ELSEVIER

Information Processing and Management 40 (2004) 239–255

INFORMATION PROCESSING & MANAGEMENT

www.elsevier.com/locate/infoproman

Automatic topics discovery from hyperlinked documents

Kuo-Jui Wu [a,*], Meng-Chang Chen [a,1], Yeali Sun [b,2]

[a] *Institute of Information Science, Academia Sinica, 128, Section 2, Academic Road, Nankang 115, Taipei, Taiwan*

[b] *Department of Information Management, National Taiwan University, No. 50, Lane 144, Kee-Lung Road, Section 4, Taipei, Taiwan*

Received 20 August 2002; accepted 29 April 2003

Abstract

Topic discovery is an important means for marketing, e-Business and social science studies. As well, it can be applied to various purposes, such as identifying a group with certain properties and observing the emergence and diminishment of a certain cyber community. Previous topic discovery work (J.M. Kleinberg, Proceedings of the 9th Annual ACM-SIAM Symposium on Discrete Algorithms, San Francisco, California, p. 668) requires manual judgment of usefulness of outcomes and is thus incapable of handling the explosive growth of the Internet. In this paper, we propose the Automatic Topic Discovery (ATD) method, which combines a method of base set construction, a clustering algorithm and an iterative principal eigenvector computation method to discover the topics relevant to a given query without using manual examination. Given a query, ATD returns with topics associated with the query and top representative pages for each topic. Our experiments show that the ATD method performs better than the traditional eigenvector method in terms of computation time and topic discovery quality.

Keywords: Topic discovery; Hyperlink analysis; Authority; Hub

1. Introduction

With the explosive growth of the World Wide Web, effective and efficient information discovery becomes more difficult. The emergence of large portals and search engines are designed to help

* Corresponding author. Tel.: +886-2-27883799x1614.

E-mail addresses: wugray@iss.sinica.edu.tw (K.-J. Wu), mcc@iis.sinica.edu.tw (M.-C. Chen), sunny@im.ntu.edu.tw (Y. Sun).

[1] Tel.: +886-2-27883799x1802.

[2] Tel.: +886-2-23630231x2870.

0306-4573/$ - see front matter
doi:10.1016/S0306-4573(03)00039-6

users meet their information needs. Directory services such as Yahoo! (http://www.yahoo.com/), are like the Yellow Pages of Internet documents (also called *web pages* interchangeably in this paper) collected and categorized manually. Documents collected in the directories are only part of existing Internet documents, and most collected pages are already well-known and popular. Therefore information discovery using directory services is limited to the services' collections.

Search engines, like AltaVista (http://www.altavista.com/), are another form of document collection service. Search engines use robots or spiders to crawl through the Internet and to collect and index the documents the robots/spiders encounter. When receiving queries (composed of keywords) from users, search engines scan their databases and present users with a list of matched documents sorted by similarity scores between page contents and queries. After receiving a long list of documents, users must browse the documents and after several clicks, they may become discouraged due to the amount of irrelevant results.

The problem with search engines is that they often return too many results, and document ranking may not meet the user's expectations (Chaffee & Gauch, 2000). In addition, many search engines only return a set of individual documents. Internet documents returned from a search engine may contain several topics about the input query. For example, the results of query "Jordan" may contain topics such as the famous NBA player "Michael Jordan", "The Formula 1 Jordan Grand Prix Team", "Jordan Middle School", and the Middle East country, "Jordan", etc. It is obvious that the documents from different topics represent different interests. For web document retrieval, it helps users to assimilate the results by partitioning the documents into topics and annotating each topic. Furthermore, it is sometimes important to rank the documents according to their importance to a specific topic. In some cases, a topic may represent a cyber community, such as the "Michael Jordan Fan Club", that is of great interest to many e-Business applications or social science studies. Periodical scanning of the Internet to discover new cyber communities has become a common practice for many WWW related applications.

Internet documents are associated via hyperlinks. When Internet document authors prepare documents, hyperlinks are added for reference, related pages, etc. Note that Internet documents link to related pages more often than unrelated pages (Davison, 2000). For example, if web page h_1 is about NBA basketball games and h_1 links to web page h_2, h_3, and h_4, then h_2, h_3, and h_4 are probably NBA related pages. While the links may have different intentions, cross-references among documents as a whole provide useful information about their underlying associations. Topic discovery is an emerging technology that can be applied to enhance search engine results. Kleinberg (1998) proposes the algorithm HITS (Hyperlink-Induced Topic Search) in which the major concept is authority endorsement and conferral by hyperlinks between web pages. Links between web pages contain latent human judgment when links are written into web pages. One application of the HITS algorithm is to re-rank the results returned from the search engine based on link information among web pages.

In order to find more specific topics within the main search concept, Kleinberg suggests calculating all eigenvectors of the hyperlink matrix. Other topics may be represented by non-principal eigenvectors. However, it cannot be determined automatically which non-principal eigenvector represents a meaningful topic. To correct this shortcoming, we propose an algorithm that partitions the web pages in search results into clusters. For each cluster we run a conventional principal eigenvector computation algorithm to find the representing vector. In this way, we can automatically discover meaningful topics for a given query, and rank each topic according to its authority.

The remainder of this paper is organized as follows. First, we discuss related literature in Section 2. In Section 3, we describe the ATD algorithm. Experiment setup and results are discussed in Section 4. We discuss conclusions and future work in Section 5.

