D_r is the composite vector of cluster *r*. The second criterion function attempts to find the clustering solution that maximizes the similarity between each document and the cluster's centroid. Using the cosine function to measure the similarity between a document and a centroid, the criterion function becomes [7]:

$$\text{maximize } I_2 = \sum_{r=1}^{k} \sum_{d_i \in S_r} \cos(d_i, C_r) = \sum_{r=1}^{k} \sum_{d_i \in S_r} \frac{d_i{}^t C_r}{\| C_r \|} = \sum_{r=1}^{k} \frac{D_r{}^t C_r}{\| C_r \|} = \sum_{r=1}^{k} \frac{D_r{}^t D_r}{\| D_r \|} = \sum_{r=1}^{k} \| D_r \|$$

where C_r is centroid of the cluster *r*.

4.2. External Criterion Function

The external criterion functions emphasises on optimizing a function based on the difference between various clusters. For many problems this criterion function has trivial solutions that can be achieved by assigning to the first *k - 1* clusters a single document that shares very few terms with the rest, and then assigning the rest of the documents to the *k*th cluster [4].

The aim of the external criterion function is to minimize the cosine between the centroid vectors of each cluster and centroid vector of the entire collection. Thus, this will increase the angle between them as much as possible. Based on the cluster size, the contribution of each cluster is weighted. The external criterion function was motivated by multiple discriminant analysis and is similar to minimizing the trace of the between-cluster scatter matrix [7], the external criterion function defined as:

$$\text{minimize } \varepsilon_1 = \sum_{r=1}^{k} n_r \frac{D_r{}^t D}{\| D_r \|}$$

where *D* is the composite vector of the entire document collection.

4.3. Hybrid Criterion Functions

The internal criterion tried to maximize various measures of similarity over the documents in each cluster, and the external criterion tried to minimize the similarity between the cluster's documents and the collection. To simultaneously optimize multiple individual criterion functions, various clustering criterion function can be combined to define a set called *hybrid* criterion functions [4, 7].

There are two hybrid criterion functions. The first one obtained by combining criterion I_1 with ε_1 and defined as:

$$\text{maximize } H_1 = \frac{I_1}{\varepsilon_1} = \frac{\sum_{r=1}^{k} \| D_r \|^2 / n_r}{\sum_{r=1}^{k} n_r D_r{}^t D / \| D_r \|}$$

The second is obtained by combining I_2 with ε_1 and defined as:

$$\text{maximize } H_2 = \frac{I_2}{\varepsilon_1} = \frac{\sum_{r=1}^{k} \| D_r \|}{\sum_{r=1}^{k} n_r D_r{}^t D / \| D_r \|}$$

4.4. Graph-Based Criterion Function

To view the relations between the documents, the Similarity graph is used as an alternative way. The similarity graph G_s for a given collection of n documents is obtained by modelling each document as a vertex. The Graph-Based criterion method can be used to view the clustering process as partitioning the documents into groups by minimizing the edge-cut of each partition [4, 7]. If this criterion function used the cosine function to measure the similarity between the documents, then it will be defined as:

$$\text{minimize}\,\mathrm{G1} = \sum_{r=1}^{k} \frac{D_r{}^t D}{\| D_r \|^2}$$

4.5. Criterion Functions for Agglomerative Algorithms

This algorithm uses each document as an individual clusters, and then repeatedly joins the most similar two clusters using definition of cluster similarity or distance until there is only one cluster. Thus, agglomerative algorithm builds the tree from bottom toward the top. There are three cluster selection scheme summarized as follows

Single-link scheme is used to measure the distance between two clusters; that is, by taking the *minimum* of the distances between all pairs of patterns drawn from the two clusters (one pattern from the first cluster, the other from the second). The single-link algorithm suffers from chaining effects and it produces clusters that are straggly or elongated in case of noisy patterns, but it can works well on data sets that are well-separated, chain-like and have concentric clusters [6, 21]. The similarity between two clusters S_i and S_j is given by:

$$Sim_{\text{single-link}}(S_i, S_j) = \max_{d_i \in S_i, d_j \in S_j} \{\cos(d_i, d_j)\}$$

