

Local Cluster Analysis as a Basis for High-Precision Information Retrieval

Amir Hossein Jadidinejad
Computer Engineering Department,
Islamic Azad University,
Qazvin, Iran
jadidi@basu.ac.ir

Hadi Amiri
Database Research Group
School of ECE, University of Tehran,
Tehran, Iran
h.amiri@ece.ut.ac.ir

Abstract

This paper presents a simple architecture based on Local Cluster Analysis for information retrieval systems with the intention of: (1) Improve the effectiveness of information retrieval by clustering search results and (2) Create high-precision retrieval. In this context we examine Principal Direction Divisive Partitioning algorithm and experimentally show that a clustering and re-ranking of the retrieved documents can be significantly more effective than traditional ranked list approach. Experimental results on a standard Persian test collection which is created based on TREC specifications, shows that this method is better than the best known Persian retrieval systems.

1. Introduction

Users of retrieval systems are often forced to spending a lot of time to sift through a diversity of the results and make it hard for the users to find the information they are looking for. Although information retrieval research has always been concerned with improving the effectiveness of retrieval, in some applications, such as Web search engines, a more specific requirement exists for *high-precision retrieval* [22, 6, 14]. This means that achieving high precision in the top document ranks is paramount.

In this paper we present work aimed at achieving high-precision in ad-hoc document retrieval by make clustering on retrieved results. Our experiments on *Hamshahri*, a standard Persian text collection which is created based on TREC specifications, shows that this method is almost 79 percent better than the best known Persian retrieval systems [1, 2, 18] (See also Table 2 and Fig. 3).

The rest of the paper is organized as follows. Section 2 presents the related works in close domains. Section 3 explains proposed system architecture. Section 4 gives our experiments and evaluation results. Conclusion is given in Section 5.

1.1. Overview of data clustering

The data clustering, as a class of data mining techniques, is to partition a given data set into separate clusters, with each cluster composed of the data objects with similar characteristics. Most existing clustering methods can be broadly classified into two categories: *partitioning methods* and *hierarchical methods*. Partitioning algorithms, such as *k-means*, *k-medoid* and *EM*, attempt to partition a data set into *k* clusters such that a previously given evaluation function can be optimized. The basic idea of hierarchical clustering methods is to first construct a hierarchy by decomposing the given data set, and then use agglomerative or divisive operations to form clusters. In general, an agglomeration-based hierarchical method starts with a disjoint set of clusters, placing each data object into an individual cluster, and then merges pairs of clusters until the number of clusters is reduced to a given number *k*. On the other hand, the division-based hierarchical method treats the whole data set as one cluster at the beginning, and divides it iteratively until the number of clusters is increased to *k*. See [11] for more information.

Although [17, 20, 23, 31, 33] have developed some special algorithms for clustering search results but now we prefer to use traditional methods in this paper. We will show that our method with basic clustering algorithms such as *k-means* and *Principal Direction Divisive Partitioning* achieves significant improvement over the methods based on similarity search ranking alone.

1.2 Motivation

As we mentioned before, finding relevant information from the mixed results is a time consuming task. In this context we introduce a simple high-precision information retrieval system by clustering and re-ranking retrieval results with the intention of eliminate these shortcomings.

The proposed architecture has some key features:

- *Simple and high performance.* Our experimental results (Section 4) shows that it's almost 79 percent better than the best known standard Persian retrieval systems [1, 2, 18].
- *Independent of initial system architecture.* It can embed in any fabric information retrieval system. It cause proposed architecture very good envisage for the web search engines.
- *High-Precision.* Relevant documents exhibit at top of the result list.

2. Related works

Using some kind of documents clustering technique to help retrieval results is not new, although we believe we are the first to explicitly present and deal with the low-precision problem in terms of clustering search results.