2. Related work

In recent years, hyperlink analysis has become an emerging research issue for web document retrieval. Prior works (Chakravarthy & Haase, 1995; Luke, Spector, Rager, & Hendler, 1997) in this research field mainly focused on text-based and meta-level methods. There were also some interesting works on analyzing links in hypertexts (Botafogo, Eivlin, & Shneiderman, 1992; Ellis, Furner-Hines, & Willett, 1994). Hyperlinks can also be used to construct a basic indexing unit of related web documents (Géry, 2002), to represent semantic contents and structures. This indexing unit can help with retrieving and navigating information on the Web.

In 1998, Kleinberg proposed the HITS algorithm for query-dependent web page ranking. In the same year, Brin and Page proposed PageRank algorithm (Brin & Page, 1998) for query-independent web page ranking. These two algorithms use simple but very efficient models to exploit hyperlink topology underlying the web documents. This hints that hyperlinked relations are an effective means to rank web documents by importance. In Amento, Terveen, and Hill (2000), it is verified that authority score, PageRank value, and even the in-degree number of a page could be used to identify high quality web pages.

Kleinberg's algorithm has sparked great interest, and much related research has been conducted. Gibson and Kleinberg extended the experiments in Kleinberg (1998) and concluded in Gibson, Kleinberg, and Raghavan (1998) that the HITS algorithm is robust in that it can discover similar principal topic from different sources or different starting points. Furthermore, the HITS algorithm tends to generalize an original query into a broader topic because of the aggregate behavior of vast user populations. There has been many investigations and discussions about the ranking behavior of HITS and other link-based algorithms (Ding, He, Husbands, Zha, & Simon, 2002; Ng, Zheng, & Jordan, 2001). Borodin, Roberts, Rosenthal, and Tsaparas (2001) analyzed various hyperlink analysis algorithms: HITS, PageRank, SALSA (Lempel & Moran, 2000), PHITS (Cohn & Chang, 2000), their own modified version of HITS, and the Bayesian estimation algorithm. They concluded that different algorithms emerge as best for different queries, while there are some queries for which no algorithm seems to perform well.

Bharat and Henzinger improved the HITS algorithm by reducing the influence from web pages in a specific site (Bharat & Henzinger, 1998). In addition, they included a content similarity measure to prune irrelevant web pages. After purifying the dataset and regulating the influence of web pages, they tackled the problem of topic drift and raised precision in their definition. An alternative approach to solve the topic drift or contamination problem was proposed by Chakrabarti, Joshi, and Tawde (2001). It analyzes text, markup tags and hyperlinks to identify and extract microhub regions relevant to a query. Farahat et al. proposed a method to estimate authoritativeness of a document based on textual, no-topical cues (Farahat, Nunberg, & Chen, 2002). This method is complementary to hyperlink-based methods, and can be applied to combine textual authoritativeness with social authority. An interesting work of Meghabghab applied Kleinberg's web algorithm to different web graphs in the new coordinate space: out-degree and in-degree (Meghabghab, 2002).

Dean and Henzinger proposed an improved HITS-like algorithm (Dean & Henzinger, 1999) to find related web pages for a specific URL based on the hyperlinks and the order the hyperlinks appear in a web page. Chang, Cohn, and McCallum (2000) also varied the HITS algorithm to allow users to prepare their own lists of authorities which helps to align the ranking results in relation to users' opinions of authorities. By manipulating the weight of the link matrix in HITS, this algorithm can spread more influences from user granted high-authority web pages to neighbors. In the paper the authors regard the existence of secondary topics in the collected dataset as a bad influence on ranking results. Contrarily, we believe that by identifying those topics in the dataset and ranking them separately, we can discover more topics related to a single input query.

Kumar et al. propose to find a special structure, named *core*, to enumerate all communities (topics) in the web graph (Kumar, Raghavan, Rajagopalan, & Tomkins, 1999a, 1999b; Kumar et al., 2000). This is somewhat different from HITS-like algorithms because it is a topic-independent topic distillation algorithm. A *core* $c(i, j)$ is a complete directed bipartite structure in web graph which combines *i fan* vertices (specialized hubs) and *j center* vertices (specialized authorities). After cores are found from the web graph, each core is expanded to build a larger dataset. Web pages in each dataset are then ranked by mutually reinforcing relationship as defined in the HITS algorithm.

Davison et al. (1999) used a fast eigenvalue solver that converges quickly to find the eigenvectors of the link adjacency matrix. They indicate that by looking at the top 1/4 eigenvalues and the associated eigenvectors, they could find clusters of web pages that are more interesting than the one extracted by the principal eigenvector. However, they are still unable to automatically determine which eigenvalue and associated eigenvector can locate interesting topics. In the next section, we will illustrate how our algorithm can do this job automatically.

3. Topic discovery

We propose an automatic topic discovery algorithm for a user given query, called *ATD algorithm*. ATD algorithm is composed of a method of base set construction, a clustering algorithm and a principal eigenvector computation algorithm. The aim of the ATD algorithm is to identify and isolate each strongly inter-connected cluster as topic in the web vicinity graph, and then select top-ranked web pages within each cluster to be its representing concept. The task of automatic topic discovery is composed of five fundamental parts:

- *INPUT*: a broad-topic query.
- A method to build a web vicinity graph for the query using either a focused crawler or a search engine to provide web pages related to the input query that then employs hyperlink expansion to create a web vicinity graph.
- A clustering algorithm to partition the web graph into separate clusters.
- A ranking algorithm to rank web pages within each cluster.
- *OUTPUT*: Each cluster is regarded as a topic and top-ranked web pages within the cluster are presented to the user as the representation of the topic.

Each fundamental part will be explained in the following sections.