Complete-link scheme measures the distance between two clusters by the *maximum* of all pairwise distances between patterns in two clusters. The complete-link algorithm produces tightly bound or compact clusters but it cannot extract concentric clusters [6, 21] the similarity between two clusters S_i and S_j is computed as follow:

$$Sim_{complete-link}(S_i, S_j) = \min_{d_i \in S_i, d_j \in S_j} \{\cos(d_i, d_j)\}$$

The UPGMA scheme measures the distance between two clusters by computing the *average* distance between all pairs of documents in the two clusters. Same as complete linkage, this method performs quite well when the objects form distinct 'clumps' [6, 11, 21]. The similarity between two clusters S_i and S_j is computed as follow:

$$sim_{UPGMA}(S_i, S_j) = \frac{1}{n_i n_j} \sum_{d_i \in S_i, d_j \in S_j} \cos(d_i, d_j) = \frac{D_i{}^t D_j}{n_i n_j}$$

4. DATA SET

Five datasets have been used as in Table 1. Information about colonoscopy procedures is included in the data set which is called ‘colo’. Colonoscopy is refers to the passage of the ‘colonoscope’ to the entire large intestine, from the lowest part which is the caecum through the colon to the small intestine. Medical problems such as bleeding, colon cancer, polyps, colitis… etc. can be checked by colonoscopy [22, 23, 24, 25]. The data sets ‘Endo_1’, ‘Endo_2’ and ‘Endo_3’ contain information about upper GI endoscopy procedures. Upper GI endoscopy sometimes called EGD (esophagogastroduodenoscopy). However, from the ‘mouth’ through the ‘oesophagus’ to the ‘stomach’ and part of the small ‘intestine’ (duodenum) is called the upper gastrointestinal tract. Endoscopy is a visual examination of the upper intestinal using endoscope. The aim of endoscopy is to discover the reason of swallowing difficulties, abdominal pain, chest pain … etc. [26, 27]. Information about the ‘sigmoidoscopy’ procedure is included in the last data set which is called ‘Sigmoid’. Also, the visualisation and examination of inside the rectum and sigmoid colon is called ‘sigmoidoscopy’ and it using endoscope. The reason for performing ‘sigmoidoscopy’ is to diagnose the cause of certain symptoms such as bleeding, diarrhea, pain…etc. [28, 29, 30, 31]. After each colonoscopy, endoscopy or sigmoidoscopy procedure, the ‘endoscopist’ writes detailed report about the current status of the examined part and the result of the procedure itself.

We have removed stop words from all data sets using stop-list contains the common words such as ‘are’, ‘be’, ‘do’. The Porter’s suffix-stripping algorithm is used to stem words [14]. The words considered being same words if they share the same stem. The class labels of all different data sets are generated by Doc2Mat [13]. The largest data set contains 3151 documents and the smallest data set contains 2105 documents.

Table 1. The data sets summery.

Data	Source	# of Document	# of Terms	# of Classes
Colo	Norfolk &Norwich University Hospital	2158	1494	8
Endo_1	Norfolk &Norwich University Hospital	2113	802	5
Endo_2	Norfolk &Norwich University Hospital	3151	1004	9
Endo_3	Norfolk &Norwich University Hospital	3006	1157	7
Sigmoid	Norfolk &Norwich University Hospital	2105	1883	13

6. CLUSTER QUALITY EVALUATION

There are two types of cluster quality measurement that allow comparing different set of clusters without reference to external knowledge, this type of measures are called *internal quality* measures. The second type of measures evaluates how well the clustering is working by comparing the group produced by different clustering techniques to the classes. This type of measures is called *external quality* measure. There are two external measures will be used in this paper entropy [1, 4, 7, 12] and purity [4, 7]. The best criterion function that performs better than other criterion functions using these three similarity measures, thus, we will be confident that this criterion function is the best for the situation being evaluated.

6.1. Entropy

As we mentioned earlier, entropy is one of the external similarity measures of quality of clusters. The best clustering solution will be the one that leads to clusters that contain documents from only a single class, in which case the entropy will be zero. Generally speaking, the better clustering solution is accomplished when the entropy values are small [1, 4, 7, 12]. With Entropy, we can measure how the different documents classes are distributed within each class. First, the class distribution is calculated for each cluster; then this class distribution will be used to calculate the entropy for each cluster according to the following formula:

$$E_j = -\sum_i p_{ij} \log(p_{ij})$$

Where p_{ij} is the probability that a member of cluster j belongs to class i and then the summation is taken over all classes. After the entropy is calculated the summation of entropy for each cluster is calculated using the size of each cluster as weight. In other words, the entropy of all produced clusters is calculated as the sum of the individual cluster entropies weighted according to the cluster size, and defined as:

$$E_{SC} = \sum_{j=1}^{m} \frac{n_j * E_j}{n}$$

where n_j is the size of cluster j, n is the total number of documents, and m is the number of clusters.

