Efficient Approach for Building Hierarchical Cluster Representative

Mohammad A. Hassan[1], **Yaser A. M. Hasan**[2]

[1]Zarqa University, Jordan,

[2]Zarqa University, Jordan,

SUMMARY

CLUSTER RETRIEVAL IS USUALLY USED TO IMPROVE RETRIEVAL EFFICIENCY, SINCE USER NEEDS ARE COMPARED WITH A CLUSTER REPRESENTATIVE; OR CENTROID, INSTEAD OF ALL DOCUMENTS. IT IS IMPORTANT TO SELECT THE CENTROID IN A WAY IT IS STRONGLY REPRESENTS THE SEMANTICS OF THE CLUSTER MEMBERS. IN THIS PAPER WE PROPOSED A NEW METHOD TO FORM THE CENTROID IN CASE OF HIERARCHICAL CLUSTERING IS USED. SUCH APPROACH DEPENDS ON INDEX TERMS OF THE PARENT DOCUMENTS IN THE HIERARCHY, COMBINING THESE TERMS INTO A VIRTUAL DOCUMENT VECTOR OF ENTRIES THAT COMPOSED OF THE ACCUMULATED WEIGHT OF EACH INDEX TERM. THE CENTROIDS WERE EVALUATED USING TWO VARIABLES; DISTANCE TO OTHER CENTROIDS, AND CONNECTIVITY WITH THE SAME CLUSTER MEMBER, EMPIRICAL RESULTS PROVED THE EFFICIENCY OF SUCH APPROACH EVEN WHEN USING A SUBSET OF THE TOP MOST IMPORTANT TERMS THAT REPRESENT THE CENTROID; CALLED TOP-N% OF TERMS.

***KEY WORDS*:**

INFORMATION RETRIEVAL, CLUSTER RETRIEVAL, CLUSTER REPRESENTATIVE

1. Introduction

Clusters have been used for grouping documents of relevant semantics in order to enhance retrieval efficiency; either by returning documents that are relevant to users' needs or present results quickly in a similar way as browsing. Clustering proved efficient since documents that are belonging to a cluster are most likely being relevant to user requests, which was declared by Rijsbergen as the cluster hypothesis[1]. Other researchers extend this hypothesis in the opposite direction; i.e. documents that are relevant to a request are most likely belong to the same cluster [2].

In cluster retrieval, user requests are compared to a representative of a cluster instead of comparing them to all documents in a collection[1, 3, 4]. So the representative will have a crucial effect on efficiency and should be more accurate. Therefore, the representative should be selected carefully to reflect the characteristics of the cluster it represents.

In this paper we proposed a new method of selecting cluster representative for clusters that have hierarchical structure. In this method, each cluster is represented by a node (or a root document) in the zero level of the hierarchy, and the other documents are represented by child nodes in higher levels. The relationship between the root document and documents in higher levels is either direct relation (having similarity greater than the initial threshold), or *incremental transitive* relation, defined in section 2, where a document in level (L) has similarity to its parent equals the initial threshold (δ) plus some increment (ε) depends on L.

Clustering method has great influence on representative selection, a brief survey about the effect of clustering method have on representative selection and creation is presented next.

In case of partitioning clustering (as in the case with k-mains and its variants), the centroid is calculated as the statistical average of the objects vectors, dependent on the number of clusters which is predefined, and the initial location selected for these centroids (or means)[5, 6, 7].

In the case of agglomerative clustering, there are two approaches depending on the goal of clustering: static or incremental [8].