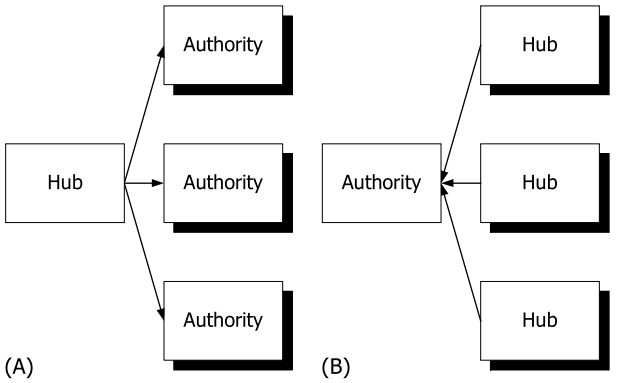

Fig. 1. Mutually reinforcing relationship.

3.1. Mutually reinforcing relationship

The mutually reinforcing relationship is the essential concept of hyperlink analysis in Kleinberg (1998). Kleinberg proposed that every web page has two properties: *authority* and *hub*. The authority attribute of a web page is the degree of representation and authority in relation to the input query topic. A web page's authority is promoted if linked by many pages. The hub attribute indicates the quality of the hyperlinks which link to high quality authority pages contained in the web page content. Thus, "a good *hub* is a page that points to many good authorities; a good *authority* is a page that is pointed to by many good hubs" (Kleinberg, 1998), as shown in Fig. 1. This is the *mutually reinforcing relationship*. In topic discovery, a partitioned cluster that contains a topic is composed of good authority and hub pages. As to the ranking in a topic, the work of Amento et al. (2000) has shown that hyperlink can be used to rank web pages in terms of authority and representation. Thus, we rank web pages using their links in each discovered topic.

3.2. Vicinity graph construction

To exploit the above concept to discover topics, we first have to prepare a collection of web pages (called a *vicinity graph*). There are several criteria for a vicinity graph. First, a vicinity graph should be broad enough to cover underlying topics. Second, the pages (nodes) in the vicinity graph should be coherent otherwise topic discovery results will be poor. A good starting point for constructing a vicinity graph is to collect web pages containing keywords from the query. A focused crawler or a content-based search engine can help provide such a collection of web pages for an input query. The HITS algorithm exploits an existing content-based search engine to provide related web pages of an input broad-topic query. It then fetches these web pages and employs only the hyperlinks in the pages. HITS has a pre-processing step to construct a base set (vicinity graph). First, the top 200 web pages of the search results are chosen to form a root set. Note that the amount of web pages (200) is a tunable parameter. Next, HITS expands the root set into a base set according to link information by the following steps:

(i) Pages that are linked by any pages in the root set are added into the base set.
(ii) Web pages which link to any pages in the root set, are added into the base set. If there are more than 50 pages that link to a single page in the root set, only 50 randomly chosen pages are added. This amount (50) is also a tunable parameter.
(iii) The root set is added into the base set. Fig. 2 shows the base set construction process.

3.3. Clustering—the authority–hub–authority (A–H–A) algorithm

Here we illustrate the authority–hub–authority (A–H–A) clustering algorithm to partition the vicinity graph into individual clusters. The A–H–A cluster algorithm finds clusters by following the underlying link topology of web pages in the vicinity graph. To some extent, a web page with large out-degree can be regarded as a hub candidate. Similarly, a web page with large in-degree can be regarded as an authority candidate.

1. Let $k = 1$.
2. Repeat following steps until no more clusters are found.
3. Step C:
Let $Cluster_k$ = NULL.
For the node O with maximum out-degree in the base set, find the node C with maximum in-degree linked by O.
Node C is added into $Cluster_k$ and removed it from the vicinity graph.
4. Step H:
Add nodes that link to C into $Cluster_k$ and remove them from the vicinity graph.
5. Step A:
Add nodes linked by nodes added in step H into $Cluster_k$ and remove from the vicinity graph.
6. Let $k = k+1$.

Fig. 3 shows one iteration of the A–H–A algorithm. The first step of the A–H–A clustering algorithm (Step C) finds the node C, which is considered as a centroid authority of the topic. This centroid can trawl in the hubs of the topic in the second step (Step H) because good hubs link to good authorities. In order to form a complete topic, the A–H–A clustering algorithm trawls in other authorities in the third step (Step A). This clustering algorithm therefore conforms to the relationship between authority and hub. Empirically, multiple levels of AHA algorithms, i.e. multiple iterations of steps 4 and 5 in the above algorithm, produce very large and topic-less clusters. A cluster is discarded if the number of nodes in that cluster is less than a pre-defined threshold value. After the vicinity graph is partitioned into several clusters, web pages are ranked in each cluster and these topics and web pages are presented in the ranking order to the user.

In order to remove noise in the vicinity graph, the A–H–A clustering algorithm discards hyperlinks with the following properties during the clustering process:

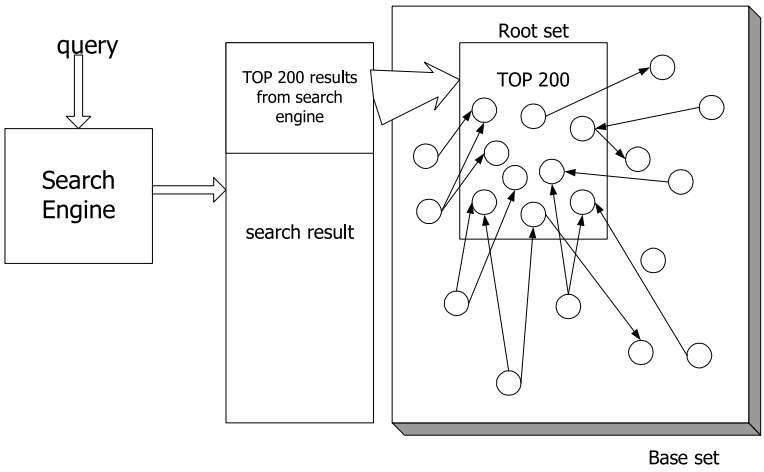

Fig. 2. Vicinity graph construction of HITS.