6.2. Purity

The *purity* measures the extent to which each cluster contained documents from primarily one class. In general, the better clustering solution is accomplished when the values of purity are large [4, 7]. In similar entropy way, the purity of each cluster is calculated as:

$$P(S_r) = \frac{1}{n_r} \max_i (n_r^i)$$

where S_r is a particular cluster of size n_r. The purity of all produced clusters is computed as a weighted sum of the individual cluster purities and is defined as:

$$Purity = \sum_{r=1}^{k} \frac{n_r}{n} P(S_r)$$

7. THE METHODOLOGY

The clustering method used in this experiment is agglomerative. We used the most common similarity measure between documents which is cosine. There are number of schemes to choose which cluster to split [8]. The suitable number of clusters is 5 clusters calculated using SAS Text Miner which is tool in SAS Enterprise Miner [15]. In this experiment, 5-, 10-, 15-, and 20 clusters were obtained for each one of the different datasets. The quality of the obtained clusters will be measured by *entropy* and *purity*.

8. RESULTS

The detailed results showed that the selection schemes *single-link* and *UPGMA* are the worst because they produced very poor clusters for all data sets. For this reason those two selection schemes are exempted from the comparison. *Single-link* assigned over 91% of the documents in one cluster and in some data sets over 99%, where as *UPGMA* is slightly better the *single-link* but also it assigned over 77% of the documents in one cluster. The values illustrated in Table 2 are the percentages of the documents that assigned to one cluster.

Table 2. The percentages of the documents assigned in 1 cluster by single-link and UPGMA.

	Single-Link				UPGMA			
	5	10	15	20	5	10	15	20
Colo	99.8	99.6	99.4	99.1	98.8	96.3	93.1	77
Endo1	94.5	94.3	94	93.8	98.5	91.6	83.7	78.7
Endo2	93.8	93.7	93.5	93.2	98.6	91.7	85.6	83
Endo3	91.7	91.5	91.3	91.2	98.4	88.8	80.2	78.5
Sigmoid	99.8	99.6	99.3	99.1	99	88.8	88.1	86.1

The *entropy* and *purity* results of the criterion functions for the 5 datasets and for 5-, 10-, 15-, and 20-way clustering solutions are shown in Tables 3 and 4 respectively, which shows both the entropy and the purity results for the entire set of experiments. To summarize these results (entropies and purities), we will calculate the average of each criterion function over the entire set of datasets. Two ways are used to calculate such average; the first way is the simple averaging, which is calculated by summing the entropies or purities of a particular criterion function for the 5 data set and then divided by five (the number of data sets), but using simple averaging is not recommended by [7] because they felt such simple averaging may distort the overall results. Whereas, the second way is the averaging relative, which is recommended by Jain *et. al* [7] and is calculated by dividing the entropy obtained by a particular criterion function for each dataset and value of *k* (5-, 10-, 15- or 20) by the smallest entropy - the best entropy- obtained for that particular dataset and value of *k* over the different criterion functions. The degree to which a particular criterion function performed worse than the best criterion function is represented by the calculated ratios, which will be referred as relative entropies. After we calculated the relative entropies, for each criterion function and value of *k* we averaged these relative entropies over the various datasets. A criterion function that has average relative entropy close to 1.0 will indicate that this function did the best for most of the datasets.

Table 3. Entropy values for the various datasets and criterion functions for the clustering solutions obtained via *k*-cluster.

