C4-2: Combining Link and Contents in Clustering Web Search Results to Improve Information Interpretation

Yitong Wang and Masaru Kitsuregawa
Institute of Industrial Science, The University of Tokyo
Komaba 4-6-1, Meguro-ku, Tokyo, 153-8505 Japan
{ytwang, kitsure@tkl.iis.u-tokyo.ac.jp}

Abstract

With information proliferate on the web, it is far beyond human's ability to digest this huge, heterogeneous information, e.g. locating related resources as well as providing accordingly information interpretation. While web search engine could retrieve information on the Web for a specific topic, users have to step a long ordered list in order to locate the needed information, which is often tedious and frustrating. In this paper, we investigate how to combine link and contents analysis in clustering web search results to improve information interpretation for a specific topic. By filtering some irrelevant pages, the proposed approach clusters high quality pages in web search results into semantically meaningful groups with additional tagging keywords to facilitate users' accessing and understanding. We especially study the contribution of link and contents to clustering procedure. Preliminary experiments and evaluations are conducted to investigate its effectiveness. ***Keywords: link analysis, co-citation, coupling, anchor text, snippet***

1. Introduction

With information proliferate on the web as well as popularity of Internet, how to locate related information as well as providing accordingly information interpretation has created big challenges for research in the fields of data engineering, IR as well as data mining due to features of Web (huge volume, heterogeneous, dynamic and semi-structured etc.)

While web search engine could retrieve information on the Web for a specific topic, users have to step a long ordered list in order to locate the needed information, which is often tedious and less efficient due to various reasons like huge volume of information; users differ with requirements and expectations for search results; sometimes a search request cannot be expressed clearly with few keywords or users may be just interested in "most qualified" information or one peculiar part of information returned etc. Especially, synonym (different terms have similar meaning) and homonym (same word has different meanings) make things more complicated. In general, the resources locating (recall and precision) and accordingly interpretation of search results of current search engines are far from satisfying.

Many works [1,2,3,15,18] argued that links between web pages could provide valuable information to determine related pages (with query topic) and try to explore link analysis to improve quality of web search process or mine useful knowledge on the web. Kleinberg proposed in [1] that there are two kinds of pages in search results: "hub page" and "authority page" and they reinforce each other. HITS algorithm in [1] might provide a solution to the above challenges, however, sometimes one's "most authoritative" pages are not useful for another one and further investigations on it are still in high demand.

We think clustering web search results could help a lot. The goal of our work is to cluster high-quality pages in web search results into more detailed, semantically meaningful groups with tagging keywords to facilitate user's searching and information interpretation. By doing so, it is helpful for users to identify the main ideas around the topic on the web since users could have an overview or just select the interested group to view. When we talk about *web search results /search results*, we mean web pages returned from web search engine on a specific query topic. We use URLs or pages interchangeably when referring to search results.

Although traditional document clustering algorithms that are based on term frequency could be applied to web pages, it does not adapt well to web environment since web pages are more miscellaneous comparing with text corpus. Our target is clustering web search results for a specific topic to improve its information interpretation, which is different from clustering a web page corpus that might cover a wide range of topics.

We also emphasize some requirements for clustering of web search results that has been stated in [7]: relevance and overlap. Relevance means that clustering should separate related web pages from irrelevant ones, that is to say, *not all web pages* but high-quality pages in search results need to be clustered and overlap means that one web page could *belong to more than one cluster* since it could have more than one topic.

In [22], we have proposed a link-based clustering approach by co-citation and coupling analysis. According

to preliminary experimental results, link-based clustering could cluster web search results into several more detailed groups. However, it suffers the shortcoming that pages with few links will not be clustered, that is low recall. In this paper, we investigate how to combine link and contents in clustering algorithm to overcome the shortcoming and especially study their contributions in clustering process.

The paper is organized as follows. Section 2 is an assessment of previous related works of clustering in web domain. In Section 3, we describe clustering algorithm by combing link and contents analysis. Subsequently in Section 4, we report experimental results and evaluations. The contents used in our approach include snippet and anchor text attached with each URL in search results. The paper is concluded with summarizing and future work.