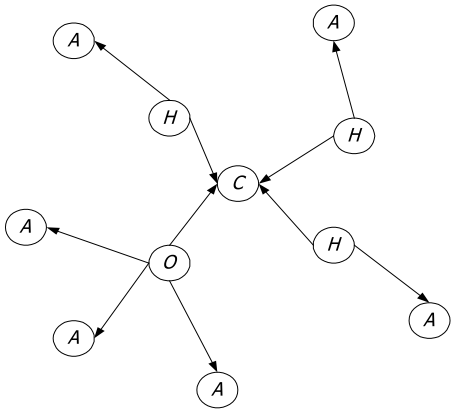

Fig. 3. A–H–A clustering.

- *The source and destination are in the same domain*. Many of this kind of hyperlinks serve only navigational purposes, such as "back" and "click here to go to the main page". These hyperlinks have no contribution to the topic. In addition, a good topic should be formed by web pages from many different authors and sites. A good topic can attract great interest from many different people. If hyperlinks from the same domain are allowed, pseudo-topics may be created and page ranking will favor these self-linked pages.
- *The destination of a hyperlink is a large portal site, such as Yahoo! or other popular, general-purpose sites.* Destination pages are irrelevant to almost any topic, but many web pages contain these links. Moreover, for the same reason, URLs with destinations such as *host.domain/**copyright.html*** or *host.domain/**privacy.html***, etc., should also be removed. We use an URL stop-list to remove these hyperlinks.

3.4. Topic ranking algorithm

For each cluster, a link adjacency matrix **A** is built to run the ranking algorithm. The link adjacency matrix **A** is an $n * n$ matrix. The web pages within a topic are numbered from 1 to n to build the matrix where n is the size of that cluster. If web page i links to web page j, then (i, j) element of the matrix **A** is set to 1, otherwise, (i, j) is set to 0. For example, suppose there are three web pages h_1, h_2, and h_3, in a cluster. If web page h_1 links to h_2 and h_3, h_2 links to h_3, and h_3 links to h_1, then the adjacency matrix **A** is built as

$$A = \begin{bmatrix} 0 & 1 & 1 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \end{bmatrix}.$$

Following, we describe the iterative version of the HITS algorithm. It shows how to calculate authority and hub score of web pages within a cluster.

1. The initial value of authority and hub score for each page is set to 1.
2. For each page, the new authority score is the sum of the hub scores of the web pages that link to it (as shown in Fig. 1B).

$$Auth_i = \sum_{j:(j,i)\in E} Hub_j \tag{1}$$

   where E is the set of all hyperlinks in a cluster.
3. For each page, the new hub score is the sum of the authority scores of the web pages that it links (as shown in Fig. 1A).

$$Hub_i = \sum_{j:(i,j)\in E} Auth_j \tag{2}$$

4. Normalize the authority and hub scores of the web pages.
5. Repeat step 2 to 4 until the scores are converged or a pre-defined number of iteration is reached.

Kleinberg also proposed a linear algebra version of the HITS ranking algorithm to calculate authority and hub scores. Step 2 and step 3 of the HITS algorithm can be written as $\mathbf{Auth} = \mathbf{A}^{\mathrm{T}} * \mathbf{Hub}$ and $\mathbf{Hub} = \mathbf{A} * \mathbf{Auth}$. The authority scores of each web page (i.e. **Auth**) correspond to the principal eigenvector associated with the largest eigenvalue of matrix $\mathbf{A}^{\mathrm{T}}\mathbf{A}$, and the hub scores (i.e. **Hub**) correspond to the principal eigenvector of matrix $\mathbf{A}\mathbf{A}^{\mathrm{T}}$, where **A** is the link adjacency matrix of the cluster. From linear algebra, if λ is a eigenvalue of an n^*n matrix **A**, and **x** is a non-zero vector such that $\mathbf{Ax} = \lambda\mathbf{x}$, then λ is an eigenvalue and **x** is an eigenvector. In other words, eigenvectors become multiples of themselves when transformed by matrix **A**. Matrix **A** has k distinct eigenvalues if the characteristic polynomial $\det(\mathbf{A} - \lambda\mathbf{I}) = 0$ has k roots. The largest eigenvalue of matrix **A** is the principal eigenvalue and the corresponding eigenvector is the principal eigenvector.

After calculating authority and hub scores of the web pages of each cluster, HITS ranks the pages according to authority scores and presents the top N results to the user. The ranking is ordered by the topic's authority value in relation to the input query.

The original HITS algorithm runs through the entire vicinity graph, and discovers topics with the strongest connected web pages. Kleinberg points out that it may include additional topics by calculating non-principal eigenvectors of matrix $\mathbf{A}^{\mathrm{T}}\mathbf{A}$ or $\mathbf{A}\mathbf{A}^{\mathrm{T}}$ ($\mathbf{A}$ is the adjacency matrix of the entire, original vicinity graph). While all the eigenvectors can be obtained by using an eigenvector computing algorithm, it is not clear which non-principal eigenvector is associated with a meaningful topic.