	# of Clusters = 5					# of Clusters = 10				
	Colo	Endo1	Endo2	Endo3	Sigmoid	Colo	Endo1	Endo2	Endo3	Sigmoid
I1	0.444	0.516	0.372	0.432	0.423	0.320	0.457	0.350	0.375	0.372
I2	0.403	0.476	0.419	0.444	0.423	0.310	0.463	0.355	0.413	0.395
E1	0.386	0.489	0.422	0.473	0.423	0.327	0.478	0.381	0.420	0.385
G1	0.419	0.479	0.435	0.448	0.411	0.268	0.458	0.382	0.429	0.368
H1	0.441	0.487	0.388	0.445	0.434	0.331	0.467	0.372	0.376	0.390
H2	0.399	0.493	0.385	0.448	0.43	0.320	0.480	0.381	0.418	0.375
Clink	0.510	0.500	0.439	0.469	0.477	0.375	0.488	0.381	0.439	0.433
	# of Clusters = 15					# of Clusters = 20				
	Colo	Endo1	Endo2	Endo3	Sigmoid	Colo	Endo1	Endo2	Endo3	Sigmoid
I1	0.317	0.442	0.342	0.365	0.358	0.305	0.437	0.318	0.361	0.346
I2	0.304	0.450	0.339	0.378	0.366	0.298	0.418	0.331	0.366	0.357
E1	0.313	0.470	0.355	0.386	0.361	0.306	0.442	0.348	0.381	0.352
G1	0.257	0.438	0.381	0.399	0.354	0.252	0.429	0.360	0.396	0.343
H1	0.313	0.455	0.342	0.355	0.370	0.300	0.423	0.336	0.349	0.350
H2	0.314	0.474	0.357	0.382	0.346	0.310	0.460	0.344	0.360	0.343
Clink	0.343	0.477	0.378	0.430	0.388	0.334	0.473	0.356	0.424	0.374

Table 4. Purity values for the various datasets and criterion functions for the clustering solutions obtained via *k*-cluster.

	# of Clusters = 5					# of Clusters = 10				
	Colo	Endo1	Endo2	Endo3	Sigmoid	Colo	Endo1	Endo2	Endo3	Sigmoid
I1	0.719	0.519	0.618	0.626	0.525	0.819	0.614	0.634	0.681	0.569
I2	0.726	0.631	0.574	0.626	0.502	0.837	0.631	0.640	0.664	0.544
E1	0.755	0.603	0.592	0.619	0.551	0.823	0.619	0.640	0.673	0.600
G1	0.692	0.609	0.544	0.624	0.570	0.861	0.635	0.618	0.637	0.623
H1	0.692	0.559	0.625	0.629	0.519	0.825	0.599	0.625	0.676	0.567
H2	0.759	0.593	0.638	0.631	0.562	0.824	0.612	0.638	0.669	0.605
Clink	0.692	0.572	0.535	0.630	0.509	0.760	0.577	0.651	0.646	0.556
	# of Clusters = 15					# of Clusters = 20				
	Colo	Endo1	Endo2	Endo3	Sigmoid	Colo	Endo1	Endo2	Endo3	Sigmoid
I1	0.819	0.628	0.638	0.681	0.592	0.819	0.634	0.677	0.683	0.616
I2	0.837	0.631	0.655	0.686	0.605	0.837	0.651	0.668	0.691	0.626
E1	0.823	0.622	0.641	0.679	0.623	0.823	0.634	0.657	0.679	0.626
G1	0.861	0.645	0.618	0.662	0.650	0.861	0.645	0.648	0.662	0.653
H1	0.825	0.626	0.645	0.690	0.590	0.836	0.649	0.655	0.690	0.625
H2	0.824	0.622	0.641	0.680	0.644	0.824	0.629	0.657	0.691	0.644
Clink	0.769	0.604	0.653	0.646	0.604	0.769	0.605	0.662	0.649	0.609

In the same manner we calculate the average relative purity. Since the higher values of purity are better, the only difference is we divide a particular purity value with the highest purity value (the best purity), and then averaged them over the various datasets. The average relative purity will be interpreted in a similar manner as those of the average relative entropy (they are good if they are close to 1.0 and they are getting worse as they become greater than 1.0).

The values of the calculated average relative entropies and purities for the 5-, 10-, 15-, and 20-way clustering solutions are shown in Table 5. The columns labeled 'Avg' contain the simple average of these relative averages over the four sets of k-clustering solutions. The columns labeled '%W' show the difference of percentages between each criterion function and the best one, for example, the average relative entropy value of *I1* criterion function is 9% worse that the best one which is *UPGMA*. The bolded and underlined entries show the criterion functions that performed the best, and the bolded entries show the criterion functions that performed within 2% of the best. The relationship between the different criterion function and entropy is showed in Figure 1 and for Purity in Figure 2.