Incremental methods [7, 9] used an arbitrary document vector as a seed (or representative) of a cluster. If a new added document is not relevant to this seed, it will be considered as a new seed of a new cluster, such as in the case of STC (Suffix Text Clustering) [8, 10]. Browsing retrieval results using scatter/gather [3] also considered cluster representative as a vector of terms represent some topic as a vector of topical entries. Users can browse using topical terms, and cluster summaries, or *tags* as in [11]. Clusters created by agglomerative methods were represented by a wide range of centroid choices, such as:

- In maximally-linked document [1], the centroid is introduced as “that document which is linked to the maximum number of other documents in the cluster”. The centroid is not unique, and the centroid could be a vector of topics in some context or prototypical document [12, 13].
- In center of gravity approached [1], centroid is obtained by dividing the sum of all normalized vectors in a cluster over the number of vectors (documents) in that cluster. Or the average of the weights in all documents vectors [7], where top-n ranked documents in a cluster could be used as a representative [2, 14].

Manuscript received January 5, 2011
Manuscript revised January 20, 2011

- It also could be the maximal predictor of the cluster [1], where a term is considered as a member of the representative if it occurs in at least half of the documents belonging to the cluster. The ratio selected as a threshold will affect the widening or shortening of context represented by such a cluster[12]. Therefore, it is recommended not to choose terms that have very low or very high document frequency for context definition [13]. It is known from information theory that information content "entropy" related to the logarithm of the document frequency. So, high document frequency terms have higher entropy values [12], therefore, it is important to make some kind of tradeoff when using document frequency as a criterion for term selection into a cluster centroid. In this paper we included both frequent terms, and high weighted terms.

We can conclude the following general principles regarding selecting cluster centroid:

- Generally, terms, topics, or categories that used to represent both documents and centroids of clusters are selected from the same set of documents.
- The way in which clusters are created affects the centroid selection, which means that the centroid should be consistent with the clustering method.
- Cluster centroid should support the objective of clustering. So, if the objective is browsing then the centroid is built as a hierarchy of topical terms or summaries.
- The centroid is selected to distinguish one cluster from others. So, the distance between the centroid and other objects in the cluster should be minimized, while the distance to other clusters should be maximized.
- The centroid should be selected in a way that minimizes the effect of similarity measure, and/or the indexing method.
- In a peer-to-peer environment, clusters belonging to a node should reflect the topics it includes. So, it is more useful to include high frequency terms, as well as highly weighted terms in the cluster representative. Similar condition presented in [13] for text categorization.

Throughout the paper, cluster representative may referred as: representative vector, centroid, or virtual document.

The rest of the paper is organized as follows; section 2 introduces the clustering method used to build cluster, section 3 presents the proposed method for selecting a cluster centroid, evaluation of the selected centroid is presented in section 4, and section 5 concludes the paper.

2. Incremental Transitive Clustering

Definition: Incremental transitive relevance:

Given a collection of normalized vectors of documents (D), and a relevance relation ($\Re$) defined on D, such that $\Re=\{(x,y)$: sim(x, y) ≥ δ, $\forall$ documents x, y $\in$ D}, where (δ) is the relevance threshold, and sim(x,y) is the similarity measure. $\Re$ is the incremental transitive relevance relation if for any d1, and d3$\in$D, and sim(d1,d3)<δ, then d1 and d3 have incremental transitive relevance if there exists d2$\in$ D such that sim(d1, d2) ≥ δ, and sim(d2, d3) ≥δ+ε, where ε is a positive real number,.

Incremental transitivity proposes increments on threshold value to get a sequence:

$$\delta i=\delta°+i.\varepsilon \qquad (1)$$

as converting to higher levels of transitivity, where δ° is the initial relevance threshold.

By expanding this definition to d1, d2, … ,dn , of n-1 incremental transitive relevance levels, then any document vector di $\in$ D, i = 1 ,…, n , that has a direct relevance with di-1 can has transitive relevance with d1 if sim(d1, di-1) ≥δ°+(i-1)×ε. Documents in higher levels do not have direct similarity to the root document but it has incremental transitive relevance through its parent node(s), hence the hierarchy is called incremental transitive hierarchy.

