NeSReC: A News meta-Search Engines Result Clustering Tool

Hassan Sayyadi
Web Intelligence Laboratory
Sharif University of Technology
sayyadi@ce.sharif.edu

Sara Salehi
Web Intelligence Laboratory
Sharif University of Technology
sarasalehi@ace.tju.ir

Hassan AbolHassani
Web Intelligence Laboratory
Sharif University of Technology
abolhassani@sharif.edu

Abstract—**Recent years have witnessed an explosion in the availability of news articles on the World Wide Web. In addition, organizing the results of a news search facilitates the user(s) in overviewing the returned news. In this work, we have focused on the label-based clustering aproaches for news meta-search engines, and which clusters news articles based on their topics. Furthermore, our engine for NEws meta-Search REsult Clustering (NeSReC) is implemented along. NeSReC takes queries from the users and collect the snippets of news which are retrieved by The Altavista News Search Engine for the queries. Afterwards, it performs the hierarchical clustering and labeling based on news snippets in a considerably tiny slot of time.**

Index Terms— **News, Clustering, Labeling, News Retrieval, News Mining**

I. Introduction

CURRENTLY, there exist a considerable number of online news sites and traditional news agencies provide electronic version of news on their web sites. To find news more effectively, special tools and search engines have been developed recently. For example News Feeder software, RSS standard and Google News site (which uses near 4500 news sources) can be mentioned. On the other hand, online news is specific type of public information available on the Web with unique features which result in different processing demands for gathering, searching and exploration on them in comparison to the ordinary web contents. Of those features, we can mention of the trustworthiness of news sources as well as the rapid update of them.

One of the problems of many search engines which is also true for many news search engines is the lack of the ability to categorize the search results before showing to a user. In fact results are ranked and displayed as a long list. When the user query is very specific, the results are not so much and then the user can rapidly find the relevant items. Nevertheless, unfortunately most of the time the query is general and ambiguous, resulting in a considerable number of items to display to the user. Average number of terms in a query is near 3 and most of the times a user only checks the first 3 results [2]. For an example when we use Google and search for "jaguar", pages talking about feline mammal are displayed in ranks 10, 11, 32 and 71. Therefore, it is more appropriate to cluster the results before showing to a user. Such clustering helps in resolving two important issues. Firstly, it helps user to have an overall view of the results and to refine the query needed more appropriately. Secondly, one of the pitfalls of the link-analysis based algorithms is when results are from different categories. In such a case the top results are normally, belong to one of the categories and considering the fact that users usually only check top ranked results they may miss many relevant results.

Van Rijsbergen [12] was the first one that investigated the effect of clustering hypothesis in Information Retrieval for query-based systems: documents similar to each other are relevant to similar queries with high probability. In fact the main point is that relevant documents are more similar to each other compared to non-relevant ones. Based on this hypothesis we can use clustering in two different ways:

1) Before retrieval which has a long history and we can mention Scatter as a famous example,
2) After retrieval which is what we are interested on in this paper.

It should be noted that the web search result clustering has important differences with traditional text clustering. One of the main differences is the existence of links between pages. The other important difference is the need to do fast processing and dealing with a multi-line abstract instead of whole document. According to [10] desire characteristics for clustering of search results are:

- **No need for all pages to be clustered:** Not all pages should be clustered, because some pages can be not related to any generated clusters.
- **Overlapping:** Clusters may have some overlaps, because one page can point to several topics. Consequently, it can fall in several clusters. Furthermore, clusters overlapping should be as few as possible. Because two clusters, which have great overlaps generally, point to the same topic, they should be merged.

Manuscript received October 13, 2006. This work was supported in part by the Web Intelligence Research Laboratory, Sharif University of Technology.

- **Incremental Clustering:** Finally, because of handling time complexity, incremental clustering methods are more satisfactory.

One of the reasons that commercial search engines are not doing this is the high runtime complexity of it. It should be noted that some search engines like Altavista[1] does aggregation in very simple form. In NorthernLight[2] results are divided in some Custom folders. Such division is not intelligent and is based on attributes like page type (personal page, product page, etc.), language, domain, site and so on. Better commercial examples are Kartoo[3], Grokker[4], Mooter and specifically Vivisimo which apply simple clustering algorithms. They also have very user-friendly interfaces. In addition, Clusty[5] uses Vivisimo[6] but its internal clustering algorithm is not known. The result of Vivisimo for query "iran" is shown in figure 1.