2. Background

Cluster analysis has a long history and serves for many different research fields, like data mining, pattern recognition as well as IR. Vector Space Model, also called *TFIDF* method, which is based on terms frequency is the commonly used one for document representation in IR. *K-means* and *agglomerative hierarchical clustering* are two fundamental clustering methods in IR. The advantage of K-means is its speed and the disadvantage is that the quality and structure of final clusters will depend on the choice of k value and k initial centroids when clustering n data points into k groups. According to [5], hierarchical clustering produces "better" clusters but with high cost.

2.1 Prior Related Work on Clustering Search Results

Related works can be classified into following categories: clustering hypertext documents in a certain information space and clustering web search results. As for the latter one, some works are basing on the whole document and some works are focusing on clustering snippet attached with each URL in search results in order to achieve high speed. The snippet of one page is usually the first several sentences of its contents and is often considered as a good summary to capture the main idea of the page under consideration.

It is in [9] that a hierarchical network search engine is proposed to cluster hypertext documents to structure a given information space for supporting various services like browsing and querying. All hypertext documents in a certain information space (e.g. one website) can be clustered into a hierarchical form based on the contents as well as the link structure of each hypertext document. By considering about links within the same website, related documents in the same website could be grouped into one cluster. However, our target is not a general situation, but search results classification, which clusters search results into more narrow and detailed groups. In [11] clustering hypertext documents by *co-citation analysis* (its explanation is in Section 2.2) is explored. Scatter/Gather [11] proposed a document browsing system based on clustering, using a hybrid approach involving both k-means and agglomerative hierarchical clustering. Other approaches like hyper-graph partitioning [14], which applying data mining technique to terms in search results are all term-based clustering approaches.

In [7], an algorithm called Suffix Tree Clustering (STC) is proposed to group together snippets attached with web pages in search results. The algorithm use techniques that construct a STC tree within a linear time of number of snippets. Each node in this tree captures a phrase and associates it with snippets that contain it. If one node in the tree associates more than one snippet, the associated snippets form a base cluster. After obtaining base clusters in this way, final clusters are generated by iteratively merging two base clusters if they share majority (50%) members. The advantage of STC algorithm is that it can capture the word order in snippet, which is useful for identifying similar pages. Since snippets usually bring noises and outliers, an algorithm called fuzzy relational clustering (RFCMdd), which is based on the idea of identifying k-medoids, is proposed in [8] to compensate the work in [7] with the ability to process noises and outliers brought by snippets. However, snippets are not always available in search results and they are also not always a good representation of the whole documents for their subjectivity. Moreover, the fact that in STC algorithm, two snippets is clustered into same group even if they only share one word will lead to very high overlap and in turn generate a big cluster.

2.2 Link and Contents analysis

Hyperlink is helpful since it demonstrates the other people's objective evaluation of the page it links to. It is also useful to overcome Spam problem.

Co-citation [21] and bibliographic **coupling** [20] are two more fundamental measures to be used to characterize the similarity between two documents. **Co-citation** measures the *number of citations* (*out-links*) *in common* between two documents and **coupling** measures the number of document (*in-links*) that cites both of two documents under consideration.

In the Figure 1(a), *p* and *q* ***co-cite*** *Q* and *R* and their co-citation frequency is 2; *P* and *R* ***are coupled*** by *r* and their coupling frequency is 1.

The anchor text for a hyperlink is the text that implies the real link appearing in search results. During our experiment, we find that anchor text for a link usually

include very important keywords to imply the main topic of the page it links to. Usually, the anchor text as well as snippet could provide a reasonable summary of the page.

As for link analysis, since there are potential " hubs" and "authorities" in web search results, both co-citation and coupling are considered in the proposed approach. Just as indicated in [22] that in addition to keywords, common links shared by different pages could be invaluable to judge the similarity between them. That is, pages that are sharing common keyword, out links or in-links are very likely to be related in more narrow way.