Many research efforts such as [27, 9] have been made on how to solve the keyword barrier which exists because there is no perfect correlation between matching words and intended meaning. [9] presents *TermRank*, a variation of the *PageRank* algorithm based on a relational graph representation of the content of web document collections. Search result clustering has successfully served this purpose in both commercial and scientific systems [30, 10, 23, 16, 25, 33]. The proposed methods focus on separating search results into meaningful groups and user can browse and view of retrieval results. One of the first approaches to search results clustering called *Suffix Tree Clustering* would group documents according to the common phrases [30, 31]. STC has two key features: the use of phrases and a simple cluster definition. This is very important when attempting to describe the contents of a cluster. [12] proposes a new approach for web search result clustering to improve the performance of approaches that uses the previous STC algorithms. Search Results Clustering has a few interesting characteristics and one of them is the fact that it is based only on document snippets. Certainly Document snippets returned by search engines are

usually very short and noisy. Another shortage with these systems is the cluster's name. Cluster's name must accurately and concisely describe the contents of the cluster, so that the user can quickly decide if the cluster is interesting or not. This aspect of these systems is difficult and sometimes neglected [33, 23]. In this context our tendency to provide very simple high-precision system based on cluster hypothesis [26] without any user feedback.

Document clustering can be performed, in advance, on the collection as a whole (static clustering) [7, 15], but post-retrieval document clustering (dynamic clustering) has been shown produce superior results [10, 24]. Tombros et al. [24] conducted a number of experiments using five document collections and four hierarchic clustering methods to show that if hierarchic clustering is applied to search results (query-specific clustering), then it has the potential to increase the retrieval effectiveness compared both to that of static clustering and of conventional inverted file search. The actual effectiveness of hierarchic clustering can be gauged by Cluster-based retrieval strategies perform a ranking of clusters instead of individual documents in response to each query [13]. The generation of precision-recall graphs is thus not possible in such systems, and in order to derive an evaluation function for clustering systems some effectiveness function was proposed by [13]. In this paper, Firstly we want to propose a simple architecture which use local cluster analysis to improve the effectiveness of retrieval and yet utilize traditional precision-recall evaluation. Secondly, this paper is devoted to high-precision retrieval. Thirdly, we use a larger Persian standard test collection which is created based on TREC specifications that validate [24] findings in a wider context.

Query expansion is another approach to improve the effectiveness of information retrieval. These techniques can be categorized as either global or local. While global techniques rely on analysis of a whole collection to discover word relationships, local techniques emphasize analysis of the top-ranked documents retrieved for a query [28]. While local techniques have shown to be more effective that global techniques in general [29, 2]. In this paper we don't want to expand a query based on the information in the set of top-ranked documents retrieved for the query, instead use very simple and more efficient re-ranking approach to improve the effectiveness of search result and make high-precision system that contain more relevant documents at top of the result list to help user that find information needs efficiently.

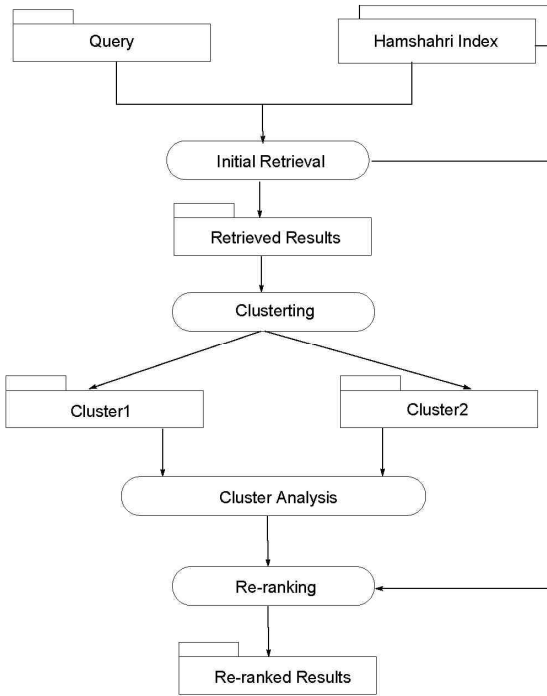


Figure 1: System architecture.