In section 2 we have a review of the related works in which we are mainly focused on clustering and labeling tasks. Then in section 3 our proposed method is introduced and elaborated. Section 4 discusses on evaluation results on the implemented system (NeSReC). Conclusions and future works is given in the final section.

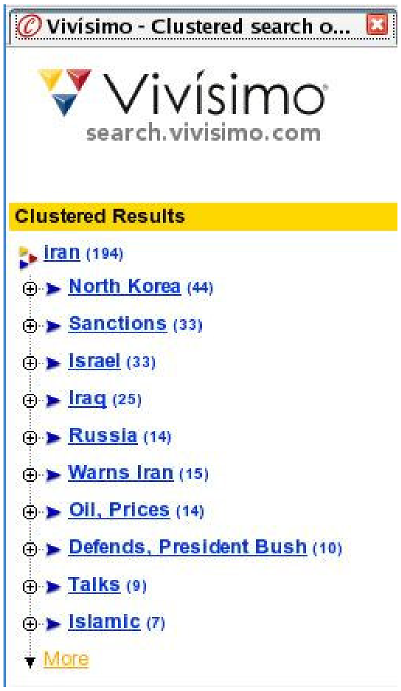


Fig. 1. Vivisimo result for query "iran"

II. Related Works

For the first time clustering and its effects on the retrieval was reported in [13]. Its main purpose is to create a directory for documents, which facilitates users' access to them. It first divides the repository to a small number of clusters. Then a user selects some of them and they are combined and a sub-repository is constructed. Such operations are repeated until user is satisfied. To reduce the cost of algorithm for large repositories a sampling approach is taken. The study in [14] is a good but rather old study on it.

A. *Clustering*

Clustering is done in different levels of a text for different purposes. As noted above one usage of clustering is for providing better navigation. However, a clustering also can be used to do summarization in the level of paragraph [4] or even sentences [3] for event detection. Approaches for better review of results can be categorized in the following two [1]:

1) Document based approach
2) Label based approach

Document based approaches are those they does clustering based on similarity of texts like their terms vector similarity. After clustering is done, some words or phrases of the texts in a cluster are combined to build a label for the cluster like [7,8]. Clusters made this way have no overlap and the quality of their labels is highly dependent on the quality of the clusters themselves. Works in the document based clustering methods are different from the following two aspects:

1) The clustering algorithm they use
2) Measurement of distances between texts or clusters

Since human better understands hierarchical structures, the tendency for creation of clusters in a hierarchical structure is high. In addition, specification of parameters like number of clusters and the similarity threshold is a difficult task. Therefore, normally the generated labels do not satisfy users. For this reason, such techniques are not used anymore.

One of the document-based methods is what reported in [5], which first eliminates the stop-words and does stemming to unify words like "went" and "go". Then n words having higher TF values are selected from news. If S1 and S2 are words extracted from two news then similarity value of them are intersection of S1 and S2 divide by n. In addition, distance measure is easily computed from the similarity value:

$$Sim(n_1,n_2)=\frac{|s_1 \cap s_2|}{n}$$

$$Dissim(n_1,n_2)=1-Sim(n_1,n_2)$$

Using such a measure k-nearest neighbor algorithm is applied on the news. Furtheremore, the single-link algorithms is reported in the paper and it is claimed that a combination of them produces best results.

STC test is also a linear incremental algorithm which instead of treating a document as a set of words considers it as a sequence of words and then combines clusters based on their

[1] Http://www.altavista.com/
[2] Http://www.northernlight.com/
[3] Http://www.kartoo.com/
[4] Http://www.grokker.com/
[5] Http://www.clusty.com/
[6] http://www.vivisimo.com/

intersection of postfixes of their documents. It allows overlaps of clusters that can handle noise efficiently and let documents themselves specify the number of clusters.

In [10] link structure of pages is used as a characteristic for clustering retrieved results of a search. In this method, firstly, the pages are loaded and their output links are extracted. In-links are also gathered using a standard search engine. Such two vectors of in-links and out-links are used to compute the similarity of two documents.

Authors of [9] as like [11] used classification instead of clustering. Statistical analysis is used to learn some specifications of some classes (for example from DMOZ[7] categories) and then the trained system is used for classification of new data. This method is highly dependent on the train set and therefore can't be extended for the whole web.