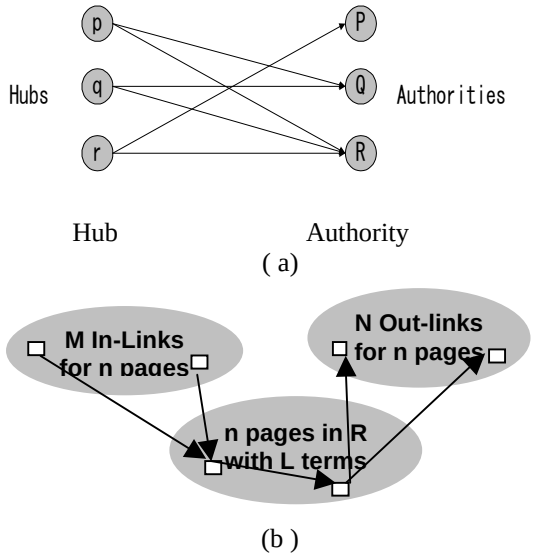


Figure 1. Potential Hub pages and Authority pages in Web search results

3. Combining Link and Contents Analysis in Clustering

By contents analysis and link analysis (co-citation and coupling), our approach clusters search results based on common terms, in-links and out-links they shared, which is a natural extension of the approach proposed in [22]. In the rest of our discussion in the paper, we have several notations: n, m, M, N, L are positive integers, *R* is the set of specified number of search results for a topic. We use n to denote specified number of search results used for clustering, m to denote specified number of in-links extracted for each URL/page in *R*. M, N, L denote total number of distinct in-links and out-links extracted as well as keywords extracted from snippet and anchor text by stemming process for all *n* pages in *R* respectively. This is depicted in Figure 1 (b).

1) Representation of each page *P in R*

Each web page *P* in *R* is represented as 3 vectors: P_{Out} (*N*-dimension), P_{In} (*M*-dimension) and P_{KWord} (*L* – dimension). The *ith* item of vector P_{Out} indicates whether *P* has the correspondent out-link as the *ith* one in N out-links. If yes, the *ith* item is 1, else 0. Identically, the *jth* item of P_{In} indicates whether *P* has the correspondent in-link as the *jth* one in M in-links. If yes, *jth* item is 1, else 0. The *kth* item of vector P_{KWord}, indicates the frequency of the corresponding *kth* term of L appeared in page *P*.

2) Similarity measure

We adopt the traditional *Cosine* measure to capture common links (in-link and out-link) or common terms shared by pages *P, Q* that is under consideration. The similarity of two pages includes three parts: out-link similarity *OLS(P,Q)*, in-link similarity *ILS(P,Q)* and contents similarity *CS(P,Q)*.

The *OLS(P,Q)*is defined as:

$$(P_{Out} \bullet Q_{Out}) / (\| P_{Out} \| \| Q_{Out} \|) \quad (1)$$

$\| P_{Out} \|^2 = (\sum_1^N P_{Out\,i}{}^2)$ (Total number of out-links of *P*),

$\| Q_{Out} \|^2 = (\sum_1^N Q_{Out\,i}{}^2)$ (Total number of out-links of Q),

The *ILS(P,Q)*is defined as:

$$(P_{In} \bullet Q_{In}) / (\| P_{In} \| \| Q_{In} \|) \quad (2)$$

$\| P_{In} \|^2 = (\sum_1^M P_{In\,i}{}^2)$ (Total number of in-links of *P*),

$\| Q_{In} \|^2 = (\sum_1^M Q_{In\,i}{}^2)$ (Total number of in-links of Q),

The *CS(P,Q)* is defined as:

$$(P_{KWord} \bullet Q_{KWord}) / (\| P_{KWord} \| \| Q_{KWord} \| \quad (3)$$

$\| P_{KWord} \|^2 = (\sum_1^L P_{KWord\,i}{}^2)$ (Total terms weight of P)

$\| Q_{KWord} \|^2 = (\sum_1^L Q_{KWord\,i}{}^2)$ (Total terms weight of Q)

Dot product in the above formula (1), (2) and (3) is to capture the common out-links, in-links and keywords shared by *P* and *Q*. || ||is length of vector.

3) Usage of Centroid

Centroid or center point *C* is used to represent the cluster *S* when calculating similarity of page *P* with cluster *S*. Centroid is usually just a logical point.

$$C_{out} = 1/|S| \sum_{P_i \in S} P_{i\,Out}\,,\quad C_{In} = 1/|S| \sum_{P_i \in S} P_{i\,In}$$

$$C_{Kword} = 1/|S| \sum_{P_i \in S} P_{iKWord} \quad (4)$$

|S| is number of pages in cluster S.
So the similarity of pages p and clusters S, ***Sim(P, S)*** is defined as:
P1* *OLS (P,C)*+P2* *ILS(P,C)*+P3* *CS(P,C)*, (5)
where the sum of P1, P2 and P3 is 1. By varying value of P1, P2 and P3, we could study the contribution of out-link, in-link as well as keywords in clustering process in depth.