3. Building the Centroid

The proposed centroid is consistent with the incremental transitivity clustering method, it should reflect the contents of the transitive hierarchical structure, as in Fig.1. Each parent node (has a children) which can found in any level in the hierarchy represents a document all of its children are relevant documents with similarity greater than or equal to the threshold plus the product of the increment and the child level. However those children don not have a direct relevance to the zero-level node. So the cluster representative should include those terms of the node in the zero-level that represent the topmost topics, in addition to the terms that belong to parents in higher levels. Such terms do not belong to the top most parents in the zero level of the hierarchy.

Formally, let C_i be a cluster whose root document is d_i, and d_i is represented by the vector: $d_i=(w_{\Delta(1)}, w_{\Delta(2)}, \ldots , w_{\Delta(n)})$, where $\Delta(j), j=1, .. , n$ is the index of the jth none zero entry in the vector d_i, and $w_{\Delta(j)}$ is the weight of the index term $t_{\Delta(j)}$.

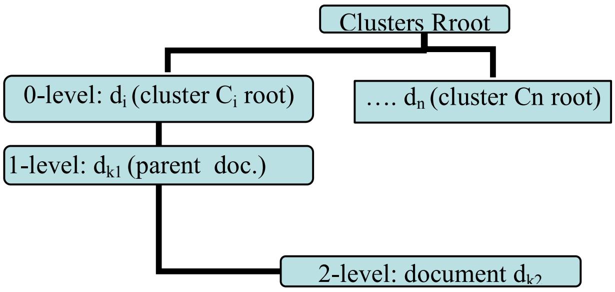

Clusters Rroot
0-level: d_i (cluster C_i root)
…. d_n (cluster Cn root)
1-level: d_{k1} (parent doc.)
2-level: document d_{k2}

Fig.1. A cluster hierarchy

Let d_k be a child node of d_i in the cluster hierarchy, then d_k itself is a parent to other set of relevant children that are specified to have relevance to d_i through incremental transitivity. So, there must be some terms in d_k that do not exist in d_i, i.e. $\exists\, t_{\Delta(\upsilon)}$ entries in d_k, that do not exist in d_i. If $\Delta(\upsilon)$ is a set of indexes of terms that are elements of d_k, then the set of terms that form the centroid of the cluster C_i is the set union of all sets of terms that have non zero entries in the parent document nodes in all other levels. In this case, if there are k parent documents in the cluster C_i, with $\Delta(\upsilon_1)$, $\Delta(\upsilon_2)$, ..., $\Delta(\upsilon_k)$ indexes, then the set of terms that used to create the centroid has the following form:

$$\pi(C_i) \subseteq \{t_{\Delta(1)}, \ldots, t_{\Delta(n)}\} \cup \{t_{\Delta(\upsilon 1)}\} \cup \{t_{\Delta(\upsilon 2)}\} \cup \ldots \cup \{t_{\Delta(\upsilon k)}\} \quad (2)$$

Centroid vector is formed from the *accumulated* weights of terms belonging to $\pi(C_i)$, where the accumulated weight for a term t_k is given by the equation:

$$w(t_k, C_d) = \sum_{j=1}^{n} w(t_k, d_j) \quad (3)$$

where $w(t_k,d_j)$ is the weight given to the term t_k in a document d_j, and C_d is the centroid of the cluster whose root is the document d, and n is the number of documents in the C_d that contain the term t_i.

All terms could be included, otherwise the centroid space could be reduced to a subset of the original terms. In this paper all (100%), half (50%), and one third (33%) of terms of the representative are tested. Term vectors of parent documents in a cluster hierarchy were combined with each other, then adding weights of terms whenever a term was repeated in more documents in order to give higher accumulated weights to more frequent terms.

