Inferring web communities through relaxed cocitation and dense bipartite graphs

P.Krishna Reddy and Masaru Kitsuregawa
Institute of Industrial Science, The University of Tokyo
7-22-1, Roppongi, Minato-ku, Tokyo 106, Japan
{reddy, kitsure}@tkl.iis.u-tokyo.ac.jp

Abstract

Community forming is one of the important activity in the Web. The Web harbors a large number of communities. A community is a group of content creators that manifests itself as a set of interlinked pages. Given a large collection of pages our aim is to find potential communities in the Web. In the literature, Ravi Kumar et al. [18] proposed a trawling method to find potential communities by abstracting a core of the community as a group of pages that form a complete bipartite graph (CBG) (web-page as a node and link as an edge between two nodes). The trawling approach extracts a small group of pages that form a CBG, which is a signature of a potential community.

In this paper we propose a different approach to find potential community patterns by analyzing linkage patterns in a large collection of pages. We mathematically abstract a community as a dense bipartite graph (DBG) pattern. Given a large collection of pages, we propose an algorithm to find all DBGs. It can be noted that a CBG pattern is an instance of a DBG pattern. By extracting potential DBG patterns, the proposed approach extract potential community patterns that exist in a page collection including CBG patterns.

We report experimental results on 10 GB TREC (Text REtrieval Conference) data collection that contains 1.7 million pages and 21.5 million links. The results indicate that as compared to trawling approach, the proposed approach extracts community patterns of significantly large size. Also, the proposed approach has good scale-up properties and can be easily parallelized.

Keywords: Web mining, Communities, Trawling, Link analysis.

1 Introduction

One of the most powerful socializing aspects of the Web is its ability to connect groups of like minded people independent of geography or time zones. The Web lets people join communities across the globe and provides the opportunity to form associations with outside world. In the Web environment one is limited only by his/her interests. As a result, the Web dramatically increases the number of communities you can bond to. For instance, in the past one might have had time to be a part of his/her neighborhood community and one or two social organizations. However, in the Web environment, one gets vast opportunity to form connections as entire world is at his/her disposal. Thus, community forming is one of the important activity in the Web. The Web has several thousand well-known, explicitly defined communities- groups of individual users who share a common interest. Most of these communities manifest themselves as news groups, Web-rings, or as resources collections in directories such as Yahoo and Infoseek, or home pages of Geocities.

In this paper we focus on the problem of finding all potential communities in a given page collection. Such communities include those emerging communities which are not manifested or not well-known as those listed in the Yahoo or other search engines. Some of such emerging communities have a potential to become a full-fledged communities in future. If we find these communities early it may serve many purposes. These communities provide valuable and possibly the most reliable resources for the user interested in them. They also represent the sociology of the Web. By enabling the people to know the existence of such communities, they can target their advertising selectively. Also since interest-based communities are forming with members from all over the world, the governments can engage (or disengage) these communities to meet their objectives. For instance, communities can enable people to shop, get news, meet each other, be entertained, and gossip or in other ways.

In the literature, Ravi Kumar et al. [18] proposed a trawling method to find potential communities by abstracting a core of the community as a group of pages that form a complete bipartite graph (CBG)(web-page as a node and link as an edge between two nodes). Given a large collection of pages, the trawling algorithm first extracts all the potential CBG cores and then expands each core to full-fledged community using the HITS algorithm. The trawling approach extracts a small group of pages that form a CBG, which is a signature of a potential community.

In this paper we propose a simple and efficient approach to find potential community signatures that exists in a large collection of pages. The community signature is extracted by analyzing a linkage pattern among a small

group of pages. We mathematically abstract community as a dense bipartite graph (DBG) pattern by considering web pages as nodes and links as edges. For each page, the community is extracted among a group of related pages retrieved using the proposed $relax\_cocite$ relationship. After gathering related pages using $relax\_cocite$ relationship, the potential community is extracted based on the existence of a potential DBG pattern through iterative pruning techniques. By extracting potential DBGs, the proposed approach is able to extract potential communities that exist in any page collection. It can be noted that a CBG is an instance of a DBG. So the proposed approach is able to extract DBG patterns that could form potential communities including CBG patterns.

To show the effectiveness of proposed approach we report experimental results conducted on 10 GB TREC (Text REtrieval Conference) data collection that contains 1.7 million pages and 21.5 million links. The results show that proposed approach extracts significantly big community patterns as compared to corresponding CBG patterns extracted with trawling approach.

The trawling approach emploes a priori algorithm [1] to extract all CBGs. We follow a different approach in which the time to find all communities increases linearly with number pages in a page collection. In addition, the proposed approach can be easily parallelized.

The rest of the paper is organized as follows. In the next section we define the proposed community through bipartite graph abstraction and discuss the motivation. In section 3 we discuss the related work. In section 4 after explaining $cocite$ and $relax\_cocite$ relationships, we propose the community extraction algorithm. In sections 5, we report experimental results conducted on 10GB TREC data. In section 6, we discuss performance advantages of proposed approach and other related issues. The last section consists of summary and future research.

2 Communities and bipartite graphs

We first explain some terminology used in this paper. A page is referred by its URL, which also denotes a node in a bipartite graph. We refer a page and its URL interchangeably. If there is an hyper-link from page u to page v, we say u is a parent of v and v is a child of u. An hyper-link from one page to other page is considered as an edge between the corresponding nodes in the bipartite graph. For a page u, parent(u) is a set of all parent pages (nodes) of u and child(u) is a set of children pages of u.

The input to the community detection process is a large collection of pages, which is denoted by a set $page\_set$ (PS). The terms $targets$ (T) and $interests$ (I) denote the set of URLs which also denote two groups of nodes in a bipartite graph. Note that, $T \subset PS$ and $I \subset PS$. Also, the terms targets-count (tc) and interests-count (ic) denote the number of pages in T and I respectively.