Table 5. Averaged relative entropies and purities over the 5 datasets for different criterion functions for the clustering solutions obtained via *k*-cluster.

	Entropy						**Purity**					
	5	**10**	**15**	**20**	**Avg.**	**%W**	**5**	**10**	**15**	**20**	**Avg.**	**%W**
I1	1.176	1.171	1.151	1.159	**1.164**	1.8	0.836	0.810	0.820	0.837	**0.826**	1.10
I2	**1.074**	1.249	1.209	1.188	1.180	3.1	0.843	**0.792**	0.816	0.830	**0.820**	0.37
E1	1.136	1.218	1.204	1.195	1.188	3.8	0.826	0.815	0.823	0.831	**0.824**	0.86
G1	**1.067**	1.422	1.423	1.413	1.331	14.1	0.878	**0.784**	**0.798**	0.806	**0.817**	0.00
H1	1.131	1.17	1.173	1.172	**1.162**	1.6	0.874	**0.798**	0.818	0.827	**0.829**	1.47
H2	1.119	1.234	1.193	1.172	1.180	3.1	0.839	0.813	0.828	0.836	**0.829**	1.47
Clink	1.091	1.129	1.176	1.174	**1.143**	0.0	0.849	0.839	0.852	0.857	0.849	3.92

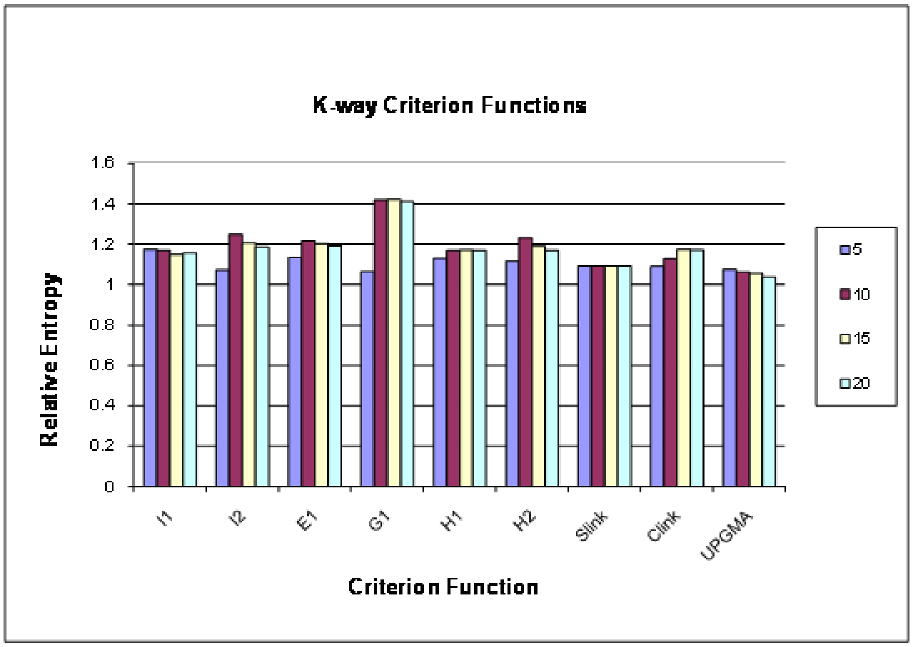

Figure 1. Entropy vs. Criterion functions for four clustering.