4) Clustering method

We extend standard K-means to meet requirements for clustering of web search results as well as to overcome disadvantages of K-means. Our clustering method is as follows:

a) Filter irrelevant pages
   By filtering some irrelevant pages, we could improve the *precision of final results*.
b) Define *similarity threshold*
   Similarity threshold is pre-defined to control the process of assigning one page to a cluster. Since similarity is meant to capture the common links and terms shared by different pages, similarity threshold could be easily defined and adjusted.
c) Assign each page to clusters
   Each page *is assigned to existing clusters* when the similarity between the page and the correspondent cluster is above the *similarity threshold*. If none of current existing clusters meet the demand, the page under consideration becomes a new cluster itself. Centroid vector is used when calculating the similarity and it is *incrementally* recalculated when new members are introduced to the cluster. While one page could belong to more than one cluster, it is limited to *top 3* clusters based on similarity values. This is according to empirical results. All pages that join clustering procedure are processed sequentially and the whole process is iteratively executed until it converges (centroids of all clusters are no longer changed). Preliminary experimental results show that final results are insensitive to the processing order; however, further investigation and proof are needed, which is not discussed here.
d) Generate final clusters by merging *base clusters*
   When the whole iteration process converges, *base clusters* are formed. Final clusters are generated by recursively merging two base clusters if they share majority members. ***Merge threshold*** is used. Centroid vectors for all clusters are also calculated during merging process in the same way.

5) Tagging for each cluster

Automatic tagging is the main difference between clustering and classification, which is given manually beforehand. It is very important for user to know the main topic of the group/cluster by just a glance of the tagging words. After the clustering procedure described above, we obtained final clustering results as well as information of centroid vector for each final cluster. Say for cluster *S*, *C* is its centroid, which include three vectors. By three vectors of C: C_{out}, C_{In} and C_{Kword}, it is easy to know out-links, in-links and especially keywords (terms) that have higher values and are most shared by the members of Cluster *S*. The most shared keywords could well convey the main topic of the cluster. Clustering and automatic tagging will reflect a kind of web evolution.

The convergence of the approach is guaranteed by K-means itself since our extension does not affect this aspect. The algorithm described above also has same *time complexity* (*O(nm)*) as standard K-means, where *n* is the number of pages that join clustering procedure and *m* is the number of iterations needed for clustering process to converge. Since $m << n$, the proposed approach is in a linear time to the number of URLs/ pages that join clustering procedure. The two parameters introduced in the above clustering algorithm that may affect quality of final results are *similarity threshold* and *merge threshold*. In [22], we have conducted thorough experiments to investigate their effects on the final results and indicated that for most topics, similarity threshold 0.1 and merging threshold 0.75 are our recommendations.

4. Experiments and Evaluations

In this part, we mainly report our experimentation on the proposed approach as well as its accordingly evaluations. We choose topic "jaguar" and "chair" for detailed testing. Just as mentioned, by varying the parameters in formula (5), it is possible to try different clustering patterns for a specific topic. Here, by "clustering pattern", we mean link-based clustering (denoted by "L" with P1, P2, P3 as 0.5, 0.5, 0) or term-based clustering (denoted by "C" with P1, P2, P3 as 0, 0, 1)) or combining links and contents (denoted "M" with P1, P2, P3 as 0.2, 0.3, 0.5). The choice of parameter values 0.2, 0.3 and 0.5 for clustering pattern "M" is based on empirical results. Detailed investigation is needed, which is not the scope of this paper.

The experimenting is in four steps:

1) Data collection
   Just as depicted in Figure 1(b), we download specified number of search results and extract all N out-links, M in-links for all pages in search results. By stemming algorithm, we get all L distinct terms appeared in snippet and anchor text for all n pages of search results as well as the correspondent

frequency of the each term in each page.

2) Data cleaning
   This step is meant to remove duplicates or mirrors. Two pages *p* and *q* are said duplicate if (a) they each have at least 8 out-links and (b) they share at least 80% of their out-links in common. The page with higher common link percentage and its associated out-links, in-links and keywords will be removed and does not be clustered.
3) Clustering process.
   Applying the proposed algorithm to form base clusters and generate final clusters by recursively merging two base clusters if they share majority (75%) members.
4) Presenting clusters with tagging keywords
   Presenting web search results on the topic into clusters and attach each cluster with terms that having highest values in the centroid vector as the tagging keywords of the cluster.