4. Evaluating the Centroid

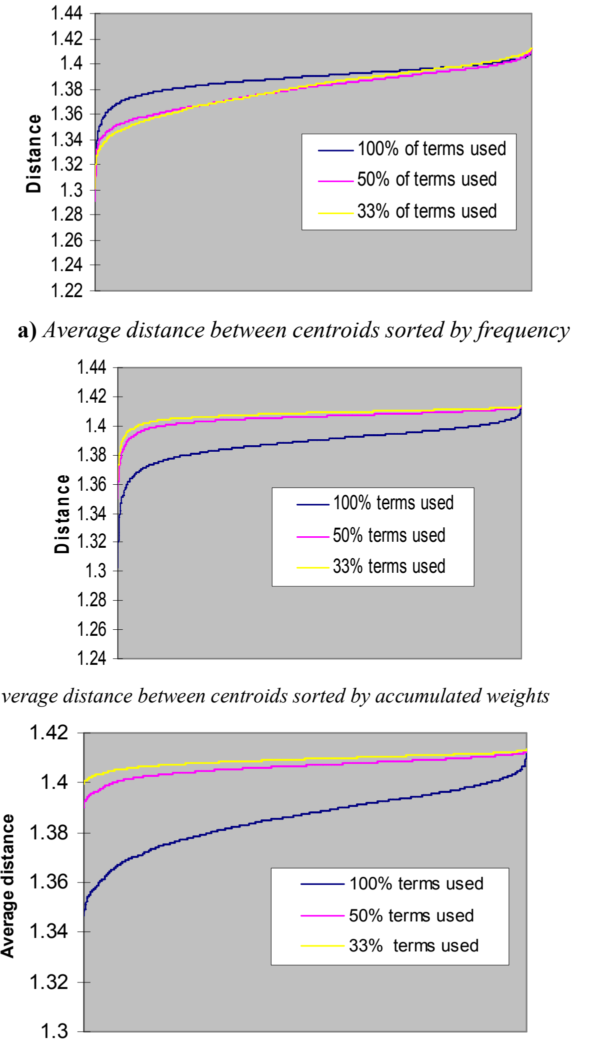
Centroids are being formed as *hypothetical*, *virtual* or *prototypical* documents [7, 10, 12, 13]. In this study clusters centroids are created as virtual documents of terms obtained form parent documents. These terms are descended sorted according to the following three criteria:

- The first criterion will consider the accumulated weight of a term while sorting, descending the overall terms in the centroid vector, where higher accumulated weights first.
- The second criterion will sort terms of a centroid according to their average weights.
- The third criterion will descend sorted centroid terms according to the frequency of that term in all parent documents of the centroid, where higher frequency terms first.

After sorting terms of a representative in descending order, to p-n% percentage of terms are selected. To evaluate the centroid efficiency, two variables need to be examined:

- The average distance among all cluster centroids, or the *dissimilarity* among clusters centroids [1], and
- The average similarity between the centroid virtual documents and all documents belonging to the same cluster, or the *internal connectivity*.

The first variable is measuring the diversity among clusters, or the efficiency of the clustering method and the adopted representative selection method, in partitioning the set of documents in a collection. The second variable is measuring the commonalty the internal *connectivity* among the centroid and the documents of a cluster. Consequently, clusters (and their centroids) are better formed when the distance between them is maximized, and the internal connectivity among their documents is maximized.



a) *Average distance between centroids sorted by frequency*

b) *Average distance between centroids sorted by accumulated weights*

c) *Average distance between centroids sorted by average weights.*

Fig. 2 Average distance between centroids.

Clusters (used in this experiment) were formed using the incremental transitive clustering algorithm, that applied on 18650 documents derived from Reuters21578 collection; all documents are selected to have a title and body text.