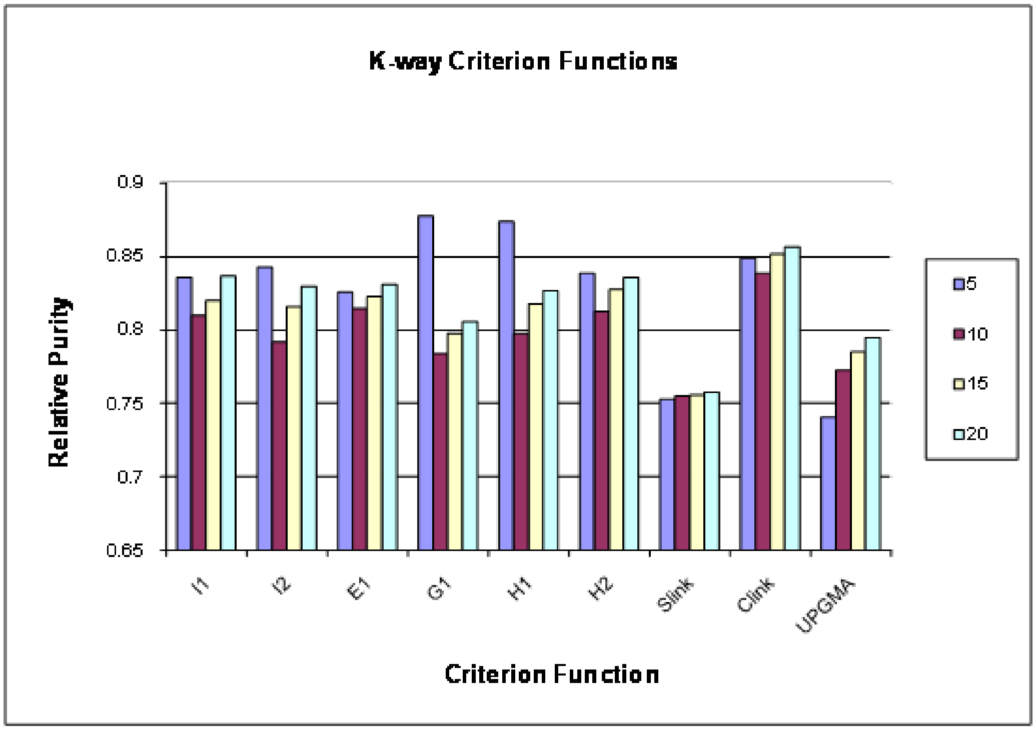

Figure 2. Purity vs. Criterion functions for four clustering.

In the same way of calculating the averaged relative of entropy, we calculate the averaged relative of the clustering time for each criterion function for the four k-clustering which is illustrated in Table 6 and Figure 3.

Table 6. Averaged relative clustering time.

	5	10	15	20	Average
I1	8.55	8.39	8.42	8.43	**8.45**
I2	8.42	8.39	8.43	8.38	**8.41**
E1	8.73	8.68	8.76	8.66	**8.71**
G1	11.86	11.81	11.91	11.80	**11.85**
H1	366.20	360.83	371.33	377.65	**369.00**
H2	271.38	281.61	275.14	278.95	**276.77**
Clink	7.80	7.78	7.84	7.80	**7.81**

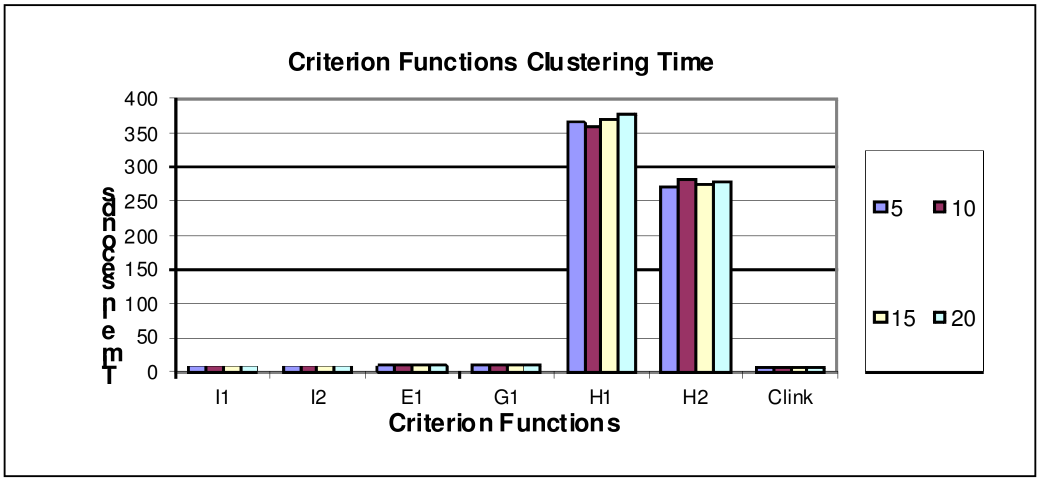

Figure 3. **:** Average relative clustering time.