Lee et al. [15] published a “*re-ranking model based on document clusters*”. Their goal is the same as our first motivation (improve the effectiveness of retrieval) and they also use a hierarchical clustering method to identify and classify results. There is significant difference between our approaches. Lee et al. [15] apply static hierarchical agglomerative clustering to the set of whole documents and view clusters dynamically depending on retrieval results in the initial ranking. On the other hand they use static clustering and dynamic view. This approach have two disadvantages: First, since the data sets are mostly dynamic, static, pre-computed clusters would have to be constantly updated, and most clustering algorithms cannot do so incrementally. This would require a huge amount of resources and it has been shown that such an approach results in clusters of lower quality [24]; second, after the costly partitioning step, the results of cluster partitioning can contain documents which are not in the result of the first step. These have a negative effect on cluster centroid for a query. They need to adjust the value of cluster centroid to minimize the negative effects. Moreover most experiments in [15] about evaluating basic retrieval methods such as inverted file and different weighting schemes but our focus is primarily to better cluster analysis and more precision without any heavy solution.

3. System architecture

Retrieval systems generally look at each document as a unique in assigning a page rank. If the document is viewed as a combination of other related documents in the query area, we can have better results. The conjecture that relevant documents tend to cluster was made by [26].

Irrelevant documents share many terms with relevant documents but about two completely different topics, so these may demonstrate some patterns. On the other hand an irrelevant cluster can be viewed as the retrieval result for a different query that share many terms with the original query.

Xu et al. [27, 29] believe that document clustering can make mistake and when this happens, it adds more noise to the query expansion process. But as we discuss in section 4, document clustering is a good tool for high-precision information retrieval systems.

In this context we proposed architecture (Fig. 1) to cluster search results and re-rank them based on cluster analysis. Although our benchmark in the Persian language but we believe that same results must be exhibit in other benchmarks.

3.1. The initial retrieval

At the initial retrieval step, we retrieve documents based on the query-document similarity. We focus on each document at this retrieval step. The initial retrieval step ranks the retrieved documents in decreasing order of query-document similarities. [24] suggest that there is not a statistically significant variation in query-specific cluster effectiveness for different values of top-ranked documents so we’ll use top-100 documents for each query.

The Persian language is one of the dominant languages in Middle-East, so there are significant amount of Persian documents available on the Web. Some experimental results [1, 2, 18, 3, 19] show that 4-gram and term based vector space model with Lnu.ltu weighting scheme has acceptable performance for Persian text retrieval so far. However some previous research such as [2] uses some additional query that were not created according to TREC specifications so we did not include them in this paper and all of the results is based on TREC specification.

We suppose this methods in the initial retrieval step and we will show that the proposed method achieves significant improvement over all of the best methods hitherto [1, 2, 18].

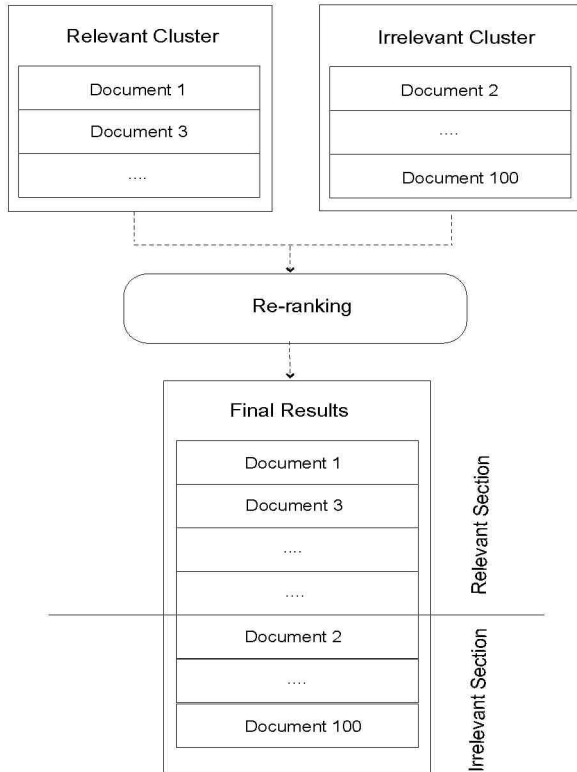


Figure 2: Search result re-ranking architecture. Document 2 does not exist in relevant cluster so sit at the down mid of output list.