Average distance: using top-n% terms: After sorting terms using the above criteria, the experiments was repeated for three selections of top-terms; the first involves all terms (100%) for calculating the distance between centroids, the second involves the top 50% terms, and the third uses the top 33% terms. Note that, a percentage of terms are used instead of a fixed number of terms, since clusters' centroid vectors are formed upon different numbers of terms (different sizes). Each vector is normalized, then the distance is calculated using the Cartesian distance formula between two points in R^n:

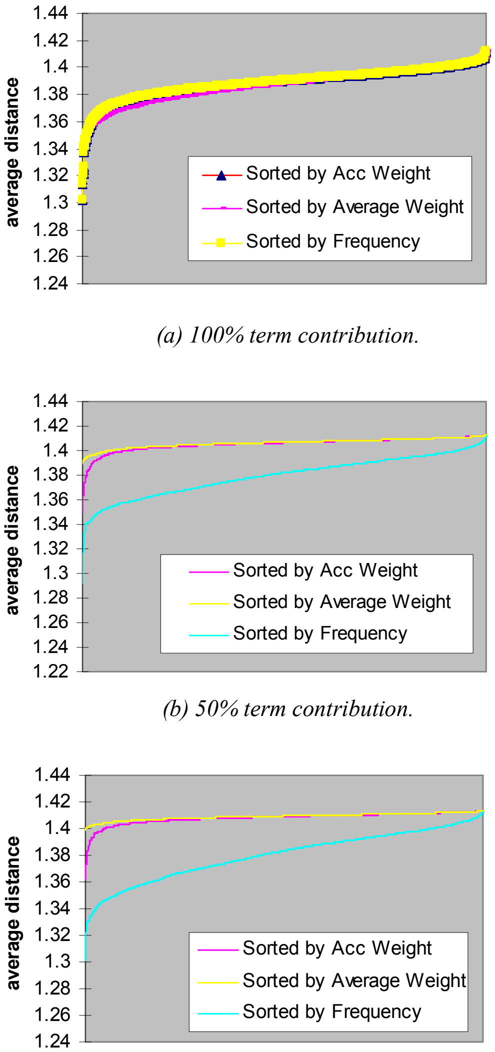
$$d=\sqrt{(y_1-x_1)^2+(y_2-x_2)^2+...+(y_n-x_n)^2} \qquad (4)$$

For each cluster centroid, the distances to all others are calculated, then the average for each is calculated. In order to compare the results for the above two sorting criteria, and for each top-n percentage, average values are ascended sorted and plotted, as in Fig.2.

When frequency is used as the sorting criterion, the distance is larger for 100% term contribution, and smaller for both 50%, and 33%; both of them have very similar values. On the other hand, when weight is used as sorting criterion, the distance became larger with the smaller number of terms. This result can be explained as follows: most frequent terms (common terms) are occurring in larger number of documents (have larger document frequency).So, if they involved in the top-n percentage vector, then more common terms are involved; i.e. X_i, and Y_i, in the distance equation are non-zeroes, and consequantly $(X_i - Y_i)^2<X_i^2$, and $(X_i - Y_i)^2<Y_i^2$, which results a smaller overall distance values. On other hand, the distance will be greater with the highest n-percentage for less frequent document terms, since in this case the probability of X_i=0, or Y_i=0 will be higher, see Fig. 2(a).

When using both accumulated weight and average weight as sorting criteria, the distance between centroids become larger when a smaller percentage of terms are used. Since terms that have high weights in a centroid are more specific to the topic of that cluster than other clusters. So, these terms are less probable to exist in other clusters, or at least will have very small weights in other cluster centroids. This result is relevant to the characteristics of specificity found by *G. Kim* in [14], where a "specific term indicates that the term is involved in a topical relationship with a document, another term, or a set of documents". Such result does not contradict *Hideo Joho and Mark Sanderson's* belief in[15], because they proved that document frequency is more accurate in determining term specificity "when the terms are very specific". Using average weight as sorting criterion will make the average distance closer, when all the terms are used; see Fig. 2 (c).

So we may conclude that, accumulated weight criterion is better in specifying the relationship between terms and topics of clusters, this is, similar to *Sebastiani's* judgment "terms that have very low or high document frequency are not being informative" [13]. It is also similar to Roony's conclusion "context defined by terms that have high document frequency will be wider " [12]. Fig.1 presents a comparison between the three criteria, and for the three term contribution percentages, note that the similarity between both accumulated and average weight sorting criteria is closer in the three situations.