Here, we give the definition of a bipartite graph.

Definition 1 Bipartite graph (BG) *A bipartite graph BG(T,I) is a graph whose node-set can be partitioned into two non-empty sets T and I. Every directed edge of BG joins a node in T to a node in I.*

Note that BG is dense if many of possible edges between T and I exist. In a BG, the linkage denseness between the sets T and I is not specified. Here, we define a DBG which captures linkage denseness between the sets T and I.

Definition 2 Dense bipartite graph (DBG) *A DBG(T, I, α, β) is a BG(T,I) where, (i) each node of T establishes an edge with at least α ($1 \leq \alpha \leq ic$) nodes of I, and (ii) at least β ($1 \leq \beta \leq ic$) nodes of T establish an edge with each node of I.*

Now we define the complete bipartite graph that contains all possible edges between the nodes of T and the nodes of I.

Definition 3 Complete bipartite graph (CBG) *A CBG(T,I) is a DBG(T,I, α, β), where $\alpha = ic$ and $\beta = tc$.*

We consider a community as a collection of pages that form a linkage pattern equal to a DBG. Our definition is based on the following intuition: *Web communities are characterized by DBGs.* In the Web environment, page-creator (a person who creates the page) creates the page by putting the links to other pages of interest in isolation. Since a page-creator by and large puts the links to display his interests, we believe that if multiple pages are created with similar interests, at least few of them have common interests. Our intuition is that such a phenomena can be captured through a DBG abstraction.

A community phenomena can also be captured through a CBG abstraction[18]. A CBG abstraction extracts a small set of potential members to agree on some common interests. Given a very large collection of pages, for each community there might exist few pages that could form a CBG. Given a reasonably large collection of pages, there is no guarantee that each community formation is reflected as a CBG core. Because a PS may not contain the potential pages to form a CBG. Further, it can be observed that, it rarely happens that a page-creator puts links to all the pages of interest in particular domain. Also, given the limited collection of pages it is not always possible to find the large potential communities through a CBG abstraction because page-creators put links in a page in an arbitrary manner.

Normally, each member in a community shares interests with few other members. Therefore, as compared to a CBG abstraction, abstraction of a community pattern through a DBG matches well with real community patterns. In general community can be viewed as a macro-phenomena created by complex relationships exhibited by corresponding members. At micro-level, each member establishes relationships with few other members of the same community. Integration of all members and their relationships exhibit a community phenomena. In the context of Web, a DBG abstraction enables extraction of a community by integrating such micro-level relationships.

Note that not all DBGs are of interest in the context of communities. Now we give the community definition by fixing threshold values for both α and β in a DBG.

Definition 4 Community. *The set T contains the members of the community if there exist a dense bipartite graph DBG(T, I, α, β), where $\alpha \geq \alpha_t$ and $\beta \geq \beta_t$, where α_t and β_t are nonzero integer values which represent threshold.*

It can be observed that we have defined the community by keeping the number of nodes in both T and I unspecified. We specify only linkage denseness with both α and β for a given PS. The values of α_t and β_t are fixed with a feedback after examining the potential correspondence with the real community patterns. These values are fixed such that by extracting such patterns we should establish a close relationship among the members of T.

For a given PS, we state some properties of CBG and DBG through the following theorem. To do that we define the following sets.

Definition 5 Dense bipartite graph set (DBGS) *Let p and q be nonzero integers. Then, DBGS(p,q)=$\{$DBG(T,I, p, q) $\mid ic \geq p$ and $tc \geq q\}$.*

Definition 6 Complete bipartite graph set (CBGS)) *Let p and q be non-zero integers. Then, CBGS(p,q)= $\{$CBG(T, I) $\mid tc \geq p$ and $ic \geq q\}$.*

It can be observed that in a DBG(T,I, p, q), both p and q specify the linkage denseness whereas in a CBG(p, q) same denote the number of nodes in T and I respectively. Figure 1 shows the difference between DBG(T,I, p, q) and CBG(p,q).

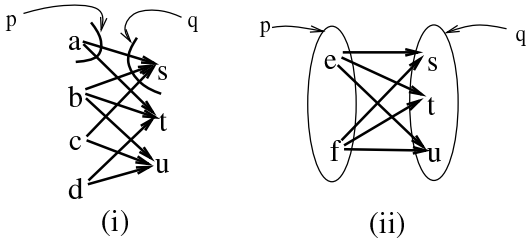

Figure 1. Graphs: (i) DBG(T,I, p, q) (ii) CBG(p,q)

Theorem 1 *For a given PS, the following always hold. Let p, q, r and s be nonzero integer values.*

i CBGS(p,q) $\subseteq$ DBGS(q, p).

ii If DBG(T,I, r,s) $\in$ DBGS(q, p) $\not\Longrightarrow$ CBG(s, r) $\in$ CBGS(q,p).

Proof: *According to case (i) all CBGs are the instances of DBGs. That is, at fixed p and q values, if we extract all DBGS(T,I, q,p), all the CBGs in CBGS(p,q) are automatically extracted. According to definition, a DBGS(T,I, q, p) includes all the DBG(T,I, q, p) patterns such that $tc \geq$ p and $ic \geq$ q. This implies that DBGS(T,I, q, p) includes a CBG(p,q) with p = tc and q = ic.*

According to case (ii), the existence of DBG(T,S, r,s) does not guarantee the existence of CBG(s,r). As per definition, a DBGS(T,I, r, s) includes all DBG(T,I, r, s) such that $ic \geq$ r and $tc \geq$ s in the given page collection. So an existence of a DBGS(T,I, r, s) does not guarantee an instance where r = ic and s = tc.