To create clusters in [15] in the first step a hierarchy for concepts is created and then assigns documents to those concepts. [16] uses a special tool (which is based on a large repository of documents) to find concepts related to a user query and uses fuzzy C-Mean for clustering.

In [6] Lycos search engine is used and the performance of two standard indexing methods (N-gram and vector model) in clustering are compared and it is shown that the first method is resistant to noise. In addition, it is mentioned that automatically created clusters even when the performance of algorithm is high has more categories than what a human may build. For example, it is possible for a human to put all news of a news agency in a cluster while an automatic system makes many smaller clusters for such news based on different topics they cover. Using N-grams results in fewer numbers of clusters and from this point of view acts like a human user. During the clustering fuzzyfication is used to reduce the unwanted results for points in boarders in both indexing and labeling phases.

One the other hand, in label based approach, words and informative phrases are first extracted using some statistical analysis like word occurrences and then create clusters base on selected labels. Vivisimo and Mooter, which have satisfactory results, use label-based approaches.

[11] at first discovers important words based on a training set and then to each of them assigns related words. It uses a within cluster similarity measure to evaluate the quality of output. For naming a cluster, also important words are used. Nevertheless, because of the difference between real words and N-grams the quality of naming is not so good.

Furthermore, Authors in [1] proposed a label-based method used named entities but with defining some new measures have reached to a more effective model. As they noted it is shown that TF-IDF cannot remove high frequent un-important words. Their new measures try to overcome this problem. Scoring of a phrase comes from two factors: local factor and global factor. In the traditional TF-IDF scoring TF is the local and IDF is the global factor. In the new proposal two new local factor named LRDF and OLF as defined below is used:

$$LF_i^{LRDF} = \log(1 + DF_{R,i})$$

$$LF_i^{OLF} = DF_{R,i} * \log(\frac{|R|}{DF_{R,i}})$$

Also a new global factor named OGF as below is defined:

$$DF_i^{OGF} = \frac{DF_{r,i} / |R|}{DF_{D,i} / |D|}$$

In their experiments, it is shown that OLF-OGF produces very good results. In the context of news the uses of named entities is very effective since unlike ordinary web pages a news is about an event in a specific location for a given person or group in a specific time. As results extraction of those entities like time and places is very beneficial.

B. Labeling

For the labeling two points are important [1]:

1) readability of labels which facilitate understanding by users, and
2) their conciseness in representing related documents.

In fact mis-labeling results in reduction of both Precision and Recall measures. It is also mentioned that the stemming quality has a strong effect on labeling [14]. In Grouper [7] labeling are based on the same phrases which are used during clustering. In [17] for cluster naming super concepts and words in the title of news are used but it is only applicable for Japanese language.

III. Proposed Method

Our approach is for clustering as well as labeling of news which is architected as a meta-search engine for news. In the other words, it is proposed for engines that by accepting user queries forward them to a search engine and process the results to cluster them and make labels for those clusters based on the snippets and title of news. As mentioned in [1] labeling based clustering makes more efficient clusters than the traditional methods, so our proposed method follows this approach. The process contains following tasks:

1) receiving initial results (texts, links)
2) extraction of candidate titles
3) ranking of the titles
4) clustering
5) display

The initial results are made by using an ordinary search engine. Display of the results is also discussed when we explain the experimental results. Therefore, in this section we focus on the following two important tasks:

- Extraction of titles and ranking them
- Clustering

[7] Http://dmoz.org/

A. *Extraction of Titles*

In this phase at first, the candidate titles should be extracted from news' snippets. Then such titles are ranked based on some factors which is explained shortly and finally some of them which satisfy selection criteria are selected. Since the best titles are noun phrases in our method, we only extract noun phrases from the snippets. In this process, un-important words are eliminated. However, as it is shown in the experimental results stemming or rooting has not much effect on the quality of clustering.

To score titles we use three factors. As like many other methods, our first two factors are local and global factors. Their sole purpose is to eliminate very frequent or very rare words since they can't be good titles for clusters. Therefore, the local factor is a score for frequency of a term and the global factor scores a term's rareness. The third factor is the length of a phrase which is used to score longer titles since they better represent a cluster than shorter ones. The final score for each phrase (or word) is computed as the multiplication of these factors:

$$R(t_i) = LF(t_i) * GF(t_i) * SF(t_i)$$

Here *R(t$_i$)* represent the score of title t_i and *LF*, *GF* and *SF* are local, global and length factors, respectively.