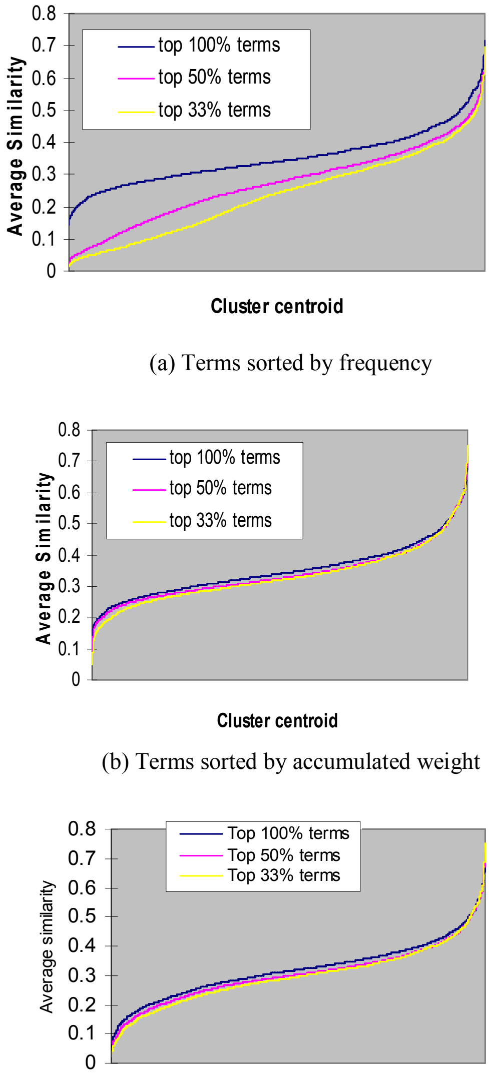
(a) 100% term contribution.

(b) 50% term contribution.

(c) 33% term contribution.

Fig.1: Average distance between cluster centroids, compared for three sorting criteria.

The similarity between documents and clusters' centroids (internal connectivity): The same experiment was repeated for the second variable; i.e. the internal similarity (connectivity) between the centroid and the documents of a cluster, using the same three sorting criteria. The effect of both term selection and space reduction (the contribution of only the top-n% of terms) on the internal connectivity is examined. The examination was done by computing the cosine similarity between the virtual document vector of a cluster centroid and all documents in the cluster, and then the average cosine similarity is calculated, sorted ascending and plotted as in Fig.*2*.



(a) Terms sorted by frequency

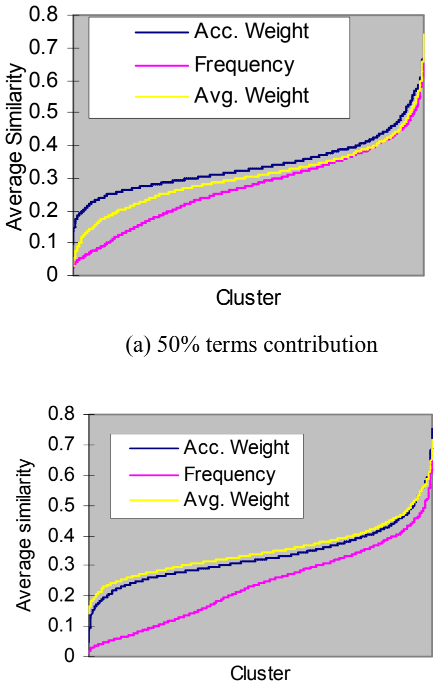
(b) Terms sorted by accumulated weight

(c) Terms sorted by average weight

Fig.2: Average similarity between cluster centroid and its documents

When term frequency is used as the sorting criterion, the average similarity gets smaller as the percentage of contributed terms reduced. So, space reduction has a major negative effect on internal connectivity, see Fig.*2*(a). However, when using either accumulated weight or average weight as the sorting criterion, space reduction has a minor effect on the internal connectivity among cluster members, as shown in Fig.*22*(b, c).