3 Related work

In this section we review the approaches proposed in the literature related to community detection, data mining and link analysis.

3.1 Community related research

The HITS algorithm

The HITS (Hperlink-Induced Topic Search) algorithm [16] is one of the widely used algorithm in search engines to find authoritative resources in the Web that exploits connectivity information in the Web. The intuition behind the HITS algorithm is that the document that points to many others is a good hub, and a document that many documents point to is a good authority. Transitively, a document that pints to many good authorities is an even better hub, and similarly a document pointed to by many good hubs is an even better authority. The HITS algorithm repeatedly updates hub and authority scores so that documents with high authority scores are expected to have relevant contents, whereas documents with high hub scores are expected to contain links to relevant contents.

In [14], communities have been analyzed which are found based on the topic supplied by the user by analyzing link topology using HITS. In that paper the community is defined as a core of central $authoritative$ pages linked together by hub pages.

The basic idea behind the community detection process using HITS is mutual reinforcement: good hubs point to good authorities; and good authorities are pointed by good hubs. The HITS algorithm finds good authority pages given a collection of pages on same topic. Our motivation is to find all the communities from a larger collection of pages covering wide variety of topics.

The Trawling algorithm

Given a large collection of pages, the trawling algorithm [18] extracts communities by first finding all possible cores and then expanding each core to a full-fledged community with HITS.

The community detection in the trawling algorithm is based on the assumption that DBGs (with two groups of nodes F and C) that are signatures of web communities contain at least one CBG with at least i nodes from F and at least j nodes from C. The CBG(i,j) is called the core of the community. Given a large collection of pages, the trawling algorithm first extracts all the potential CBG cores and then expands each core to full-fledged community using the HITS algorithm.

The proposed approach extracts members of the community by extracting the DBG patterns of potential communities. For instance, as per trawling algorithm the minimum criteria for a core is one should extract a CBG

from the collection of Web pages. Suppose we fix a minimum criteria as CBG(2,3) as shown in Figure 1(ii). Figure 1(i) depicts a possible core extracted by proposed approach. It can be observed that the DBG shown in Figure 1(i) fails to satisfy minimum criteria, so can not be extracted by employing trawling algorithm. In this paper we also consider Figure 1(i) as a possibility of community formation.

The proposed approach is complementary to HITS and trawling algorithms. Given a very large page collection, the trawling algorithm may extract all the communities by extracting CBGs. The main advantage of proposed approach (see experiment results) is that it extracts significantly big graph patterns as compared to the corresponding CBG patterns.

To find all the CBG patterns in the trawling approach, the trawling algorithm employs a priori algorithm [1]. But we follow a different approach for community detection. For each page we extract related pages using the $relax\_cocite$ relationship. We then perform iterative pruning technique to extract a DBG structure. The complexity of proposed approach is linear as amount of computation time to find all communities increases linearly with number pages in a page collection. Further, it can be easily parallelized.

Flow-based approach

In [12], given a set of crawled pages on some topic, the problem of detecting a community is abstracted to maximum flow /minimum cut framework, where as the source is composed of known members and the sink consist of well-known non-members. Given the set of pages on some topic, a community is defined as a set of web pages that link (in either direction) to more pages in the community than to the pages of outside community.

The flow based approach can be used to guide the crawling of related pages. In this paper we try to extract all the community structures in a given collection.

3.2 Other approaches

The data mining approach [1] focuses largely on finding association rules and other statistical correlation measures in a given data set. The notion of finding communities differs from the fact that, in our approach the relationship we exploit is co-citation whereas in data mining is performed based on the support and confidence.

One of the earlier uses of link structure is found in the analysis of social networks [19], where network properties such as cliques, centroids, and diameters are used to analyze the collective properties of interacting agents. The fields of citation analysis [13] and bibliometrics [24] also use citation links between works of literature to identify patterns in collections.

Most of the search engines perform both link as well as text analysis to increase the quality of search results. Based on link analysis many researchers proposed schemes [8, 9, 11, 7, 17, 16, 4] to find related information from the Web.

In this paper we extended the concept of cocitation to the web environment to detect communities in the Web.

4 Proposed approach for DBG extraction

Web-page creators keep links in a page for different reasons. For example, one may put a link to other page to direct the relevant information, to promote the target page or as an index pointer. In this paper we consider the existence of a link from one page to another page as a display of interest by the former on the later page.

In the web environment, web pages can be grouped based on the type of relationship (association, pattern, or criteria) defined among pages. For example, in an information retrieval environment, the documents are searched based the notion of syntactic relationship that is measured based on the existence of number of common keywords. Similarly, one could define any type of relationship among the web pages and investigate the efficiency through experiments. In the Web environment researchers have defined different types of relationships to group web pages. Existence of a link, cocitation, coupling, number of paths between web pages are some examples of relationships.

In this paper we have investigated finding communities based on the $relax\_cocite$ relationship which is a relaxed version of the $cocitation$ relationship. We first discuss about the $cocite$ relationship to search related information in the Web. Next. after explaining $relax\_cocite$, we present the proposed algorithm.

4.1 Cocite

The fields of citation analysis [13] and bibliometrics [24] also use citation links between works of literature to identify patterns in collections. Co-citation [20] and bibliographic coupling [15] are two of the more fundamental measures used to characterize the similarity between documents. The first measures the number of citations in common between two documents, while the second measures the number of documents that cite both of two documents under consideration.

Also, in the information retrieval literature, relationship between documents can be established with keywords that exist in both documents. Similarly, in a web environment as we have considered link as a display of interest on the target page, by dealing with only links we can establish association among pages based on the existence of common children (or URLs). That is, we can establish association among pages through number of common children. We call this relationship $cocite$ as in bibliographical terms if two documents [20] refer a collection of common references, we say, they cocite[1] them. We formally define the $cocite$ relationship in the context of Web environment as below. Figure 2(i) depicts the cocite relationship between pages p and q with $cocite\_factor$ = 3.