- Local factor is what presented in [1]. To balance its effect with other factors we also use a logarithmic form of term frequencies (DF) as below:

$$LF(t_i) = DF_i * \log \frac{|R|}{DF_i}$$

Here |R| is the number of extracted news and DF_i is the document frequency of term i which expresses the number of documents that contain term i.

- Our global factor is:

$$GF(t_i) = \log \frac{|R|}{DF_i}$$

- For the length factor we propose three function of linear, exponential and logarithmic forms:

$$SF^{linear}(t_i) = \max(\alpha, sizeof\ (t_i))$$

$$SF^{power}(t_i) = \sqrt{sizeof\ (t_i)}$$

$$SF^{log}(t_i) = \log(sizeof\ (t_i)) + 1$$

α parameter is a threshold value to control the maximum value obtained from the first equation and the *sizeof(ti)* is the number of words constructing term *i*.

After computation of scores of titles, they are sorted. Now we should select K titles among them that also satisfy following condition:

- If a title is substring of another title and their size ratio is higher than a ß threshold then the shorter title is eliminated from the list of candidates.

This condition is applied to prevent the selection of almost similar titles for different clusters.

B. *Creation of Clusters*

As said before in the clustering by labeling, clusters are made according to the selected label. Therefore, for each label t_i a cluster *Ci* is created and news having such a title is put on that cluster. It is clear that a news can be put in several clusters. Since a news may point out to different subjects this behavior is rational and is also in the interest of users. In our approach a news is belonged to a title when:

- it completely has that title
- it has a substring of the title with the length ratio of more than ß threshold

After completion of this process for a level of clustering, it can be applied to each cluster to obtain a hierarchical clustering structure. To control the level of hierarchy number of documents in a cluster is used and when such number is less than σ it is not anymore sub-clustered. In addition, we can define another h threshold to control the height of the tree and reduce complexity of having a very tall tree of clusters.

IV. Experimental results

In this section, our meta-search engine named NeSReC which is developed for the evaluation of our proposed method. It is a meta-search engine that clusters the results of a search for news. To have an initial result set Altavista news search is used. When a user search some terms, Altavista is called to receive titles, addresses and snippets of the relevant news. Then candidate titles are selected from snippets. As mentioned, titles are noun phrases and to distinguish them we use JMultilingua[8] tool. Noun phrases as well as words in them are considered as candidate titles. A title with less than three letters or being a stop-word is eliminated. Our stop-word list contains 1000 words.

Stemming or rooting has not so much effects on the titles selection. For example, in one of our experiments without stemming, we reached to 760 titles while stemming reduced to 670 words. Considering its rather high execution cost it is not so much effective to be considered. In addition, it is noted that such a words gains low ranks by our ranking model and ultimately they are not selected so the existence of them has no harm on the quality of results. The main point is that without stemming, the execution time is less than 1 second in a typical run but when stemming is considered by WORDNET tool, it raises to 20 seconds.

The reasons why stemming has not much effect on news

[8] http://web.media.mit.edu/~hugo/montylingua/

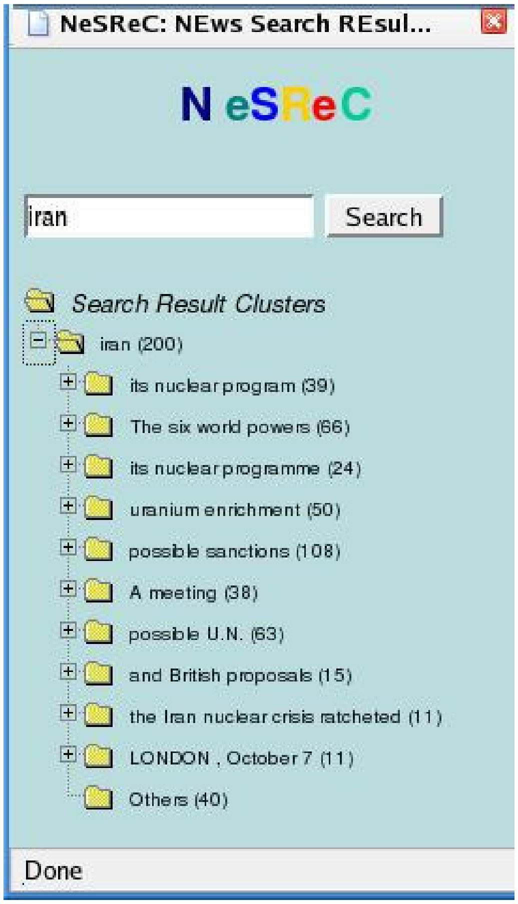
clustering is clear. It is that the news agencies has many cross references to each other a news which is published by different sources has a main source which publishes it at first and others use very similar wording to the original source.