These results can be explained as follows: When accumulated weight is used to sort terms, frequent terms with small original weights (given to a term while document indexing) will gain more accumulated weight since the suggested method adds weights when a term is repeated in more than one document. So, both mostly frequent and original high weighted terms are positioned at the top of the list, and consequently will be selected when reducing the centroid space, which results minor changes to the average similarity. On other hand, when using the frequency as a sorting criterion, the less frequent but originally high weighted terms will not be selected in the top most terms of the sorted term list. Thus, when reducing centroid to a percent of its original space these terms will not included; i.e. average similarity will get smaller in case of lower percentage term involvement.



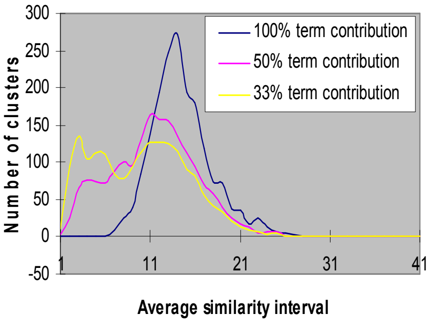
(a) 50% terms contribution

(b) 33% terms contribution

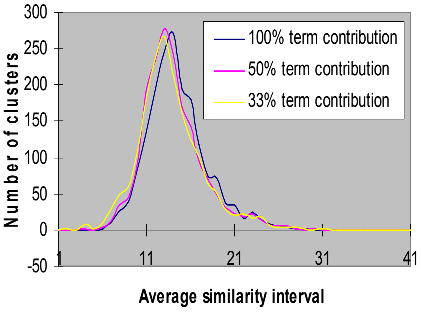
Fig.3: Comparing average similarity for three sorting criteria.

Fig.3 presents a comparison between the three sorting criteria's effects on similarity regarding the two term percentages 50%, and 33%. The 100% case is a trivial one and makes no difference, since all terms will remain regardless the sorting criteria. Using the top-most percentage of terms to form the centroid (using frequency as a sorting criterion) produced weaker connectivity among cluster members and the centroid. Whereas, in the case of using either accumulated weight or average weight the shift is smaller. In the mean wile, when the percentage is getting smaller; i.e. in case of 33% for example, using the average weight will produced better internal connectivity, hat is still closer to the case when using accumulated weight as sorting criterion.

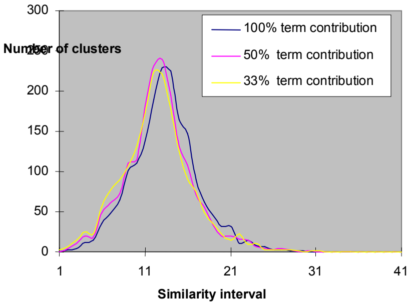
Fig. 4 presents similarity distributions for the three sorting criteria. Fig. 4(a) shows that the average similarity distribution is skewed toward the left; i.e. biased to smaller values when using frequency as a sorting criterion. While in the case of using accumulated weight, the distribution is slightly shifted to the left with a minor change of the distribution, as shown in Fig. 4(b), and also when using average weight as in Fig. 4(c). It could be noted from the distribution that the accumulated weight distribution has a better concentration around the mean (comparing to the average weight). More specifically, in the case of accumulated weight there are more than 250 clusters having internal similarity values closer to the mean, whereas with the average weight case, it is less than 250.



(a) When frequency is used.



(b) When accumulated weight is used.



(c) When average weight is used.

5. CONCLUSION

While creating the centroid, it is better to select terms from parent documents of a cluster hierarchy, since they are relevant to more documents in the cluster with higher similarity values. Accumulated weight of terms is the preferred criterion while choosing the top-n% terms; i.e. when reducing the space of centroid virtual document. When using accumulated weight as a criterion to select contributing terms, both weights and document frequencies are involved in the process, which considered more efficient while selecting the terms of the centroid.