In order to measure the quality of an eigenvector as a topic, we define a *Topic Goodness Metric* (*TGM*). With each eigenvector $\mathbf{x}$, let $TGM(\mathbf{x})$ be defined as

$$TGM(x) = \left| \underset{i \in TA_x}{Auth_x(i)} \right| + \left| \underset{i \in TH_x}{Hub_x(i)} \right|, \tag{3}$$

where web page $i \in TA_x$ is a top k authority of eigenvector $\mathbf{x}$, and $Auth_x(i)$ is the associated authority value (similarly, web page $i \in TH_x$ is a top k hub and $Hub_x(i)$ is its hub score). The reason for TGM is that authorities and hubs associated with eigenvectors with larger absolute values will typically be densely linked in the vicinity graph, and will probably have a more concrete relationship to the query. Using an eigenvector computation algorithm such as GNU Scientific Library, we can find the principal eigenvector and other non-principal eigenvectors. By setting a threshold of minimal TGM value, we may find several interesting topics associated with non-principal eigenvectors and discard the non-prominent ones. For simplicity, we call this method (using TGM value to rank topics associated with eigenvectors) the *ECT* (eigenvector calculation with TGM) *method.*

3.5. Automatic topic discovery (ATD) algorithm

The complete workflow of automatic topic discovery is illustrated in Fig. 4. First a broad-topic query is sent to a content-based search engine. The content-based search engine searches its index against the query and returns a list of matched documents. Based on the search results, page contents are examined to obtain link information. Then, an expanded focused web vicinity graph, composed of relevant web pages and hyperlinks, is built. By following the underlying hyperlink

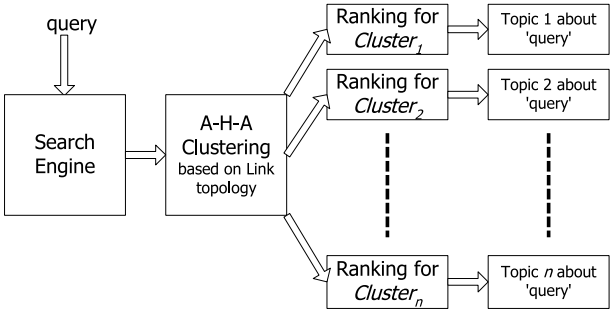

Fig. 4. Automatic topic discovery.

topology and relationship between authority and hub, the vicinity graph is partitioned into several inter-connected clusters relevant to the broad-topic query. These clusters are ranked by an eigenvector computing algorithm and become the topics retrieved by our algorithm.

4. Experiments and discussions

4.1. Topic discovery experiment

In experiments we submit queries to the AltaVista search engine as the content-based search engine. The base set is built with the following parameters: the top 200 search results are placed into the root set. When expanding the root set into the base set, we let each page in the root set trawl in at most 50 web pages. While running the A–H–A clustering algorithm of ATD, the minimal cluster size is set to 30 web pages. As for the *ECT*, we include the top 10 eigenvectors (note non-principal eigenvectors may contain two opposing concepts of a topic, such as pro-gun and anti-gun in the gun control issue, placed in positive and negative ends of the eigenvector space), and measure only the top 20 authorities and hubs ($k = 20$) of each eigenvector (potential topic). The minimal TGM threshold is set at 5 to discard unlikely topics. Each discovered topic is manually annotated with an appropriate label. In the following sections we show two experiments using *ECT* and *ATD* for the same base set of the query "Jaguar".

4.1.1. Experiment: ECT method

In this experiment, we use an Intel Pentium III 550 MHz PC running FreeBSD, and the gsl-0.7 library to calculate the eigenvectors of the link adjacency matrix. Before running the ECT method, the hyperlinks in web pages in the base set are checked against the URL stop-list. Also, hyperlinks that link to or are linked from the same domain are removed, as are their associated web pages. The computation time in this experiment is around 20 min. Note that the number of web pages in a topic is at most 40 (20 authority pages and 20 hub pages, but some of them may be the same pages). We examined the retrieved topics ranked by TGM value and found a topic with a smaller TGM value may be a subset of a topic with a higher TGM value.

In this experiment we show that TGM value is helpful in determining which eigenvector indicates a meaningful topic. By given query "Jaguar", Table 1 shows the two discovered topics with the largest TGM values: "Jaguar car" and "Atari Jaguar games". The two topics are different in their members and meanings, although they are mixed within the search engine result for the same query.

4.1.2. Experiment: ATD algorithm

In this experiment we use an Intel Pentium III 550 MHz PC with Windows 2000 to run the ATD algorithm implemented by Microsoft Visual C++. Unqualified web pages in the vicinity graph are removed as described in Section 4.1.1. Topic discovery results are showed in Table 2. It is worth noting the computation time to retrieve these three topics is less than 10 s. There are three distinct topics: "Jaguar cars", "NFL Jaguar team" and "Atari Jaguar games". Their sizes are 65, 39, and 34 web pages, respectively. These three topics are totally independent, and the only thing they have in common that they contain the query keyword "Jaguar" in their page contents. By

Table 1
Topic discovery result of query "Jaguar", ECT method

Topic_ID	TGM value	Top hub/authorities
Jaguar cars From 2nd non-principal eigenvector, positive end	6.03921	TOP 1 Hub http://www.webfocus.co.nz/jaguar/ TOP 1 Auth http://www.jaguarmagazine.com/ TOP 2 Auth http://www.collection.co.uk/ TOP 3 Auth http://www.jec.org.uk/
Atari Jaguar games From principal eigenvector	5.13644	TOP 1 Hub http://atarihq.com/interactive/ TOP 1 Auth http://jaguar.holyoak.com/ TOP 2 Auth http://songbird.atari.net/ TOP 3 Auth http://members.aol.com/atarijag/

Table 2
Topic discovery result of query "Jaguar", ATD method

Topic_ID	Top hub/authorities
Jaguar cars	TOP 1 Hub http://www.xks.com/ TOP 1 Auth http://www.jaguarcars.com/ TOP 2 Auth http://www.jagweb.com/ TOP 3 Auth http://www.classicjaguar.com/
NFL Jaguar team	TOP 1 Hub http://www.macjag.com/ TOP 1 Auth http://www.footballfanatics.com/football.taf?partner_id = 9 TOP 2 Auth http://www.nflfans.net/afccentral/jaguars/ TOP 3 Auth http://jaguars.jacksonville.com/
Atari Jaguar games	TOP 1 Hub http://atarihq.com/ TOP 1 Auth http://jaguar.holyoak.com/ TOP 2 Auth http://www.telegames.com/ TOP 3 Auth http://songbird.atari.net/

using A–H–A clustering algorithm which conforms to mutually reinforcing relationship, we observe that ATD algorithm is capable of discovering more topics than the ECT method.