3.2 Construction of clusters

Existing approaches to document clustering are generally based on either probabilistic methods, or distance and similarity measures. Although there are many well defined distance measures in information retrieval and specially for clustering in high dimensional situations but whereas initial retrieval is depend on similarity measures, we prefer *non-similarity-based methods* such as PDDP for clustering search results.

Principal Direction Divisive Partitioning proposed by [4] capable of partitioning a set of documents based on an embedding in a high dimensional Euclidean space. The basic idea is to recursively split the data set into sub-clusters based on principal direction vectors. PDDP has many key features such as unsupervised, deterministic, good scalability, high quality and identifies the distinct features of the individual clusters. Furthermore, the splits are not based on any distance or similarity measure [4, 21] thus it seems suitable for our approach. [32] created a flexible

implementation of this method. We use it to cluster search results.

Table 1: Attributes of Hamshahri collection

Attributes	Value
Collection size	564 MB
Collection Length	63,513,827 Terms
Documents Format	Text
No. Of documents	166,774
No. Of unique terms	417,339
Average length of documents	380 Terms
No. Of categories	82
No. Of Topics	65

Boley et al. [21, 4] believe that using K-means with PDDP clusters as initial configuration can be achieve higher quality (Hybrid approach), so we examine it to clustering search results but we don't get any improvement (See Hybrid column in Table 2). It seems famous k-means algorithm calculate mean values which do not differ significantly from each other. [21] presented a comparative analysis on the bisecting K-means and PDDP clustering algorithms.

As we mentioned at Section 1.2, it is not the main aim of this paper to compare the effectiveness of clustering methods. The main reason behind the choice of this method is the fact that we guessed PDDP have suitable features for our purpose, besides it has been extensively used and examined in the context of document clustering [4, 21, 5, 34].

3.3 Cluster analysis

After the clustering step, we have two groups of documents (See Fig. 1). In the cluster analysis step we have to analyze clusters content and choose relevant and irrelevant cluster.

An irrelevant cluster can be viewed as the retrieval result for a different query that shares many terms with the original query. On the other hand for a cluster of irrelevant documents, there are some query terms that do not occur in any of the documents in that cluster [29].

We conjecture that the context of a document can be considered in the retrieved results by the combination of information search and cluster analysis.

Whereas in this context we want to propose an architecture and show that it can improve the search results so we re-ranked results based on both clusters and after that choose better one manually.

Each cluster has a cluster centroid in the form of a vector which is useful as a representative of a cluster.

Table 2: Interpolated Recall-Precision with the best initial retrieval method

Recall	Precision						
	Total avg on 65 queries results (PDDP, Hybrid)			Best query results (PDDP)		Worst query results (PDDP)	
	Initial	Re-ranked (PDDP)	Re-ranked(Hybrid)	Initial	Re-ranked	Initial	Re-ranked
0.0	0.5037	0.9075	0.9202	0.7500	1.0000	1.0000	1.0000
0.1	0.4103	0.8204	0.8063	0.7500	1.0000	0.7778	0.8333
0.2	0.3817	0.7299	0.7344	0.5833	0.9000	0.7778	0.8333
0.3	0.3653	0.6500	0.6415	0.5625	0.9000	0.7778	0.8333
0.4	0.3489	0.5805	0.5814	0.3871	0.8000	0.7778	0.8333
0.5	0.3314	0.5140	0.5258	0.3684	0.7500	0.7778	0.3529
0.6	0.3139	0.4600	0.4531	0.2769	0.6800	0.7778	0.1594
0.7	0.2886	0.3918	0.3858	0.2439	0.5263	0.7273	0.1594
0.8	0.2533	0.3227	0.3206	0.0000	0.0000	0.5000	0.1594
0.9	0.1612	0.1990	0.1985	0.0000	0.0000	0.4762	0.1594
1.0	0.0342	0.0522	0.0484	0.0000	0.0000	0.3929	0.1594
11pt avg	0.2766	0.4960	0.4947	0.3384	0.6034	0.6611	0.4629