Definition 7 Cocite *Let u and v are pages.*

[1]Note that in this paper we treat two documents are related as per $cocite$ if they cite a group of documents and as per $couple$ if a group of documents cite them. In this paper, we community extraction algorithm based on relaxed form of cocitation.

Then, cocite(u,v)=true, if $child(u) \cap child(v) \geq couple\_factor$*, where* $cocite\_factor \geq 1$.

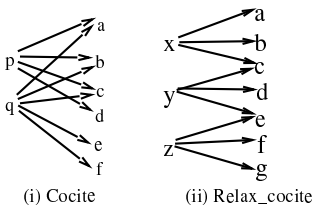

Figure 2. Depiction of $cocite$ and $relax\_cocite$.

4.2 Relax_cocite

According to *cocite,* a set of pages is related, if there exist a set of common children. Even though $cocite$ is defined to establish relationship between two documents, it could form association among multiple documents in the following way. We consider two pages u and v in the PS are related if both have common links at least equal to $cocite\_factor$. Similarly, n ($n \geq 2$) pages are related under *cocite* if these pages have common children at least equal to $cocite\_factor$. If a group of pages are related according to *cocite* relationship, these pages form an appropriate CBG.

However, to extract a DBG, we have to retrieve a collection of pages loosely related. So we relax the *cocite* relationship to find loosely related pages in the following manner. We allow pages u,v and w to to group if cocite(u,v) and cocite(v,w) are true. This modification enables relationship between a page and multiple pages taken together. That is, if a page could not form association with another page according to *cocite,* it does not imply that they are different. Even though a page fails to satisfy a certain minimum criteria page-wise, however, it could satisfy minimum criteria with multiple pages taken together. We define the corresponding new definition, $relax\_cocite$ as follows.

Definition 8 Relax_cocite. *Let* T *be the set of pages and* u *be the another page (*$u \notin comm\_set$*). For any page* $p \in T$, $relax\_cocite(u,p)$=*true if* $child(u) \cap child(T) \geq relax\_cocite\_factor$. *Here, child(T) denotes all the children of T.*

Figure 2(ii) depicts the $relax\_cocite$ relationship among web pages x, y and z, with $relax\_cocite\_factor$ equal to 1. It can be observed that for a new page u, as compared to cocite, $relax\_cocite$ increases the probability of association with p, as child(T) is larger than child(p).

However, note that for a given page, $relax\_cocite$ may gather pages that are semantically different from the starting page. However, after collecting a reasonable number of pages we employ effective pruning methods to extract a DBG by pruning non-potential pages.

4.3 Proposed algorithm

Given a large collection of pages, the proposed algorithm to extract community patterns consists of following steps: gathering related pages and community extraction. For each page, we gather related pages during gathering phase through $relax\_cocite$ relationship. We then apply the iterative pruning technique to extract a DBG(T, I, α, β). We now present the algorithm.

1. **Gathering related pages**
   In this step for a given a URL p, we find T (set of URls). The default value for $relax\_cocite\_factor$ is 1. The variable $num\_iterations$ is an integer (> 0) variable. Set T = p.
   (a) While $num\_iterations \leq$ n ($n \geq 1$)
      i. At a fixed $relax\_cocite\_factor$ value, find all w's such that $relax\_cocite$(w, T) = true.
      ii. T = w $\cup$ T.
   (b) Output T.
2. **Community extraction**
   In this step the input contains T produced from the preceding step and the output contains a dense bipartite graph, DBG(T, I, α, β). Let $edge\_file$ be the set of elements $< p, q >$ where p is a parent (source) of child q (destination). Let, T1 and I1 be a set of tuples of the form $< URL, freq >$. Set T1, I1, and $edge\_file$ to ϕ.
   (a) For each $p \in T$, insert the edge $< p, q >$ in $edge\_file$ if $q \in child(p)$.
   (b) While $edge\_file$ is not converged repeat the following.
      i. Sort the $edge\_file$ based on the source. Prepare T1 with $< source, freq >$. Remove $< p, q >$ from $edge\_file$ if $freq < \alpha$.
      ii. Sort the $edge\_file$ based on the destination. Prepare I1 with $< q, freq >$. Remove $< p, q >$ from $edge\_file$ if $freq < \beta$.
   (c) The resulting $edge\_file$ represents a DBG(T, I, α, β) where, T={ p $\mid < p, q > \in$ $edge\_file$ } and I= {q $\mid < p, q > \in$ $edge\_file$ }.

5 Experiment results

5.1 Description of data-collection

We report experimental results conducted on 10 GB TREC [22] (Text Retrieval Conference [21]) data collection. It contains 1.7 million web pages. We reproduce the following text on the web page that explains properties of the data collection.

The purpose of the Web Track is to have a framework, based on a snapshot of the World Wide Web, within which new search techniques can be reliably evaluated and within which repeatable experiments may be conducted.

Web Collections: *ACSys (Advanced Computational Systems) has developed three Web document corpuses*

based on a 320 gigabyte crawl of the World Wide Web by the Internet Archive in early 1997.

The VLC2 (Very Large Collection No.2) consists of the first 100gB of Web data from the crawl which was then minimally reformatted. This dataset is also known as WT100g, and is used in the Large Web Task.

The newest collection is WT10g, a 10.3gB subset of the VLC2 collection. It has been developed for use in TREC-9's Main Web Task. WT10g has various properties that we hope will make it more suitable for conducting particular kinds of Web retrieval experiments, including those involving link-based methods and distributed information retrieval methods.