4.1.3. Comparison

Following we evaluate topic relevance for ATD and ECT. We recruited a group of three volunteers to rate the topics. The rating criteria includes three choices: "**R**" stands for relevance, "**N**" stands for non-relevance, and "**S**" stands for relevance but with similarity to a topic previously rated "**R**". The ECT method tends to discover many topics of various sizes where some topics with smaller TGM value may be a subset of topics with larger TGM values. This type of topic is marked "**S**". Ratings are determined by the consensus of the three volunteers. As there are many different aspects or meanings of broad-topic queries, the volunteers were told to rate topics as relevant according to their own definitions of the relevance. We set the minimal threshold of TGM to 4.0 to obtain more topics than experiment in Section 4.1.1 for evaluation. Experiment results are shown in Table 3.

Table 3
Topic relevance comparison of ECT and ATD methods

Query	Algorithm		Topic_ID				
			1	2	3	4	5
Jaguar	ATD	Relevance	R	R	R	/	/
		Topic size	65	39	34	/	/
	ECT	Relevance	R	R	S	N	/
		TGM value	6.04	5.14	4.78	4.63	/
Japan	ATD	Relevance	R	R	R	R	/
		Topic size	51	35	37	35	/
	ECT	Relevance	R	N	R	R	S
		TGM value	5.17	4.85	4.80	4.39	4.05

While the ECT method can be used to discover topics (densely linked groups in the base set), some densely linked groups are irrelevant to a query or are a subset of a larger group. By using the A–H–A clustering algorithm and a minimal topic size threshold (30 in this experiment), those irrelevant topics will not be enumerated. Unlike the ECT method, the topics discovered by the ATD algorithm are distinct topics. In addition, the computation of the ATD algorithm is 100 times faster than the ECT method as ATD operates on small-sized clusters rather than the entire base set.

4.2. Performance evaluation of ATD and ECT

The experiment is designed to compare the performance of ATD and ECT in terms of metrics, such as precision and recall. We recruited three volunteers to manually identify topics from the base set as a benchmark. We picked five queries for this experiment: ''Japan'', ''basketball'', ''Jordan'', ''Movie'' and ''Jaguar''. As the vicinity graphs built for each query are more than 5000 pages, experts only investigated and identified topics from a randomly selected 25% of web pages in each vicinity graph. Majority votes were used to build a benchmark of identified topics for each query. We then ran the ATD and ECT methods on the vicinity graph of the five queries to automatically identify topics. By comparing retrieved topics to benchmarks, the number of topics, precision, and recall can be obtained to show the effectiveness of ATD and ECT, shown in Table 4. Note that when we calculate precision and recall, we have to judge whether two topics (one from the benchmark and the other is obtained from automatic methods) are the same. The judgment is that two topics are the same if more than 50% of the URLs from the retrieved topic are identical to the URLs of the benchmark topic. In Table 4, the category P@3 (precision at three) is the ratio of the expert defined topics over the top three automatic discovered topics.

4.2.1. Expert identified topics

In Table 4, we see that the number of expert identified topics is more than the automatically discovered topics by both ATD and ECT methods. This is because some expert identified topics are linked together by hyperlinks, and these topics are subsumed by a more general topic. As a

Table 4
Precision/recall evaluation of ECT and ATD methods

Comparison metric	Query_ID				
	Japan	Jordan	Jaguar	Basketball	Movie
Expert identified topics	28	14	14	17	18
ATD topics	4	4	3	4	5
ECT topics	5	4	4	5	6
ATD P@3	1.0	1.0	1.0	1.0	1.0
ECT P@3	0.67	0.34	0.67	0.67	1.0
ATD recall	0.14	0.21	0.21	0.24	0.28
ECT recall	0.07	0.07	0.14	0.12	0.17
ATD run time (s)	9	6	4	8	11
ECT run time (min)	23	15	11	18	28

result ATD and ECT retrieved these delicate topics as a single topic. For example, the topic "Research Institute" and "Universities" are two topics identified by experts of the query "Japan", but they are included in the topic "Colleges and Universities" retrieved by the ATD algorithm. As well, expert topic identification criteria are diverse and inconsistent, consequently some identified topics are close related which can be put together.

4.2.2. Automatic topic discovery algorithm

Compared to expert results, the ATD algorithm has a good performance in P@3, and a poor recall rate. Its poor recall is due to the fact that many small expert topics are linked by hyperlinks to form a bigger and more general topic. Therefore the ATD inherently enumerates fewer topics than experts. Furthermore, if there are only a few hyperlinks among the web pages of an expert identified topic, the ATD algorithm cannot decipher that topic either.

In contrast to the ATD algorithm, the ECT method often finds smaller topics that are regarded as subset of a more general topic. Hence, the smaller ones are not counted. Therefore the ATD algorithm has better performance than ECT, both in P@3 and recall. As for execution time, the ATD algorithm runs one hundred times faster.