We also see that the logarithmic function for size factor leads to the best results for label selection. After extraction of label, the top 10, which also satisfy condition, are considered as the labels for the clusters in the given level. In our experiments, we set ß to 0.5.

After titles and clusters are constructed, those clusters having more than σ news are divided to lower level clusters with the same method. We set σ to the number of extracted news divided by 10 times clustering level:

$$\sigma = \frac{ResultSize}{20 * ClusterLevel}$$

Additionally, we limit the height of the tree to two levels. In the other words, for 200 news, the threshold of number of news for clusters in level 1 is 10 and for level 2 is 5 the reason for limiting to two levels is to reduce the complexity of algorithm and two levels for news seem satisfactory. However, the method can be customized for each user based on his preferences. The first level of result clustering and labeling in NesRec for user query "iran" is shown in the figure 2.



NeSReC: NEws Search REsul...

N eSReC

iran

Search

Search Result Clusters

- iran (200)
  - its nuclear program (39)
  - The six world powers (66)
  - its nuclear programme (24)
  - uranium enrichment (50)
  - possible sanctions (108)
  - A meeting (38)
  - possible U.N. (63)
  - and British proposals (15)
  - the Iran nuclear crisis ratcheted (11)
  - LONDON , October 7 (11)
  - Others (40)

Done

Fig. 2. NeSReC result clusters for query "iran"

V. Conclusion

The main merit of our method is its simplicity while at the same time produces high quality results. The difference of our method with what reported in [1] is that our proposed method is used for meta-search engines and its clustering based on snippets, while their method is used for ordinary search engines. Furthermore, they limit labels to named entities for person names, places, organizations, and artifacts. It is true that such entities are important for news but limiting labels to them is not a good idea. In our method the only limitation is that the selected title should be noun phrase which seems a logical assumption. The other advantage of our method is that the created clusters are balanced which means they have almost same number of news. In addition, the number of un-categorized news is low and is near 25% of total size of retrieved news in average which is shown in diagram in fig. 3. Furthermore, these non-clustered news has low ranks in search process. Such results confirm this points that the size of cluster for a news is important for its rank. Moreover, Inasmuch as NeSReC engine generate 10 clusters in each level, the average size of first level clusters is near 15% of total size of retrieved news as shown in fig. 4. Although we escape stemming and cluster overlapping criteria, our less complicated model contributes to conspicuous results.

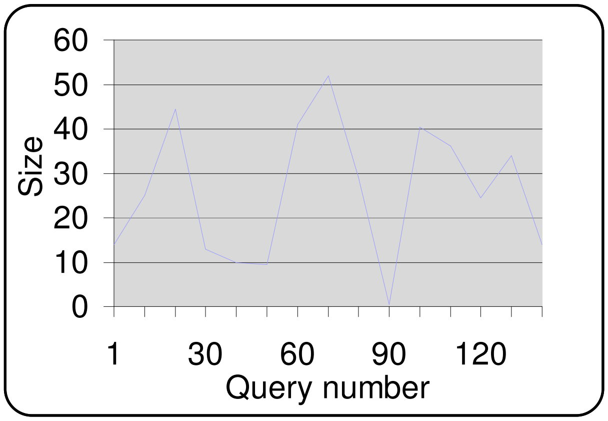


Fig. 3. Ratio of non-clutered news to the total size of retrieved news

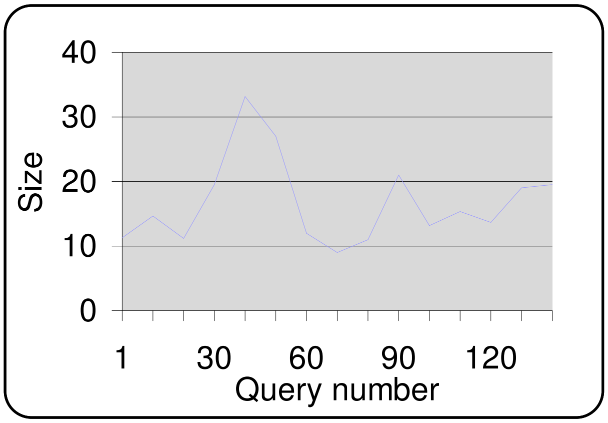


Fig. 4. Ratio of average size of first level clusters to the total size of retrieved news

In the future, we will work on some criteria for clusters overlaps to avoid similar clusters with different names. In addition, considering clusters overlapping avoid further simple problems which may arise from escaping stemming.