5.2 Preprocessing and link-file preparation

For a given page collection, link-file contains all the links of the form $< p,q >$ where $p \in parent(q)$. We prepare a link-file through the following steps (for details see [18]): extracting all the links, eliminating the duplicates and removing both popular- and unpopular pages.

The pages are in the text format with html marking information. We have extracted links by ignoring all the text information. We then created a link-file for a entire page collection in the following manner. We employed 32 bit fingerprint function to generate a fingerprint for each URL. Each page is converted into a set of edges of the form <source , destination>, where source represents the title URL and destination represents the other URL in the page. The total number of edges comes to 21.5 million.

Next, we have removed the possible duplicates by considering two pages duplicates if they have a common sequence of links. We employed the algorithm proposed in [3] to remove the duplicates. We have selectted shingle window size as four links. We kept at most three shingles per page. We have considered two pages as duplicates even one shingle is common between them. We found that considerable number of pages are duplicates. After the duplicate elimination, the total number of edges comes to 18 million.

Next we have removed edges derived from both extreme popular and unpopular pages. The popular pages are those which are highly referred in the Web such as WWW.yahoo.com. Also the unpopular pages are those which are least referred. We considered a page as popular if it has more than 50 parents (we have adopted this threshold from [18]). We considered a page as unpopular if it has less than two parents. After sorting the link-file based on the destination, those pages having number of parents greater than fifty and less than two are removed. Also, we removed pages with one child by considering that these do not contribute to community finding. So, after sorting based on the source, the links which have number of children less than two removed. The above two steps are performed repetitively until the number of edges converge to a fixed value. After this step the number of edges comes to 6.5 million.

This link-file is used to retrieve both parents and children of a given page during community extraction process.

5.3 Results

Here, we first explain the charecteristics of gathering phase. We then discuss community extraction using proposed approach and trawling approach. Next, we show some examples of real community patterns extracted using proposed approach from TREC data collection.

5.4 Gathering phase

Figure 3 shows how number of pages in T increases with number of iterations. This figure represents the curves of four URLs that are selected randomly. It can be observed that the number of pages grow exponentially (y-axis is log scale) with number of iterations. Beyond three, the number of pages explode.

It has been observed that with number of iterations beyond 1, the pages in T are found to be too loosely related. Since our aim is to find all communities, we extracted communities by restricting to one iteration.

However, it has been observed that DBG obtained by exceeding one iteration displays different properties. Let k be a URL. Starting with k, let $targets1$ be the set of pages gathered with iteration 1. Also, $comm\_targets1$ be the set of communities obtained for each URL in $targets1$ with one iteration. Now, with k as an input, if we go for two iterations, the resulting DBG may contain a high level community integrating all the communities in $comm\_targets1$. So such a property can be exploited to relate existing communities. Since the aim of this paper is to find existing communities a given page collection, we restrict to one iteration only. However, the properties of proposed algorithm going beyond one iteration will be investigated as a part of future work.

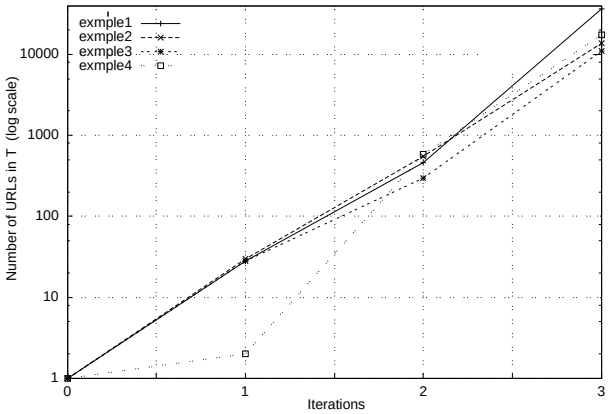

Figure 3. Expansion of pages during gathering phase.

5.5 Community extraction

In the gathering phase, for each page we find a set of related URLs by fixing number of iterations to 1. Among these pages, we find community by extracting DBG(T,I, α, β).

Figure 4 shows the number of DBG(T,I, α, β) patterns for all the pages that constitute link-file. The total number of pages that constitute link-file is around 0.7 million. For a DBG(T,I, α, β), the column ''(avg(T), avg(T))'' indicates averages number of pages in T and I. It was observed that for every instance of CBG, corresponding DBG exists. However, as compared to number of nodes in T and I of CBG(T,I), the number of nodes in T and I significantly increased in DBG(T,I, α, β). That is, in CBG(α, β) the number of pages in T and I are almost equal to α and β. However, in DBG(T,I, α, β), it can be observed from Figure 4 that the number nodes in T and I significantly high.

The results show that proposed approach extracts significantly big community patterns as compared to community patterns extracted with trawling approach.

(α, β)	**# of DBG(T, I, α, β)**	**(avg(T), avg(I))**
(2,3)	110422	(36.21, 162.6)
(2,4)	81135	(36.98, 109.65)
(2,5)	61566	(36.15, 83.465)
(3,3)	90129	(32.86, 192)
(3,4)	59488	(32.26, 140.56)
(3,5)	40708	(30.17, 114.93)
(4,3)	66670	(34.29, 244.81)
(4,4)	49051	(27.75, 159.62)
(4,5)	32309	(24.97, 134.33)
(5,5)	28296	(21.07, 145.09)
(6,6)	17335	(19.03, 161.67)
(7,7)	10960	(18.97, 198.17)

Figure 4. Graph details: number of DBG patterns, average number of pages in T and I.

Figure 5 shows how number of DBG(T,I, p, p) vary with p, where p is varied from 3 to 7. Figure 5 shows the corresponding graph. Predictably, as p increases the number of DBG patterns decreases exponentially.

