Clustering Techniques on Web Data

Nazife Karal, *Computer Engineering Department, Marmara University*

***Abstract*—Searching a specified topic on the web and finding it effectively without wasting time on unrelated topics is everybody's desire. Our concern is how to accomplish this by using web data mining. In this paper, we introduce three ways of clustering techniques on web data and compare them with each other. One of the areas we concentrated on is finding related pages on web by using Companion and Cocitation algorithms. The other one is finding replicated web collections by doing similarity analysis.**

***Index Terms*—Related pages, Search Engines, Similarity Analysis**

I. INTRODUCTION

NOWADAYS one of the most important problems on web is finding the requested documents or information in a short time. When you are searching for some topic on web, you enter that topic on the search engine such as Google or Yahoo. You will see so many pages that the engine has listed to you. Maybe there are unrelated ones with your topic, and there may be replicated ones among these pages. It will waste your time and you will be bored by doing that search. And maybe you will get the wrong information. However, if there is a system that gives you related pages about that topic or shows you the replicas of the pages when you search for something, there won't be such a problem. In this paper, we will explain how we can achieve this system and compare it with other methods.

One of our concerns is finding related pages on web. Traditional search engines take the requested data from the user, maybe formulated, and give the relevant pages by content matching. If that content exists in the page, it is listed. In contrast, we will show a system that takes an URL of the requested data and gives the related pages about it. For example, if you want to receive daily newspapers published in Turkey, you write `www.hurriyet.com` and you will receive other newspapers pages such as `www.milliyet.com`, `www.radikal.com`. But this approach requires a web site found initially different from traditional search engines. Table 1 shows an example of potential results of searching `www.hurriyet.com` by using Companion Algorithm.

Table 1 - Example Results of Companion Algorithm

INPUT: www.hurriyet.com	
www.milliyet.com	Milliyet Newspaper
www.radikal.com	Radikal Newspaper
www.cnnturk.com	CnnTurk Online Page
www.sabah.com	Sabah Newspaper

We will look at two algorithms, Companion and Cocitation, which use hyperlink structure of the Web to identify related web pages. These algorithms are fast and give high precision. They use Connectivity Server which gives the hyperlink structure of the web. They use the links on the pages and the order of the links they appear. They are not interested in content of pages and patterns.

Companion algorithm is derived from the HITS (Hyperlink Induced Topic Search) algorithm of Kleinberg [1]. Companion extends HITS by exploiting links and their order.

The Cocitation algorithm finds pages that are frequently cocited with the input URL u (it finds

other pages that are pointed to by many other pages that all also point to u).

We will also compare our algorithms with Netscape algorithm which uses connectivity, content and usage information to find related web pages.

Another concern we are looking at during our paper is finding replicated web collections. When you search Java on web, you will see lots of pages similar to each other. If these similar ones are classified in a group such as 'Replicas', one will not waste time looking at similar pages. This approach will also help crawling, ranking, archiving and caching to be performed more effectively. But there are some obstacles detecting replicated pages such as update frequency of pages, mirror partial coverage, different formats and partial crawls. Despite all these difficulties, this method will give us computable similarity measures, improved crawling and reduced clutter from search engines.

In summary, the contributions of this paper are showing new methods and algorithms for web search, comparing them with each other and showing the difficulties and usefulness of these algorithms.

In section 2, we will describe the algorithms and briefly define them, in section 3 results of implementations will be shown, in section 4 comparison of algorithms are listed, in section 5 related work about this topic is summarized, and lastly in section 6 we will explain our conclusions.

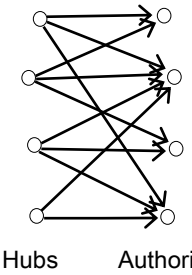
II. THE PROBLEM

A. Companion Algorithm

The Companion Algorithm takes as input a starting URL u and consists of four steps:

1. Build a vicinity graph for u.
2. Contract duplicates and near-duplicates in this graph.
3. Compute edge weights based on host to host connections.
4. Compute a hub score and an authority score for each node in the graph and return the top ranked authority nodes (looks like HITS)

Figure 1 – Hubs and Authorities



A *Hub* is a web page that links to a collection of prominent sites on a common topic. An *Authority* is a page that is linked from a collection of authoritative pages on a broad topic; web page pointed to by hubs. A good authority is a page that is pointed to by many good hubs, while a good hub is a page that points to many good authorities as in Figure 1.