4.1 Experimental Results

We tried topic "Jaguar" and "Chair", each with 200 URLs of search results for clustering. We download (also extract snippet and anchor text attached to each of) 200 search results returned from Yahoo for the topics, and for each page in search results, we extract its all out-links as well as 100 in-links. We also apply stemming algorithm to snippets and anchor text of all pages in search results to get distinct terms and correspondent appearance frequency in each page.

As final clustering results reveal, one page could belong to more than one cluster or belong to singleton cluster, which means that it cannot be grouped with others. In the rest of discussion, "pages/URLs clustered" means pages or URLs that appear in final clusters whose size is bigger than 3. The size of a cluster is the number of pages in the cluster. In the whole experimentation process, merging threshold 0.75 is used as recommended in [22]. We ignore singleton clusters or very small clusters.

It is in Table 1 that clusters distribution of final clustering results of topic "Jaguar" for different clustering patterns are depicted. We get impression from Table 1 that term-based clustering is very coarse. It could only identify the most popular ideas around the topic and is sensitive to the variation of similarity thresholds. Since snippets usually bring noises, the final quality depends heavily on how well the selection of final terms used to represent the vector. With low similarity threshold, most pages are clustered in more than one cluster, and in turn pages with different topics might be in the same cluster by merging process. As shown in Table 1, for term-based clustering, it generates several very big clusters and some very small clusters, which are not interpretable as in Table 2. When we say one cluster is "not interpretable", we mean that we cannot identify the main idea of the cluster, which might include pages with different topics. For C/0.1 (0.1 denotes similarity threshold) only subtopics "car", "club", "game" are identified, while L/0.1 could identify some medium but semantically meaningful groups. Since only depend on the link information, recall is some low and average entropy of the big –size clusters are not very good. Combining link and contents analysis in clustering could compensate the link-based clustering in increasing the recall greatly and making final clusters evenly distributed.

In Table 2, we check the quality of final clustering results of different clustering patterns from semantic point of view. We list all clusters whose size are bigger than 3 in descend order based on cluster size as well as main tagging keywords for each cluster. Each entry in Table 2 lists the terms with average weight bigger than 0.5 in parenthesis for the correspondent cluster, usually just with top one or two terms that most shared by the members of the cluster. We could see from Table 2 that term-based clustering is not able to effectively separate one page from another page in a very narrow way. E.g. for clustering pattern C/0.1, different subtopics pages are mixed as Cluster No.1 both on "car (1.7)" and "club (1.4)". In Table 2, ** is used to indicate the correspondent cluster is not interpretable, which due to the fact that its most shared terms are useless to identify the main topic like "click", "move" or the value is smaller than 0.5. While for L/0.1 in Table 2, it identifies some medium size but rather "pure", meaningful clusters, e.g. groups "magazine", "cat", "tour" (touring place)", "part" (information about parts of a car) etc. According to tagging words and its weight value, we also find that the main idea of these clusters is prominent. For M/0.1 in Table 2, it is clearly that it could "pull" some pages with the same topic but missing sharable links into the cluster. Of course, it also brings some "noise", which represented by the cluster with keywords "frame" but actually unable to interpret its meaning.

Table 3 is the summary of clustering results for topic "chair", which is rather general. In addition to "furniture", "university/ department chair", some interesting groups like wheelchair, rock chair for outdoor event are also identified. The contribution of link and terms in clustering process are also hold in this case.

One phenomenon observed is that some small clusters produced by the proposed approach are semantically very similar and should be in one group from a more general point of view. We think this could be solved by introducing hierarchical clustering to make the clustering results more natural and easy to interpret.