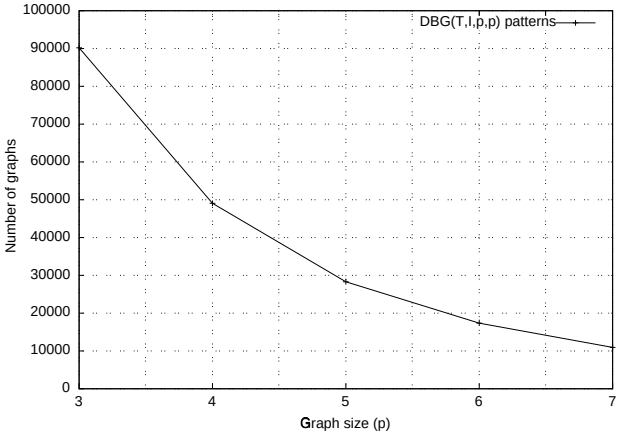

Figure 5. Size(p) versus DBGS(T,I,p,p). Number of pages = 733582.

5.6 Examples

Here we provide four potential communities extracted from 10GB TREC data collection. The $Targets$ represents the potential members of the community (corresponding topics are indicated using brackets) and $Interests$ represent the potential children of the community. Also, corresponding edges of the graph are also given. In the graph the notation (p,q) represents an edge, where $p \in targets$ and $q \in interests$. Also, in the corresponding figure edges (p,q) is represented as an arrow from p to q.

All graphs represent DBG(T, I, 3, 3) where T contains members and I contains thirteen children, i.e., each parent has at least 3 children and at least 3 parents have one common child.

1. Topic: Kids
In this community pattern CBG(3,3) is embedded in DBG(T, I, 3, 3). Figure 6 shows corresponding graphs.

Targets
1. http://www.cais.com/makulow/kid.html (KID List)
2. http://www.highstar.com/kids/childlist.html (Highstar Kids's Links
3. http://www.eagle.ca/ matink/kids.html (KIDS PAGE)
4. http://sdirect.com/einstein/klinks.html (Einsteins Chalkboard..SERVDirect)
5. http://about.ferris.edu/weblinks/kids.htm (Kid's World)
6. http://people.delphi.com/GEAATL/ (DIXIE and LAW - The Two Topic Site)
7. http://www.starpoint.net/kids.html (NetPoint's Kids Page)
8. http://ns.tincup.com/kids/index.html (TinCup.com Kids Page)
9. http://www.scs.on.ca/fun.htm (Fun)
10. http://www.come2az.com/info/4kids.htm (Alice Held's Arizona Relocation Guide!-4Kids)

Interests
1. http://www.safesurf.com/kidswave.htm
2. http://www.public.iastate.edu/ jmilne/pooh.html
3. http://www.internet-for-kids.com/
4. http://www.youcan.com/
5. http://www.crayola.com/
6. http://www.cyberjacques.com/
7. http://www.pbs.org/rogers/mrr_home.html
8. http://www.mca.com/home/playroom/
9. http://www.telenaut.com/gst/
10. http://www.uoknor.edu/oupd/kidsafe/start.htm
11. http://www.seussville.com/
12. http://www.ieighty.net/ matthewg/
13. http://www.aha-kids.com/

Graph:{(1, 3), (1,6), (1, 11), (1,12), (1, 13), (2, 2), (2, 5), (2,11), (2, 13), (3,1), (3, 2), (3, 3), (3, 4), (3,5), (3,6), (3,8) (3, 9), (3, 10), (3, 11), (3, 12), (4, 2), (4, 6), (4, 7), (4, 8), (4, 11), (4, 13), (5, 2), (5, 3), (5, 4), (5, 7), (5, 8), (5, 11), (5, 13), (6,1), (6, 5), (6, 11), (7, 7), (7, 10), (7, 11), (8, 4) (8, 8), (8, 11), (9, 3), (9, 9), (9, 10), (9, 11), (9, 12), (10, 1), (10, 9), (10, 11) }

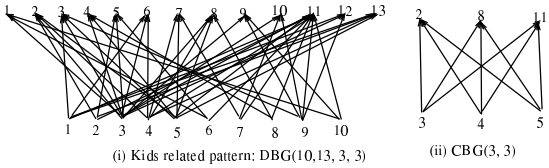

Figure 6. A community pattern: (i) DBG(10,13,3,3) (ii) CBG(3,3)

2. Topic: Comedy In this community pattern only DBG(6,6,3,3) exists without CBG(3,3) pattern. Figure 7 shows corresponsing graph.

Targets

1. http://www.tnef.com/jim_carrey.html (Jim Carrey - (15 links) Actors)
2. http://www.comedyweb.co.uk/cwlinks.htm (Comedy Web Links Page)
3. http://www.starcreations.com/abstract/laughriot/lr-fam01.htm (LAUGH RIOT - FAMOUSLY FUNNY)
4. http://www9.yahoo.com/Business_and_Economy/ Companies/Entertainment /Comedy/Comedians/ Carrey_Jim/ (Yahoo! - Business and Economy: Companies: Entertainment: Comedy:Comedians:Carrey, Jim)
5. http://www.scar.utoronto.ca/ 93kolmeg/starp.html (Personalities on Chog)
6. http://www.allny.com/comedy.html (New York Comedy Clubs)

Interests

1. http://q.continuum.net/ scout/jimpage.htm
2. http://www.halcyon.com/browner/
3. http://www.nd.edu/ jlaurie1/dmhome.html
4. http://www.cheech.com/
5. http://meer.net/ mtoy/steven_wright.html
6. http://www.en.com/users/bbulson/jim.html

Graph: { (1, 1), (1, 2), (1, 6), (2, 2), (2, 4), (2, 5), (3, 2), (3, 3), (3, 4), (3, 5), (4, 1), (4, 2), (4, 6), (5, 1), (5, 3), (5, 4), (5, 5), (6, 2), (6, 3), (6, 5), (6, 6)}

3. Topic: Environment and safty In this also we found a DBG(6,6,3,3) which does not contain corresponding CBG(3,3). Figure 7 shows corresponding graph.