REFERENCES

[1] C. J. V. Rijsbergen, Information Retrieval, second ed., 1979.

[2] S.-H. Na, I.-S. Kang, J.-E. Roh, and J.-H. Lee, "An empirical study of query expansion and cluster-based retrieval in language modeling approach," Information Processing and Management, vol. 43, pp. 302-314, 2007.

[3] M. A. Hearst and J. O. Pedersen, "Reexamining the Cluster Hypothesis: Scatter/Gather on Retrieval Results," in the Nineteenth Annual International ACM SIGIR Conference, Zurich, June 1996.

[4] A. Gionis, P. Indyk, and R. Motwani, "Similarity Search in High Dimensions via Hashing," The Very Large Data Bases Journal, pp. 518-529, 1999.

[5] J. B. MacQueen, "Some Methods for classification and Analysis of Multivariate Observations," in 5-th Berkeley Symposium on Mathematical Statistics and Probability, Berkeley, 1967, pp. 281-297.

[6] D. J. C. MacKay, Information Theory, Inference, and Learning Algorithms, Fourth printing ed.: Cambridge University Press, 2005.

[7] J. Allan, A. Leouski, and R. Swan, "Interactive cluster visualization for information retrieval," Department of Computer Science, University of Massachusetts, Amherst, Technical Report IR-116, CIIR, 1996.

[8] E. Greengrass, "Information Retrieval a Survey," 2000.

[9] M. Charikar, C. Chekuri, T. Feder, and R. Motwani, "Incremental Clustering And Dynamic Information Retrieval," Society for Industrial and Applied Mathematics, SIAM J. COMPUT. , vol. 33, pp. 1417-1440, 2004.

[10] O. Zamir and O. Etzoni, "Web Document Clustering: A Feasibility demonstration," in ACM/SIGIR, Melbourne, Australia, 1998.

[11] J. Caverlee, L. Liu, and D. Buttler, "Probe, Cluster, and Discover: Focused Extraction of QA-Pagelets from the Deep Web," in the 20th International Conference on Data Engineering (ICDE'04), 2004.

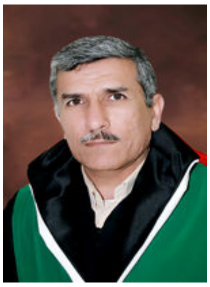
[12] N. Rooney, D. Patterson, M. Galushka, and V. Dobrynin, "A scaleable document clustering approach for large document corpora," Information Processing and Management, vol. 42, pp. 1163-1175, 2006.

[13] F. Sebastiani, "Machine Learning in Automated Text Categorization," ACM Computing Surveys, vol. 34, pp. 1-47, March 2002 2002.

[14] G. Kim, "Relationship between index term specificity and relevance judgment," Information Processing and Management, vol. 42, pp. 1218-1229, 2006.

[15] H. Joho and M. Sanderson, "Document frequency and term specificity," in Recherche d'Information Assistée par Ordinateur Conference (RIAO), 2007.

Mohammad Hassan received the B.S from Yarmouk Univ. –Jordan in 1987, the M.S. degree from Unive. Of Jordan in 1996, and the PhD in Computer Information Systems from Bradford Univ. (UK) in 2003.He working as an assistant professor in the department of computer science , Zarqa Univ. (Jordan). His research interest includes information retrieval systems, database systems, and data mining.



Yaser A. Al-Lahham received the B.S degree from Univ. of Jordan in 1985, the M.S. degree from Arab Academy (Jordan) in 2004, and the PhD in Computer Information Systems from Bradford Univ. (UK) in 2009. He working as an assistant professor in the department of computer science, Zarqa Univ. (Jordan). His research interest includes information retrieval systems, database systems, and data mining.