4.2 Evaluations

Validating clustering algorithm and evaluating its quality is complex because it is difficult to find an objective measure of quality of clusters. We would like to use three metrics *precisions***,** *recall* and *entropy* to evaluate quality of final clusters. We manually check 200 web pages for the topic and mark each one page with "relevant" or "irrelevant" to indicate whether it is relevant to the corresponding query topic. Precision and recall are defined as follows:

Precision=number of URLs that are both clustered and 'relevant' marked / number of URLs clustered (6)

Recall=number of URLs that are both clustered and 'relevant' marked /number of 'relevant' marked URLs (7)

Entropy provides a measure of "goodness" or "purity" for un-nested clusters by comparing the groups produced by the clustering technique to known classes. Small entropy value of the cluster indicates its high intra-cohesiveness while big entropy value means that its

Pattern/ Similarity Thresholds	Number of Very Big Clusters (size> 30)	Number of Medium clusters (15< size< 30)	Number of Medium Clusters (8<size<15)	Number of Small Clusters (3< size<8)	Number of Singleton
C/ 0.1	3	0	0	6	37
C /0.15	2	1	1	4	50
C/0.2	2	1	0	4	67
L / 0.1	1	1	2	5	73
M/0.1	1	2	4	5	46

Table 1. Clusters distribution for different clustering patterns for topic "jaguar"

Main Keywords (Cluster Size>3)	C/ 0.1	C/0.15	C/0.2	L/0.1	M/0.1
1	Car (1.7), Club (1.4)	Car (1.8), Club (1.2)	Car (1.9), Club (1.2)	Car (1.6), Club (0.8)	Car (2.2), club (0.5)
2	Club, (1.7) Car (1.2)	Club (1.9), part (1.1), Car (0.7)	Club (1.6)	Club (1.5), frame (0.6)	Club (1.7)
3	Game (1.4), Atari (1.1)	Game (1.7)	Game (1.2)	Game (1.7)	Game (2)
4	**	**	**	Magazine (1.4)	Atari Emulate (1.9)
5	**	**	Cat (1.4)	Cat (1.6)	Cat (1.8), onca (1.5)
6	**	Magazine (1.3)	**	Atari Emulate (1.5)	Magazine (1.5)
7	**	**	**	Part (1.1), type (0.7)	Part (1.2), Type (0.9)
8	**	**		Race (0.9)	Race (0.8)
9	**			Tour (1.1)	Tour (1.1)
10					Link (1.2)
11					Frame** (0.9)
12					Support (1)

Table 2. Main tagging keywords for different clustering patterns for topic "jaguar"

Main Keywords (Cluster Size>3)	C/0.1	L/0.1	M/0.1
1	Company, furniture	Company, furniture	Company, furniture, office
2	University, department, engine	University, department	Rock, shop, outdoor
3	**	**	University, department
4	**	Rock, shop,	Electronic, history
5	**	Ergonomic, seat	Ergonomic, seat
6	**	Massage	Massage, service
7	Wheelchair, evacuate, stair	Wheelchair, evacuate	Wheelchair, evacuate
8	**	**	Toyota, armchair

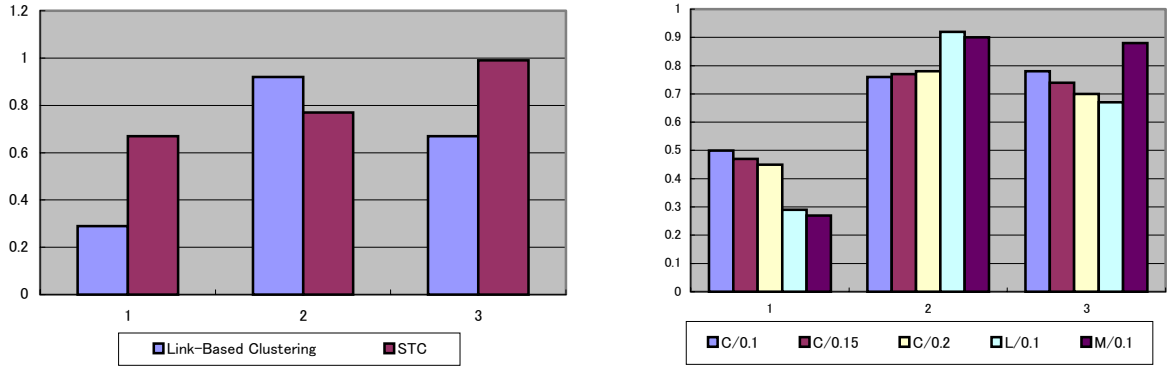
Table 3 Main tagging keywords for different clustering patterns for topic "chair"