Targets

1. http://www.saul.com/env/index.html (Saul, Ewing, Remick & Saul - 10: Environmental Law (PA, NJ, DE))
2. http://www.crystalcity.org/cfd/sitelinks.html (CFD links to other sites)
3. http://www.safetylink.com/ (Safety Link)
4. http://wwell.com/safety-resources/related-links.html (Safety Resources on the Web)
5. http://www.pixelmotion.ns.ca/WCB/links.html
6. http://www.mcaa.org/safety.htm (Safety & amp; Health)

Interests

1. http://atsdr1.atsdr.cdc.gov/toxfaq.html
2. http://www.ccohs.ca/
3. http://turva.me.tut.fi/ oshweb/
4. http://atsdr1.atsdr.cdc.gov/hazdat.html
5. http://www.wpi.edu/ fpe/nfpa.html
6. http://www.osha-slc.gov/

Graph: {(1, 1), (1, 4), (1, 6), (2, 4), (2, 5), (2, 6), (3, 2), (3, 3), (3, 5), (3, 6), (4, 3), (4, 5), (4, 6), (5, 1), (5, 2), (5, 3), (5, 4), (5, 6), (6, 1), (6, 2), (6,6) }

4. Telecommmunications related community This community forms a DBG(8, 13, 3, 3) without having corresponding CBG(3,3). Figure 7 also shows corresponding graph.

Targets:

1. http://gatekeeper.angustel.com/links/l-mfrs.html (Telecom Resources: Manufacturers)
2. http://gemini.exmachina.com/links.shtml (Wireless Links)
3. http://millenniumtel.com/ref-voic.htm (Millennium Telecom:References)
4. http://www.buysmart.com/phonesys/ phonesyslinks.html (BuyersZone: Phone systems)
5. http://www.commnow.com/links.htm (WirelessNOW Links Page)
6. http://eserver.sms.siemens.com/scotts/010.htm (http://www.smutking.com:80/)
7. http://www.searchemploy.com/research.html (Search & Employ)
8. http://www.electsource.com/elecoem.html (Electronics OEM's)

Interests

1. http://www.harris.com/
2. http://www.nb.rockwell.com/
3. http://www.cnmw.com/
4. http://www.mpr.ca/
5. http://www.brite.com/
6. http://www.pcsi.com/
7. http://www.ssi1.com/
8. http://www.mitel.com/
9. http://www.centigram.com/
10. http://www.adc.com/
11. http://www.dashops.com/
12. http://www.octel.com/
13. http://www.isi.com/

Graph : { (1, 4), (1,8), (1,9), (1,10), (1, 11), (2, 2), (2, 3), (2, 5), (2, 6), (2, 9), (2, 10), (3, 5), (3, 9), (3, 12), (4, 1), (4, 8), (4, 11), (5, 1), (5, 3), (5, 4), (5, 5), (5, 6), (5, 9), (5, 12), (5, 13), (6, 2), (6, 7), (6, 10), (6, 11), (6, 13), (7, 6), (7, 7), (7, 12), (7, 13), (8, 1), (8, 2), (8, 3), (8, 4), (8, 7), (8, 8), (8, 10) }

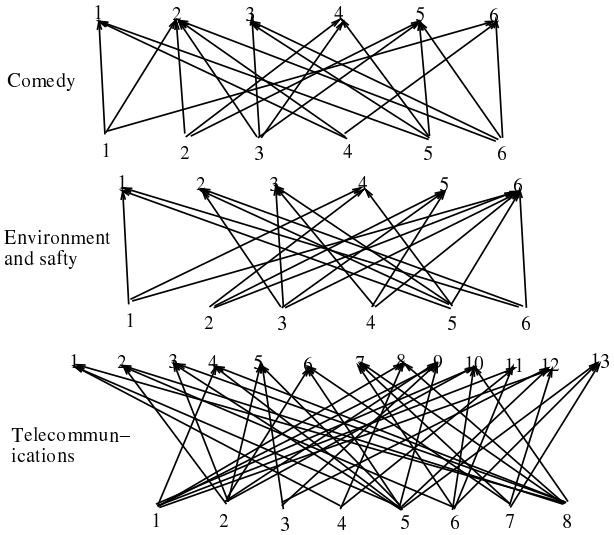

Figure 7. Community examples: Kids-, environment and safty, and telecommunitycations.

6 Discussion

In this section we discuss performance and other related issues.

Related to performance, the proposed approach has two main features: scalability and parallelism. The approach scales-up well as the time to find all the communities increases linearly with number of pages in the page collection. Let, P be the number of potential pages. The total processing time to find all the communities comes to P(t_g + t_c), where t_g and t_c represent averages times required in gathering step and community extraction step, respectively. If the link-file fits in a main memory the processing time required to find all the communities increases linearly with number of pages that constitute the link-file. It can be noted that current memory technology allows allows to keep a reasonably a large link-file in a main memory.

Also, it can be observed that the proposed algorithm can be easily parallelized. By copying the link-file at different nodes, the load can be distributed equally among the nodes thus reducing the processing time by a factor equal to number of parallel nodes.

Relatively, finding all the CBG patterns is a complicated process. In the trawling approach, DBGs of various sizes are calculated by mainly employing a priori algorithm [1]. So as compared to CBG based approach, proposed approch better suits to find potential communities having features of linear complexity and parallelism.