9. COMPARISON AND DISCUSSION

As we mentioned earlier, *Single-Link* and *UPGMA* will be exempted from the comparison because of their poor quality clusters. There are number of observations can be made by analyzing the results shown in table 5. First, in term of entropy the best criterion function is the *Clink*. And the *G1* is 14% worse than the best criterion function. *I1* and *H1* criterion functions perform the next best within 2% of the best. *I2*, *E1* and *H2* criterion functions always perform somewhere in the middle and they are less than the average 4% worse in terms of entropy. On the other hand, in term of purity, *G1* is the best criterion function. The other criterion functions except *Clink* are the next best within 2% of the best. *Clink* is within 4% of the best so it is not so much worse. For both purity and entropy we will find that *I1* and *H1* within 2% of the best. *I1* is only 1.8% worse that the best in term of entropy and 1.1% worse than the best in term of purity. *H1* is 1.6% worse than the best in term of entropy and 1.47% worse than the best in term of purity.

10. CONCLUSION

In this paper we experimentally evaluated different agglomerative algorithms to obtain high-quality clusters for clustering medical documents. Six criterion functions and three selection schemes are included in the comparison. Our experimental results showed that *G1* is the best criterion function when the quality is evaluated using purity. In addition, *Clink* criterion function is the best solution when the quality is evaluated using entropy. If we take into account both entropy and purity for the evaluation of different criterion function we found out that *I1* is the best followed by *H1*.

REFERENCES

[1] Jonathan C. Prather, David F. Lobach, Linda K. Goodwin, Joseph W. Hales, Marvin L. Hage and W. Edward Hammond. *Medical Data Mining: Knowledge Discovery in a Clinical Data Warehouse*. American Medical Informatics Association Annual Fall Symposium (formerly SCAMC). p 101-5. 1997.

[2] Douglass R. Cutting, David R. Karger, Jan O. Pedersen, and John W. Tukey, *Scatter/Gather: A Cluster-based Approach to Browsing Large Document Collections*, SIGIR '92, Pages 318 – 329, 1992.

[3] Daphe Koller and Mehran Sahami, *Hierarchically classifying documents using very few words*, Proceedings of the 14th International Conference on Machine Learning (ML), Nashville, Tennessee, Pages 170-178, 1997.

[4] Michael Steinbach, George Karypis, Vipin Kumar. A Comparison of Document Clustering Techniques. Technical Report #00-034. Department of Computer Science and Engineering. University of Minnesota. USA.

[5] Ying Zhao, George Karypis. *Evaluation of Hierarchical Clustering Algorithms for Document Datasets*. Technical Report #02-022. Department of Computer Science. University of Minnesota.

[6] Sudipto Guha, Rajeev Rastogi, and Kyuseok Shim. *CURE: An efficient clustering algorithm for large databases*. In *Proc. of 1998 ACMSIGMOD Int. Conf. On Management of Data*, 1998.

[7] A. K. Jain, M. N. Murty, P. J. Flyn. 1999. *Data Clustering: A Review*. ACM Computing Surveys, Vol. 31, No. 3, September 1999.

[8] Ying Zhao and George Karypis. *Hierarchical Clustering Algorithms for Document Datasets*. Technical Report #03-027, Department of Computer Science, University of Minnesota, Minneapolis, MN, 2002. available at http://www.cs.umn.edu/~karypis

[9] Ying Zhao and George Karypis. Comparison of Agglomerative and Partitional Document Clustering Algorithms.

[10] Raza Ali, Usman Ghani and Aasim Saeed. *Data Clustering and Its Applications*.

[11] Ying Zhao and George Karypis. *Criterion functions for document clustering: Experiments and analysis*. Technical Report TR #01–40, Department of Computer Science, University of Minnesota, Minneapolis, MN, Feb 2002. Available on the WWW at *http://cs.umn.edu/~karypis/publications*.

[12] King-Ip Lin, Ravikumar Kondadadi. A Word-Based Soft Clustering Algorithm For Documents. Department of Mathematical Sciences, The University of Memphis, Memphis, TN 38152, USA.

[13] Un Yong Nahm and Raymond J. Mooney. Text Mining with Information Extraction. AAAI Symposium on Mining Answers from Texts and Knowledge Bases, Stanford, CA, 2002.

[14] SAS Institute Inc. *SAS Enterprise Miner*. 2004.

[15] SAS Institute Inc. *SAS Text Miner*. 2004.

[16] M. F. Porter. *An Algorithm for Suffix Stripping*. Program. 1980.

[17] George Karypis. *CLUTO: A Clustering Toolkit*. Technical Report: #02-017. University of Minnesota, Department of Computer Science. November 28, 2003.