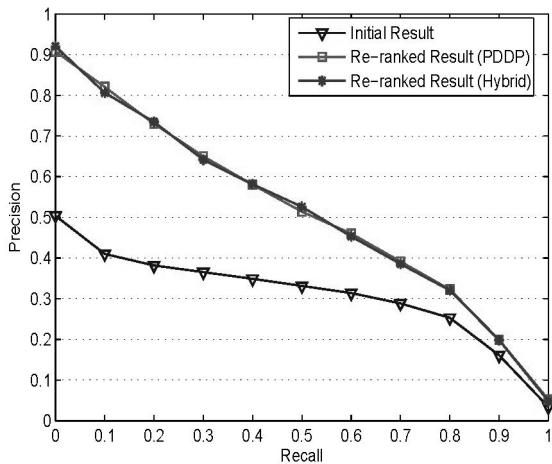


Figure 3: Average Interpolated Recall-Precision over all 65 queries. Average R-P for Initial, PDDP and Hybrid results is 0.2766, 0.4960 and 0.4947. See Table 2 for more detail.

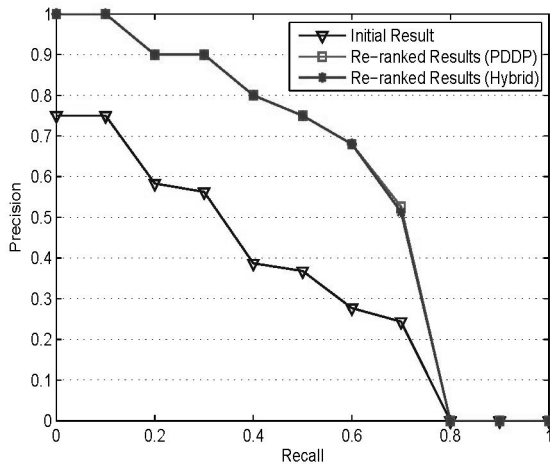


Figure 4: Interpolated Recall-Precision at the best query. Average R-P for Initial, PDDP and Hybrid results is 0.3384, 0.6034 and 0.6010. See Table 2 for more detail.

We conjecture that relevant cluster centroid must be near than irrelevant cluster centroid to the query. So we can choose which cluster centroid that toward to the query (relevant cluster).

3.4. Documents re-ranking

The architecture of document re-ranking model is shown in Fig. 2. This model is combining the initial retrieved documents and the cluster analysis results.

We focus on *initial retrieved documents* and combine it with *clusters evidence* (See Fig. 2). Re-ranked list consist of two sections. *Relevant section* contains documents in the relevant cluster and the

irrelevant section contains documents in the irrelevant cluster in order of initial retrieved documents.

The re-ranking method is very simple. If current document exist in the relevant cluster, go to the re-ranked output otherwise sit in the down mid. Here is pseudo code for re-ranking:

```

FOR each document in the retrieved result list
  IF document exist in relevant cluster THEN
    CALL AppendToRelevantSection(document)
  ELSE
    CALL AppendToIrrelevantSection(document)
  END IF
END FOR

```

4. Experiments and evaluation

The goal of this section is to validate the proposed model and present strong evidence that the document re-ranking using clusters is one which can produce significant improvement over the method based on similarity search ranking alone.

In this paper we used a standard Persian text collection, named *Hamshahri*¹ [1, 8], which is built from a large number of newspaper articles according to TREC specifications. Hamshahri is the largest Persian text collection, so far.

4.1 Test collection

*Hamshahri*² is one of the first on line Persian newspapers in Iran. It has presented its archive to the public through its website since 1996. Darrudi et al. [8] employed a crawler to download available on line news from the website. The collection contains 166,774 articles covering the following subject categories: politics, city news, economics, reports, editorials, literature, sciences, Society, foreign news, sports, etc. Table 1 shows the complete attributes of this collection. It contains 65 natural language queries and relevance information of entry lists related to each query according to TREC specifications [1, 8]. We evaluated the proposed architecture with this corpus.