members are not tightly related and focus on different sub-topics under the same general topic. In our initiative evaluation, we manually check each page that joins the clustering procedure and then give our judgment. Each page is given two estimates: relevant (to the query topic), its main topic and then create *classes* manually. Although it is time-consuming and it could lead to bias in our evaluation, we plan to carry out user experiment to counteract potential bias. We adopt the computing of entropy introduced in [9]: Let CS be a cluster solution and E (j)= $-\sum_i p_{ij} \log(p_{ij})$ is the entropy for each cluster j. p_{ij} is used to compute the "probability" that a member of cluster j belongs to the given class i. The average entropy for a set of clusters is calculated as the sum of entropy of each cluster weighted by its size: $E_{CS} = \sum_{j=1}^{m} \frac{n_j * E(j)}{n}$, where n_j is the size of cluster j, m is the number of clusters and n is the total number of data points. In Figure 2 (a) and (b), the number 1, 2, 3 in x-axles are average entropy, precision and recall respectively. In order to get in-depth understanding of the proposed approach, we conducted detailed evaluations on topic "jaguar" by comparing the clustering results of proposed approach with other clustering algorithms as well as the comparison among different clustering patterns.

4.2.1 Link-based Clustering verse STC (Suffix Tree Clustering)

We implemented STC algorithm proposed in [7] (also see explanation in Section 2.1) that is based on snippet attached with each URL in search results.

We first compared the final clustering results produced by link-based approach with snippet-based STC algorithm on the three metrics. We choose 0.1 and 0.75 as similarity and merge threshold respectively in links-based clustering. The results are depicted in Figure 2(a). According to the final clustering result produced by STC algorithm, it just discerns the main group, which usually includes most pages in dataset as well as several very small groups that include just 3 or 4 pages. It fails to identify some medium, but meaningful groups around the main topic. This result lead to very high entropy, as depicted in Figure 2(a), which means the main group is not so "pure" and some pages in it should be separated and grouped into more cohesive clusters. This could explain the high precision value for both link-based approach and snippet-based approach shown in Figure 2(a). As for recall, clustering results produced by STC are of high recall value since most URLs in search results are clustered and most of URLs clustered are in one cluster. We think that one possible reason that STC algorithm works poor under our experimenting environment might be that the query topics for testing are quite general (one word or two terms) while the advantage of STC is to capture the relationship and order of the terms appeared as keywords.



(a) **(b)**

Figure 2. (a) Comparison of STC with Link based clustering pattern and (b) Comparing of different clustering patterns for topic "jaguar" based on average entropy, precision and recall

4.2.2 Evaluation Among Different Patterns

We also evaluate the quality of clustering results for different clustering patterns by three metrics, as depicted

in figure 2(b). In general, the average entropy for term-based clustering is rather high, which means that the clusters obtained by this way are very coarse, pages in one cluster actually covers different subtopics. Link-based clustering could improve some for this but with low recall since it could identify some medium but tightly related, meaningful clusters. Combining link and contents will improve without sacrificing "purity" but at a little cost of precision, which is clearly conveyed in Figure 2(b) since snippets usually bring noises. Since clustering web search results is meant to give classified information to facilitate user's locating and accordingly information interpretation, combining link and contents are promising and in general works much better than current term-based clustering and link-based clustering approach.

5. Conclusion

In this paper, we extend the previous work on link-based clustering by combine links as well as contents appeared in snippets as well as anchor text in clustering process. Snippet and anchor text are considered to give a brief summarization for the topic of the page under consideration. Our goal is to cluster high quality pages (by filtering some irrelevant pages) in search results returned from web search engine for a specific query topic into semantically meaningful groups with useful tagging keywords to facilitate users' locating and information interpretation. We also extend standard K-means algorithm to overcome its disadvantages to make it more natural to handle noises. In order to investigate effectiveness of the proposed approach, we carry out experiments on query topic "Jaguar" and "chair" for different clustering patterns, especially we conducted detailed evaluations on topics "jaguar" to get a depth understanding by comparison with STC algorithm as well as among different clustering patterns based on three metrics: average entropy, precision and recall. Experimental results suggested term-based clustering is too coarse and suitable for clustering a corpus of text documents that cover a wide range of topics instead of web search results. Link-based clustering could identify tightly related, meaningful groups. However, low recall and high entropy for big-size cluster are its disadvantages. Moreover, it is difficult to tag each cluster automatically. Combining contents and links is a natural extension and solve the mentioned problems. However, since contents will bring some noises, well-extracted keywords are important. Combining links and contents could generate reasonable clustering results.