[18] A. K. Jain and R. C. Dubes. *Algorithms for Clustering Data*. Prentice Hall, 1988.

[19] C. J. Bowles, R Leicester, C. Romaya, E. Swarbrick, C. B. Williams and O. Epstein. *A Prospective Study of Colonoscopy Practice in the UK today: are we Adequately Prepared for national colorectal Cancer Screening Tomorrow?* International Journal of Gastroenterology and Hepatology. Jun 2003.

[20] Jackson Gastroenterology, *Colonoscopy*. 2002. http://www.gicare.com/pated/ epdgs19.htm

[21] The Cancer Information Network. *What Is Colonoscopy: An Overview*. http://www. ontumor. com/ colorectal/colonoscopy/

[22] National Institute of Diabetes and Digestive and Kidney Diseases. Colonoscopy. National Institutes of Health. Bethesda, MD. http://digestive.niddk.nih.gov/ ddiseases/pubs/colonoscopy/

[23] Jackson Gastroenterology, *Upper GI Endoscopy* . 2002. http://www.gicare.com/ pated/epdgs18.htm

[24] National Institute of Diabetes and Digestive and Kidney Diseases. *Upper Endoscopy*. National Institutes of Health. Bethesda, MD. http://digestive.niddk.nih.gov/ddiseases/pubs/ upperendoscopy /index.htm

[25] U.S. National Library of Medicine. *Medical Encyclopedia: Colonoscopy*. Bethesda, MD, JAN 2005. http://www.nlm.nih.gov/medlineplus/ency/imagepages/1083.htm

[26] Jackson Gastroenterology. *Flexible Sigmoidoscopy*. 2002. http://www.gicare.com/pated/ epdgs23.htm

[27] National Institute of Diabetes and Digestive and Kidney Diseases. *Flexible Sigmoidoscopy*. National Institutes of Health. Bethesda, MD. http://digestive. niddk.nih.gov/ddiseases/pubs/upperendoscopy/index.htm

[28] U.S. National Library of Medicine. *Medical Encyclopedia: Sigmoidoscopy*. Bethesda, MD, JAN 2005. http://www.nlm.nih.gov/medlineplus/ency/article/003885.htm

[29] George Karypis. *Doc2Mat: Converting Documents into vector-space format*.

[30] Claude E. Shannon, *A Mathematical theory of communication*, Bell System Technical Journal, Vol. 27, pp. 379-423. 1948.

Authors

Dr. Fathi Hassan Saad is a Business Intelligence Architect in NHS London, UK. He obtained his first degree, a first class BSc Honours in Computer and Information Systems from Sudan University of Science and Technology in 1994. MSc in Computer Science, University Putra Malaysia in 2001. Saad completed his PhD in Data Mining at University of East Anglia in the UK in 2008. Dr. Saad worked as a lecturer, teaching BSc as well as MSc students in Sudan, Malaysia and the UK. Moreover, Dr. Saad managed several projects leading to successful implementation of BI systems for private companies as well as state governments and federal ministries.

Dr. Omer I. E. Mohamed studied Computer Science at Aalborg University-Denmark. Currently, he is working as an Assistant Professor at Al Ain University of Science and Technology, United Arab Emirates. Previously, Dr. Mohamed worked as an Assistant Professor at the College of Computer Science and Information Technology, Sudan University for Science and Technology, Lecturer at Multimedia University, Malaysia and a Research Assistant at Aalborg University, Denmark.

Dr. Rafa E. Al-Qutaish received the B.Sc. in Computer Science and M.Sc. in Software Engineering degrees in 1993 and 1998, respectively. Also, he received the Ph.D. degree in Software Engineering from the School of Higher Technology (ÉTS), University of Québec, Canada in 2007. Currently, he is a Deputy Dean and an Associate Professor of Software Engineering at Al Ain University of Science and Technology in Abu Dhabi, UAE. His current research interests are in Software Measurement, Software Product Quality, Software Engineering Standardization, Reverse Engineering, Software Comprehension and Maintenance, and Compiler Construction. So far, he has published more than 17 papers in international peer-reviewed journals, 15 papers in international peer-reviewed conferences & workshops, and 3 chapters. Furthermore, Dr. Al-Qutaish is a senior member of the IEEE & IEEE-CS, and also a senior member of the IACSIT. http://www.rafa-elayyan.ca