4.2 Results

As we mentioned before, a main problem with *local cluster analysis* is its inconsistency. A query-by-query TREC specific analysis on Hamshahri collection shows that it can improve seriously some queries and hurt others (See Fig. 4, Fig. 5) but our experiments express that total average precision on all queries can improve search results efficiently (See Table 2 and Fig. 3).

The precision curve of local cluster analysis in Fig. 4 predicts the best query improvement by using the proposed architecture on search results (See also Table 2). In this specific query, using the architecture improve 0.265 average precision quantity (See Table 2). It does agree with our goals in Section 1.2.

The precision curve in Fig. 5 contrary to Fig. 4 predicts the worst query by applying the proposed architecture. It's very instructive. As you see, even in the worst query while Recall ≤ 0.4 our system remain

high-precision (See also table 2) and it curves upper the best initial retrieval, although average precision 0.1982 decrease (See Table 2). Notice that in this paper we use best methods for initial retrieval [1, 2, 18]. It does agree with our second goal in Section 1.2.

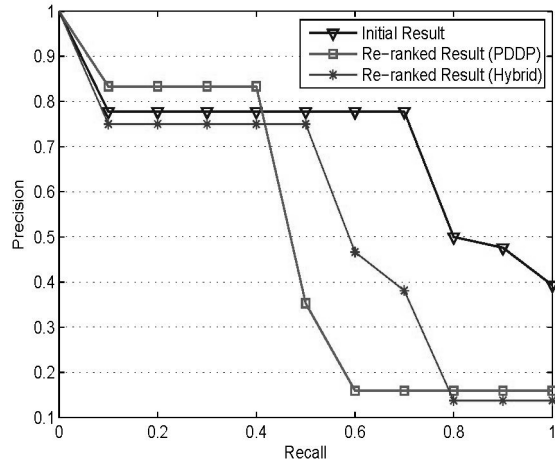


Figure 5: Interpolated Recall-Precision at the worst query. Average R-P for Initial, PDDP and Hybrid result are 0.6611, 0.4629 and 0.5119. See Table 2 for more detail.

5. Conclusion and future work

In this paper, we proposed a model for retrieval systems that is based on a simple document re-ranking method using Local Cluster Analysis. Experimental results on a Hamshahri collection show that it is more effective than existing techniques [2, 1, 3, 18]. Whereas we intended to exhibit the efficiency of the proposed architecture, we use single clustering method (PDDP) to produce clusters that are tailored to the information need represented by the query. Afterwards, utilize K-means with PDDP clusters as initial configuration (Hybrid approach) and showed that PDDP has potential to improve results individually.

Whereas in our approach, the context of a document is considered in the retrieved results by the combination of information search and local cluster analysis, cause first: relevant cluster tailored to the user information need and improve the search results efficiently, second: make high-precision system that contain more relevant documents at top of the result list. As it was shown, even in worst query that average precision 0.1982 percent decreased, still our system remain high-precision.

¹ <http://ece.ut.ac.ir/DBRG/Hamshahri/>

² <http://www.hamshahrionline.ir/>

We will pursue the work in several directions. Firstly, the current method for clustering search results is PDDP and hybrid K-means, however our experimental results had shown that PDDP has a great efficiency for our purpose but thence the total size of input in search results clustering is small, we can afford some more complex processing, which can possibly let us achieve better results. Unlike previous clustering techniques that use some proximity measure between documents, [20, 12, 30, 31] tries to discover meaningful phrases that can become cluster descriptions and only then assign documents to those phrases to form clusters. Use these concept-driven clustering approaches maybe a useful future work.

Secondly, I assumed that search results contain two clusters (Relevant and Irrelevant). In some cases irrelevant cluster can split into other sub-clusters by semantic relations. Get the optimal sub-clusters semantically can be produce better results.