We would like to continually extend our current work by introducing hierarchical clustering on final clustering results to make it easier to interpret and some heuristic rules to remove noise links/words to improve final clusters quality.

Reference:

1. **Kleinberg 98** Jon Kleinberg. *Authoritative sources in a hyperlinked environment.* In proceedings of the 9th ACM-SIAM Symposium on Discrete Algorithms (SODA), January 1998.
2. **Ravi Kumar *et. al.* 99** Trawling the Web for emerging cyber-communities In Proceedings of 8th WWW conference, 1999, Toronto, Canada.
3. **Brin and Page 98** Sergey Brin, and Larry Page. *The anatomy of a large scale hypertextual web search engine*. In Proceedings of WWW7, Brisbane, Australia, April 1998.
4. **Oren Zamir and Oren Etzioni 99** *Grouper: A Dynamic Clustering Interface to Web Search Results* In Proceedings of 8th WWW Conference, Toronto Canada.
5. **Richard C. Dubes and Anil K.Jain,** *Algorithms for Clustering Data***, Prentice Hall, 1988**
6. **Oren Zamir and Oren Etzioni 97** *Fast and Intuitive clustering of Web documents*, KDD'97, pp287-290
7. **Oren Zamir and Oren Etzioni 98** *Web document clustering: A feasibility demonstration* In Proceedings of SIGIR' 98 Melbourne, Australia.
8. **Zhihua Jiang *et. al.*** *Retriever: Improving Web Search Engine Results Using Clustering*
9. **Ron Weiss *et. al.* 96** *Hypursuit: A Hierarchical Network Search Engine that Exploits Content-Link Hypertext Clustering* Hypertext'96 Washington USA
10. **Michael Steinbach, George karypis and Vipin Kumar** *A Comparison of Document Clustering techniques* KDD'2000. Technical report of University of Minnesota**.**
11. **James Pitkow and Peter Pirolli 97** *Life, Death and lawfulness on the Electronic Frontier*. In proceedings of ACM SIGCHI Conference on Human Factors in computing, 1997
12. **Cutting, D.R. *et. al.92*** *Scatter/gather: A Cluster-based approach to browsing large document collections*. In Proceedings of the 15th ACM SIGIR Conference on Research and Development in Information Retrieval. pp 318-329; 1992
13. **A.V. Leouski and W.B. Croft. 96** *An evaluation of techniques for clustering search results.* Technical Report IR-76 Department of Computer Science, University of Massachusetts, Amherst, 1996
14. **Broder *et. al.* 97** *Syntactic clustering of the Web.* In proceedings of the Sixth International World Wide Web Conference, April 1997, pages 391-404.
15. **Bharat and Henzinger 98** Krishna Bharat, and Monika Henzinger. *Improved algorithms for topic distillation in hyperlinked environments*. In Proceedings of the 21st SIGIR conference, Melbourne, Australia, 1998.
16. **Chakrabarti *et. al.* 98** Soumen Chakrabarti, Byron Dom, David Gibson, Jon Kleinberg, Prabhakar Raghavan, and

Sridhar Rajagopalan. *Automatic Resource Compilation by Analyzing Hyperlink Structure and Associated Text*. Proceedings of the 7th World-Wide Web conference, 1998.

17. **Florescu, Levy and Mendelzon 98** Daniela Florescu, Alon Levy, Alberto Mendelzon. *Database Techniques for the World-Wide Web: A Survey.* SIGMOD Record 27(3): 59-74 (1998).
18. **Gibson, Kleinberg and Raghavan 98** David Gibson, Jon Kleinberg, Prabhakar Raghavan. *Inferring Web communities from link topology.* Proc. 9th ACM Conference on Hypertext and Hypermedia, 1998.
19. **Agrawal and Srikant 94** Rakesh Agrawal and Ramakrishnan Srikanth. *Fast Algorithms for mining Association rules*, In Proceedings of VLDB, Sept 1994, Santiago, Chile.
20. **M.M. Kessler,** *Bibliographic coupling between scientific papers* , American Documentation, 14(1963), pp 10-25
21. **H. Small,** *Co-citation in the scientific literature: A new measure of the relationship between two documents*, J. American Soc. Info. Sci., 24(1973), pp 265-269
22. **Yitong Wang and Masaru Kitsuregawa,** *Use Link-based clustering to improve web search results*, WISE'01, 2001