To extract emerging communities, we should run the proposed approach on the recent data collection. After extracting potential communities, the emerging communities can be found by excluding the communities that exist in Yahoo or other search engine portals [18]. In this paper it has been shown that the proposed approach can find significantly big community patterns in a given data collection. Also, all the CBG patterns are embedded in DBG patterns. So similar to tralwing approach, it also extracts potential emerging communities of big size if employed on a recent data collection.

It can be noted however that similar to trawling approach there are dublicate communities among the communities extracted by proposed approach. Designing an algorithm to find duplicates among the extracted communities will be investigated as a part of future work.

7 Summary and conclusions

In this paper, given a large collection of web pages, we proposed a simple and efficient approach to extract potential communities by analyzing linkage patterns among web pages. We mathematically abstracted a community with dense bipartite graph pattern. The proposed approach gathers related pages based on the $relax\_cocite$ relationship, and then follows iterative pruning technique to extract a potential DBG pattern. For a given data collection, the time to find all communities increases linearly with number pages. In addition, this algorithm can be easily parallelized. We demonstrated the effectiveness of proposed approach by finding potential communities in 10 GB TREC data collection. The results show that proposed approach extracts significantly big community patterns as compared to corresponding CBG patterns extracted with trawling approach.

We state that as compared to CBG abstraction, abstraction of community pattern through a DBG suits well with real community patterns. In general community is a macro phenomena created by complex relationships exhibited by corresponding members. At micro level, each member establishes with few other members of the same community. Integration of all members and their interests exhibit a community phenomena. A DBG abstraction enables detection of potential communities in a given page collection by capturing such micro-level relationships.

As part of future work, we will investigate the issue of grouping similar communities. We intend to extend the proposed ideas for searching relevant information and clustering in the Web.

Acknowledgments
This work is supported by ‘‘Research for the future’’ (in Japanese Mirai Kaitaku) under the program of Japan Society for the Promotion of Science, Japan.

References

[1] R.Agrawal and R.Srikant. Fast algorithms for mining association rules, in proc. VLDB Chile, 1994.

[2] Andrei Broder, Ravi Kumar, Farzin Maghoul, Prabhakar Raghavan, Sridhar Rajagopalan, Raymie Stata, Andrew Tomkins, Janet Wiener, Graph structure in the Web: experiments and models, 9th international WWW conference, May 2000.

[3] Andrei Z.Broder, Steven C.Glassman, Mark S.Manasse, and Geoffery Zweig, Syntactic clustering of the Web, 6th International WWW conference, 1997.

[4] K.Bharat and M.Henzinger, Improved algorithms for topic distillation in hyper-linked environments, in: proc: 21st SIGIR Conference, Australia, 1998.

[5] Jeffrey Dean, and Monica R.Henzinger, Finding related pages in the world wide web. 8th international WWW conference, 1999.

[6] Bill Gates, Business@the speed of thought, Warner Books, 1999.

[7] S.Brin and L.Page, The anatomy of a large scale hyper-textual web search engine, in proc. of 7th WWW Conference, April 1998, pp. 107-117.

[8] J.Carriere and R.Kazman. Web query: Searching and visualizing the web through connectivity. In proceedings of 6th WWW Conference, pp. 107-117, April 1997.

[9] S.Chakrabarti, B.Dom, D.Gibson, J.Kleinberg, P.Raghavan and S.Gopalan, Automatic resource compilation by analyzing hyper-link structure and associated text, in proc. of 7th WWW conference, 1998, pp. 65-74.

[10] Mark E.Crovella and Azer Bestavros, Self-Similarity in World Wide Web traffic evidence and possible causes, ACM SIGMETRICS, pp. 160-169, 1996.

[11] Ellen Spertus. Parasite: Mining structural information on the Web. In proceedings of 6th WWW Conference, pp. 587-595, April 1997.

[12] G.W.Flake, Steve Lawrence, C.Lee Giles, Efficient identification of web communities, The Sixth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, August 2000, pp.150-160.

[13] E.Garfield. Cocitation analysis as a tool in journal evaluation, Science, 178, 1772.

[14] D.Gibson, J.Kleinberg, P.Raghavan. Inferring web communities from link topology, in proc. ACM Conference on hypertext and hyper-media, 1998, pp. 225-234.

[15] M.M.Kessler. Bibliographic coupling between scientific papers. American Documentation, 14, 1963.

[16] J.Kleinberg, Authoritative sources in a hyper linked environment, proc. of ACN-SIAM Symposium on Discrete Algorithms, 1998. Also, appears as a IBM Research Report RJ 10076(91892) May 1997, and at http://www.cs.cornell.edu/home/kleinber/.

[17] Loren Terveen and Will Hill. Evaluating emergent collaboration on the Web. In Proceedings of ACM CSCW'98 Conference on Computer Supported Cooperative Work, Social Filtering, Social Influences, pp. 355-362, 1998.

[18] Ravi Kumar, Prabhakar Raghavan, Sridhar Rajagopalan, Andrew Tomkins, Trawling the Web for emerging Cyber-communities, 8th WWW Conference, May 1999.

[19] John Scott. Social Network analysis : a handbook. SAGE Publications, 1991.

[20] Small, H.G. Co-citation in the scientific literature: A new measure of the relationship between two documents. Journal of American Society for Information Science, 24, no. 4, pp.265-269, 1973.

[21] TREC: Text REtrieval evaluation (http://trec.nist.gov).

[22] http://pastime.anu.edu.au/TAR/vic2.html

[23] White , Howard D., and Belver C. Griffith. 1980. Author cocitation: A literature measure of intellectual structure. Journal of American Society for Information Science, 28, no. 5, pp.345-354, 1980.

[24] H.D.White and K.W. McCain, Bibliometrics, in: Annual Review of Information Science and Technology, Elsvier, 1989, pp. 119-186.