4.3. Discussions

Topic discovery works well for retrieving interesting patterns or groups associated with query. In the experiments, the query contains only one keyword. In real world applications, a query can contain many keywords in order to accurately describe the user's interest. Some problems have been observed during our study:

- *Commercialization of the Internet*: In our experiments we found that the advertisement links had a negative effect on topic discovery and ranking accuracy. These commercial pages were added into the base set because they were linked by the top 200 results returned from the search engine. A solution to the problem is to use a URL stop-list to remove these commercial pages.
- *TKC (Tightly-Knit Community) effect*: The other issue is that some companies build many web pages that link to each other. These pages may have different domain names in their URLs.

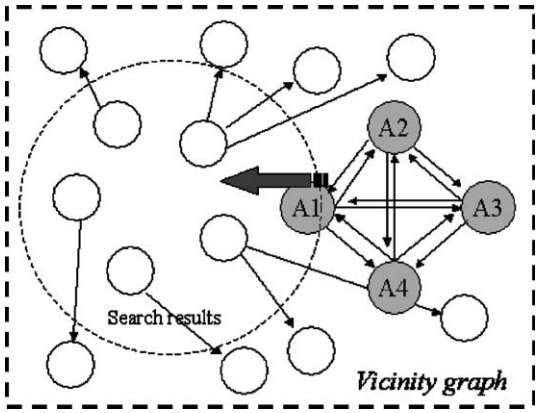

Fig. 5. TKC effect.

Once one of these pages is returned in the search results (e.g. web page A1), it will trawl other linked pages (A2, A3, and A4) into the base set as shown in Fig. 5. As a result, these web pages will manifest as an important topic, but are actually almost irrelevant to the query. This is another kind of Internet commercialization, and is known as the *TKC effect* (Lempel & Moran, 2000) and *Clique attack* (Chakrabarti et al., 2001). Since these web pages have different domain names, detection of this phenomenon by the URL checking is unlikely. One solution to this problem is to calculate the domain name distribution of web pages in the topics. If a topic is composed of web pages whose domain names belong to only a few different hosts, it is probable that it is a tightly knit topic.

- *Mirrored pages*: Some web pages are produced from copying other pages' contents as well as hyperlinks. This causes the authority scores of their target pages to be boosted. This is a common occurrence on the Internet that needs to be addressed. We solve this problem by comparing the hyperlinks of each pair of web pages. If over 80% of the links in two web pages are the same, they are regarded as mirrored pages. We keep one copy and remove the others.
- *High precision and low recall of automatic topic discovery*: Compared with experts results, the ATD algorithm generally has high precision at the top three rate and low recall rate. This situation conforms to the general opinion regarding topics which is that most people will agree on wide-spread and well-known topics and often have different opinions regarding minor topics which usually have only a few web pages and hyperlinks between web pages.
- *Annotation of topics*: While the algorithm can discover topics from Internet documents, it is difficult to annotate these topics automatically in presenting them to the user. We have observed that titles of the top hub pages can be used as topic annotation.
- *Minimal size of a topic*: If we set the threshold of minimal topic size to a smaller value, we can discover more topics (as shown in Fig. 6 for the query term 'Jordan'), but there will be two shortcomings. One is that some of the discovered topics are trivial topics. The other is that some topics are unstructured. Both of these types of topics are invaluable to users. If we choose a larger threshold, some important topics are likely to be omitted. From repeated experiments, this threshold is 30.

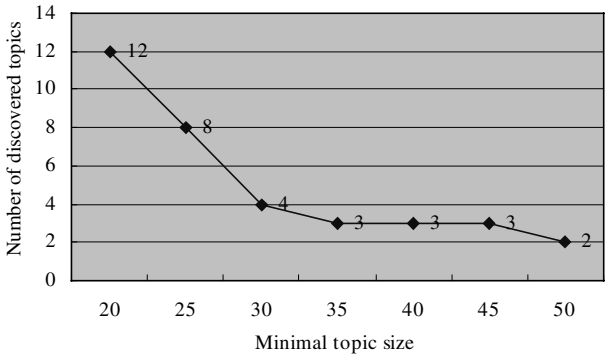

Fig. 6. Number of retrieved topics vs. minimal topic size for the query "Jordan".

5. Conclusions

In this work we propose an automatic topic discovery algorithm to discover multiple topics included in user query search results using conventional search engines. Unlike other topic discovery algorithms, our method can automatically enumerate these topics without human manipulation. The clustering algorithm presented in this work is based on underlying hyperlink topology and employs the mutually reinforcing relationship of authority and hub.

The commercialization of the Internet causes most web pages containing hyperlinks to link to commercial sites and advertisements. In addition, the quality of the top 200 search results from search engines is not high enough. Consequently the base set contains many irrelevant pages which may affect the results of topic discovery and ranking. Therefore, reduction of the number of irrelevant web pages in the base set is the key concern for constructing a base set and requires further study.

It is interesting to observe the birth, rise and fall of cyber communities via the automatic topic discovery technique because a cyber community may be considered as epitome or a special part of the society. The techniques proposed in this paper can be applied to other applications for automatic patterns or groups identification, although not necessarily in the context of web pages and hyperlinks. In bibliometrics, author co-citation analysis (ACA) is used to identify researchers and publications in related research area, in which citations are used as links in topic discovery. In e-Business, merchandise recommendations and sale/promotion activities require identification of customer groups with a certain characteristic. Topic discovery techniques may help in identification using customer purchase history, certain attributes (e.g. home location, age, sex, income level, club membership) and associations among customers as links. The precise interpretation of automatic discovery needs further study and a lot of input from domain experts.

References

Amento, B., Terveen, L., & Hill, W. (2000). Does "Authority" mean quality? Predicting expert quality ratings of web documents. In *Proceedings of the 23rd annual international ACM SIGIR conference on research and development in information retrieval, Athens, Greece* (pp. 296–303).