REFERENCES

[1] H. Toda, R. Kataoka, "*A search result clustering method using informatively named entities",* In: WIDM '05: Proceedings of the 7th annual ACM international workshop on Web information and data management, New York, NY, USA, ACM Press (2005) 81–86

[2] C. Silverstein, M. Henzinger, H. Marais, and M. Moricz, "*Analysis of A Very Large Web Search Engine Query Log*", SIGIR Forum, 33(1), 1999.

[3] M. Naughton, N. Kushmerick, J. Carthy, "*Clustering sentences for discovering events in news articles*", In: ECIR. (2006) 535–538

[4] V. Hatzivassiloglou, J. Klavans, M. Holcombe, R. Barzilay, M. Kan, K. McKeown, "*Simfinder: A flexible clustering tool for summarization"*, (2001)

[5] N. A. Shah, E. M. ElBahesh, "*Topic-based clustering of news articles*", In: ACM-SE 42: Proceedings of the 42nd annual Southeast regional conference, New York, NY, USA, ACM Press (2004) 412–413

[6] Z. Jiang, A. Joshi, R. Krishnapuram, and L. Yi., "*Retriever: Improving Web Search Engine Results Using Clustering*", In Managing Business with Electronic Commerce 02.

[7] O. Zamir and O. Etzioni, *Grouper: A Dynamic Clustering Interface to Web Search Results*". In Proceedings of the Eighth International World Wide Web Conference, Toronto, Canada, May 1999.

[8] M. A. Hearst and J. O. Pedersen, "*Reexamining the Cluster Hypothesis: Scatter/Gather on Retrieval Results*", in Proceedings of the 19th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR96), 1996, pp 76-84.

[9] H. Chen and S. Dumais, "*Bringing Order to The Web: Automatically Categorizing Search Results*", in Proceedings of the CHI 2000 Conference on Human Factors in Computing Systems, pp. 142–152, 2000.

[10] Y. Wang and M. Kitsuregawa, "*Link Based Clustering of Web Search Results*", in Second International Conference on Advances in Web-Age Information Management (WAIM), 2000.

[11] H. Zeng, Q. He, Z. Chen, W. Ma, and J. Ma, "*Learning to Cluster Web Search Results*". In Proceedings of ACM SIGIR '04, 2004.[19] Douglas Cutting, Jan O. Pedersen, David Karger, and John W. Tukey. "Scatter /Gather: A Cluster-Based Approach to Browsing Large Document Collections". In Proceedings of SIGIR'92, pages 318-329, Copenhagen, Denmark, June 21-24 1992.

[12] V. Rijsbergen, C. J., "*Information Retrieval*". London: Butterworths; 1979.

[13] M. A. Hearst and J. O. Pedersen, "Reexamining the Cluster Hypothesis: Scatter/Gather on Retrieval Results" , in Proceedings of the 19th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR96), 1996, pp 76-84.

[14] A. V. Leouski and W. B. Croft, "An Evaluation of Techniques for Clustering Search Results". Technical Report IR-76, Department of Computer Science, University of Massachusetts, Amherst, 1996.

[15] K. Kummamuru, R. Lotlikar, S. Roy, K. Singal, and R. Krishnapuram, "A Hierarchical Monothetic Document Clustering Algorithm for Summarization and Browsing Search Results". In Proceedings of the 13th international conference on World Wide Web, pages 658-665, New York, NY, USA, 2004.

[16] O. Hoeber and X. D. Yang, "Visually Exploring Concept-Based Fuzzy Clusters in Web Search Results", In Proceedings of the Fourth International Atlantic Web Intelligence Conference, 2006.

[17] T. Noda, H. Ohshima, T. Tezuka, S. Oyama, and K. Tanaka, "Automatic Extraction of Topic Terms for Web Search Result Clustering", The 1st China-Kyoto Student Workshop on Digital Content and Web Computing (CKSW2006), Beijing, China, March 2006.