Thirdly, we re-ranked results based on both clusters and after that choose better one manually. As we mentioned before, we conjecture that relevant cluster centroid must be near than irrelevant cluster centroid to the query. So we can choose which cluster centroid that toward to the query (relevant cluster).

Lastly, we evaluate the proposed architecture in ad-hoc retrieval. As we mentioned before, our approach is independent of initial system architecture so it can embed on any fabric search engine. One of the high-precision needful systems are Web search engines. Indisputable evaluate this approach on Web search engines can be a prominent future work.

6. References

- [1] A. AleAhmad, H. Amiri, F. Oroumchian, and M. Rahgozar. "Hamshahri: A standard persian text collection". *White Paper*, Database research Group, University of Tehran, 2008.
- [2] A. Aleahmad, P. Hakimian, F. Mahdikhani, and F. Oroumchian. "N-gram and local context analysis for persian text retrieval". *International Symposium on Signal Processing and Its Applications*, 2007.
- [3] H. Amiri, A. AleAhmad, F. Oroumchian, C. Lucas, and M. Rahgozar. "Using owa fuzzy operator to merge retrieval system results". *The Second Workshop on Computational Approaches to Arabic Script-based Languages*, LSA 2007 Linguistic Institute, Stanford University, USA, 2007.
- [4] D. Boley. "Principal direction divisive partitioning". *Data Min. Knowl. Discov.*, 2(4):325–344, 1998.
- [5] D. Boley, M. Gini, R. Gross, E.-H. S. Han, K. Hastings, G. Karypis, V. Kumar, B. Mobasher, and J. Moore. "Document categorization and query generation on the world wide webusing webase". *Artif. Intell. Rev.*, 13(5-6):365–391, 1999.
- [6] C. Buckley, J. Walz, C. Cardie, S. Mardis, M. Mitra, D. Pierce, and K. Wagstaff. "The smart/empire tipster ir system". In *Proceedings of a workshop on held at Baltimore*, Maryland, pages 107–121, Morristown, NJ, USA, 1996. Association for Computational Linguistics.
- [7] D. R. Cutting, D. R. Karger, J. O. Pedersen, and J. W. Tukey. "Scatter/gather: a cluster-based approach to browsing large document collections". In *SIGIR '92: Proceedings of the 15th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 318–329, New York, NY, USA, 1992. ACM.
- [8] E. Darrudi, M. R. Hejazi, and F. Oroumchian. "Assessment of a modern farsi corpus". In *Proceedings of the 2nd Workshop on Information Technology & its Disciplines (WTID)*, 2004.
- [9] F. Gelgi, H. Davulcu, and S. Vadrevu. "Term ranking for clustering web search results". In *WebDB*, 2007.
- [10] M. A. Hearst and J. O. Pedersen. "Reexamining the cluster hypothesis: scatter/gather on retrieval results". In *SIGIR '96: Proceedings of the 19th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 76–84, New York, NY, USA, 1996. ACM.
- [11] A. K. Jain, M. N. Murty, and P. J. Flynn. "Data clustering: a review". *ACM Comput. Surv.*, 31(3):264–323, 1999.
- [12] J. Janruang and W. Kreesuradej. "A new web search result clustering based on true common phrase label discovery". In *CIMCA '06: Proceedings of the International Conference on Computational Intelligence for Modelling Control and Automation and International Conference on Intelligent Agents Web Technologies and International Commerce*, page 242, Washington, DC, USA, 2006. IEEE Computer Society.
- [13] C. Jardine, N.;van Rijsbergen. "The use of hierarchic clustering in information retrieval". *Information Storage Retrieval*, 7, pages 217–240, 1971.
- [14] K. L. Kwok. "Higher precision for two-word queries". In *SIGIR '02: Proceedings of the 25th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 395–396, New York, NY, USA, 2002. ACM.