Bharat, K., & Henzinger, M. R. (1998). Improved algorithms for topic distillation in a hyperlinked environment. In *Proceeding of the 21st annual international ACM SIGIR conference on research and development in information retrieval, Melbourne, Australia* (pp. 104–111).

Borodin, A., Roberts, G. O., Rosenthal, J. S., & Tsaparas, P. (2001). Finding authorities and hubs from link structure on the World Wide Web. In *Proceedings of 10th international World Wide Web conference, Hong Kong* (pp. 415–429).

Botafogo, R. A., Eivlin, E., & Shneiderman, B. (1992). Structural analysis of hypertexts: identifying hierarchies and useful metrics. *ACM Transactions on Information System, 10*(2), 142–180.

Brin, S., & Page, L. (1998). The anatomy of a large-scale hypertextual web search engine. *Computer Networks and ISDN Systems, 30*(1–7), 107–117.

Chaffee, J., & Gauch, S. (2000). Personal ontologies for web navigation. In *Proceedings of the 9th international conference on information and knowledge management, McLean, VA* (pp. 227–234).

Chakrabarti, S., Joshi, M., & Tawde, V. (2001). Enhanced topic distillation using text, markup tags, and hyperlinks. In *Proceeding of the 24th annual international ACM SIGIR conference on research and development in information retrieval, New Orleans, LA* (pp. 208–216).

Chakravarthy, A. S., & Haase, K. B. (1995). Netserf: using semantic knowledge to find Internet information archives. In *Proceedings of the 18th annual international ACM SIGIR conference on research and development in information retrieval, Seattle, WA* (pp. 4–11).

Chang, H., Cohn, D., & McCallum, A. (2000). Creating customized authority lists. In *Proceedings of the 17th international conference on machine learning, Stanford, CA*.

Cohn, D., & Chang, H. (2000). Probabilistically identifying authoritative documents. In *Proceedings of the 17th international conference on machine learning, Stanford, CA*.

Davison, B. D. (2000). Topical locality in the web. In *Proceedings of the 23rd annual international ACM SIGIR conference on research and development in information retrieval, Athens, Greece* (pp. 272–279).

Davison, B. D., Gerasoulis, A., Kleisouris, K., Lu, Y., Seo, H.-J., Wang, W., & Wu, B. (1999). DiscoWeb: applying link analysis to web search. In *Poster proceedings of the 8th international World Wide Web conference, Toronto, Canada* (pp. 148–149).

Dean, J., & Henzinger, M. R. (1999). Finding related pages in the World Wide Web. In *Proceedings of the 8th international World Wide Web conference, Toronto, Canada* (pp. 389–401).

Ding, C., He, X., Husbands, P., Zha, H., & Simon, H. D. (2002). RageRank, HITS, and a unified framework for link analysis. In *Proceedings of the 25th annual international ACM SIGIR conference on research and development in information retrieval, Tampere, Finland* (pp. 353–354).

Ellis, D., Furner-Hines, J., & Willett, P. (1994). On the measurement of inter-linker consistency and retrieval effectiveness in hypertext databases. In *Proceedings of the 17th annual international ACM SIGIR conference on research and development in information retrieval, Dublin, Ireland* (pp. 50–61).

Farahat, A., Nunberg, G., & Chen, F. (2002). AuGEAS (authoritativeness grading, estimation, and sorting). In *Proceedings of the 7th international conference on information and knowledge management, McLean, VA* (pp. 194–202).

Géry, M. (2002). Non-linear reading for a structured web indexation. In *Proceedings of the 25th annual international ACM SIGIR conference on research and development in information retrieval, Tampere, Finland* (pp. 379–380).

Gibson, D., Kleinberg, J. M., & Raghavan, P. (1998). Inferring web communities from link topology. In *Proceedings of the 9th ACM conference on hypertext and hypermedia, Pittsburgh, PA* (pp. 225–234).

Kleinberg, J. M. (1998). Authoritative sources in a hyperlinked environment. In *Proceedings of the 9th annual ACM-SIAM symposium on discrete algorithms, San Francisco, CA* (pp. 668–677).

Kumar, R., Raghavan, P., Rajagopalan, S., & Tomkins, A. (1999a). Trawling the web for emerging cyber-communities. In *Proceedings of the 8th international World Wide Web conference, Toronto, Canada* (pp. 403–415).

Kumar, R., Raghavan, P., Rajagopalan, S., & Tomkins, A. (1999b). Extracting large-scale knowledge bases from the web. In *Proceedings of IEEE international conference on very large data bases, Edinburgh, Scotland, UK* (pp. 639–650).

Kumar, R., Raghavan, P., Rajagopalan, S., Sivakumar, D., Tomkins, A., & Upfal, E. (2000). The web as a graph. In *Proceedings of ACM symposium on principles of database systems, Dallas, TX* (pp. 1–10).

Lempel, R., & Moran, S. (2000). The stochastic approach for link-structure analysis (SALSA) and the TKC effect. In *Proceedings of 9th international World Wide Web conference, Amsterdam, Netherlands* (pp. 387–401).

Luke, S., Spector, L., Rager, D., & Hendler, J. (1997). Ontology-based web agents. In *Proceedings of the 1st international conference on autonomous agents, Marina del Rey, CA* (pp. 59–68).

Meghabghab, G. (2002). Discovering authorities and hubs in different topological web graph structures. *Information Processing and Management, 38*, 111–140.

Ng, A. Y., Zheng, A. X., & Jordan, M. I. (2001). Link analysis, eigenvectors, and stability. In *Proceedings of the 7th international joint conference on artificial intelligence, Seattle, WA* (pp. 903–910).