- [15] K.-S. Lee, Y.-C. Park, and K.-S. Choi. “Re-ranking model based on document clusters”. *Inf. Process. Manage.*, 37(1):1–14, 2001.
- [16] A. Leuski. “Evaluating document clustering for interactive information retrieval”. In *CIKM '01: Proceedings of the tenth international conference on Information and knowledge management*, pages 33–40, New York, NY, USA, 2001. ACM.
- [17] G. Mecca, S. Raunicha, and A. Pappalardo. “A new algorithm for clustering search results”. *Data & Knowledge Engineering*, 62(3):504–522, September 2007.
- [18] A. Nayyeri and F. Oroumchian. “Fufair: a fuzzy farsi information retrieval system”. In *AICCSA '06: Proceedings of the IEEE International Conference on Computer Systems and Applications*, 2006., pages 1126–1130, Washington, DC, USA, 2006. IEEE Computer Society.
- [19] F. Oroumchian, E. Darrudi, F. Taghiyareh, and N. Angoshtari. “Experiments with persian text compression for web”. *13th International World Wide Web conference*, New York, NY, USA, 2004.
- [20] S. Osinski and D. Weiss. “A concept-driven algorithm for clustering search results”. *IEEE Intelligent Systems*, 20(3):48–54, 2005.
- [21] S. M. Savaresi and D. L. Boley. “A comparative analysis on the bisecting k-means and the pddp clustering algorithms”. *Intell. Data Anal.*, 8(4):345–362, 2004.
- [22] C. Shah and W. B. Croft. “Evaluating high accuracy retrieval techniques”. In *SIGIR '04: Proceedings of the 27th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 2–9, New York, NY, USA, 2004. ACM.
- [23] J. Stefanowski and D. Weiss. “Carrot and language properties in web search results clustering”. In *AWIC*, pages 240–249, 2003.
- [24] A. Tombros, R. Villa, and C. J. V. Rijsbergen. “The effectiveness of query-specific hierarchic clustering in information retrieval”. *Inf. Process. Manage.*, 38(4):559–582, 2002.
- [25] C.-W. Tsai, T.-W. Liang, J.-H. Ho, C.-S. Yang, and M.-C. Chiang. “A document clustering approach for search engines”. In *ICSMC '06: Proceedings of the IEEE International Conference on Systems, Man and Cybernetics*, 2006., pages 1050–1055, Taipei, Taiwan, 2006.
- [26] C. J. Van Rijsbergen. *Information Retrieval, 2nd edition*. Dept. of Computer Science, University of Glasgow, 1979.
- [27] J. Xu. *Solving the word mismatch problem through automatic text analysis*. PhD thesis, Amherst, MA, USA, 1997.
- [28] J. Xu and W. B. Croft. “Query expansion using local and global document analysis”. In *SIGIR '96: Proceedings of the 19th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 4–11, New York, NY, USA, 1996. ACM.
- [29] J. Xu and W. B. Croft. “Improving the effectiveness of information retrieval with local context analysis”. *ACM Trans. Inf. Syst.*, 18(1):79–112, 2000.
- [30] O. Zamir and O. Etzioni. “Grouper: a dynamic clustering interface to web search results”. In *WWW '99: Proceeding of the eighth international conference on World Wide Web*, pages 1361–1374, New York, NY, USA, 1999. Elsevier North-Holland, Inc.
- [31] O. E. Zamir. *Clustering web documents: a phrase-based method for grouping search engine results*. PhD thesis, 1999. Chair-Oren Etzioni.
- [32] D. Zeimpekis and E. Gallopoulos. “Tmg: A matlab toolbox for generating term-document matrices from text collections”. In *Grouping Multidimensional Data: Recent Advances in Clustering*, pages 187–210. Springer, 2006.
- [33] H.-J. Zeng, Q.-C. He, Z. Chen, W.-Y. Ma, and J. Ma. “Learning to cluster web search results”. In *SIGIR '04: Proceedings of the 27th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 210–217, New York, NY, USA, 2004. ACM.
- [34] Y. Zhao and G. Karypis. “Empirical and theoretical comparisons of selected criterion functions for document clustering”. *Mach. Learn.*, 55(3):311–331, 2004.