The algorithm does not claim to find all related pages because they may have not been linked. It returns the nodes with ten highest authority scores that are most related to the starting page.

B. Cocitation Algorithm

By examining the siblings of a starting node u in the web graph, we can find the related pages to it. Two nodes are cocited if they have a common parent. Number of common parents of two nodes is their degree of Cocitation. Using this algorithm a new method is used for finding related web pages.

C. Netscape's Approach

Against our algorithms we introduce here, there is also Netscape's approach in order to solve this problem. It is 'What's related?' feature in version 4.06 of Netscape Communicator browser. However, they use connectivity, usage and content information to determine relationships.

D. Similarity Measurements

We define the notion of identical collections. As graphically, if they have same links going to same pages, they are similar. They should contain identical pages and identical link structures. Similarity of web pages by content is another

concern that we investigate during this algorithm. Similarity of link structure is also important factor in clustering web pages. By applying cluster growing algorithm to the collections that are similar, similar web clusters are obtained.

III. RESULTS OF IMPLEMENTATIONS

A. Companion, Cocitation and Netscape Algorithms

18 volunteers are asked to supply at least two URLs for which they want to find the related pages. 69 URLs exist and each of them is given as input to our algorithms. They find 10 for each URL and volunteers are asked to evaluate these pages whether they are related or not. Scoring is like that:

0: Page was not valuable/useful

1: Page was valuable/useful

-: Page could not be accessed

Table 2 - Summary of all answers for the algorithms [1]

Summary of all answers for the algorithms

Algorithm	No. of URLs with answers	No. of answers	No. of dead links
Companion	50	498	42
Cocitation	58	580	62
Netscape	40	364	29

As seen in Table 2, Companion returned 498 answers and 50 of them are related and 42 of them are useless. Cocitation returned 580 links, 58 of them is related and 62 of them are dead. Netscape returned a lower number of links 364 and 40 of them are useful and 29 of them are dead. When we looked at precision ranges, we see that Companion and Cocitation are more precise than Netscape, in other words they give more related number of pages.

B. Exploiting Similar Clusters

To show the usefulness of cluster growing algorithm, it is applied to crawling and searching. In crawling, similar pages are clustered under one page. Widely replicated collections are detected. In web searching, when one searches some topic, there are so many links that are similar to each other. If they are shown to the user under 'Replica' and 'Collection' parts, it will improve the quality of the search.

IV. COMPARISON OF ALGORITHMS

Given a Web page d, the Companion algorithm finds a set of pages related to d by examining its links. Companion is able to return a degree of how related the topic of each page in this set is to the topic of page d. This degree can be used as a similarity measure between d and other pages.

Co-citation was first proposed by Small as a similarity measure between scientific papers. Two papers are co-cited if a third paper has citations to both of them. This reflects the assumption that the author of a scientific paper will cite only papers related to his own work. But now we may use it for web pages' similarities.

Netscape's algorithm is the other one used for detecting related pages. It uses the connectivity, content and usage information to determine related web pages.

Table 3 – Comparison of three algorithms

	Connect	Content	Usage	Precision	Runtime
Companion	√	X	X	Best	Best
Cocitation	√	X	X	Middle	Middle
Netscape	√	√	√	Worse	Worse

As stated in Table 3, these three algorithms are compared with each other according to their precision, runtime and type of information used. Companion and Cocitation use connectivity information on web pages, however Netscape uses all three kinds of information such as connectivity, usage and content of web pages. However, it does not make Netscape more precise or faster as seen from data sets. Companion has detectable performance above Cocitation and also it gives more precise solutions, more related pages.

Precision is evaluated from the results that the volunteers give for the URLs, 1, 0 or none. After

calculating these, companion has a better rank over Cocitation and Netscape. Cocitation has also better precision over Netscape algorithm.

As the number of edges increase, the iteration-running time- increases linearly in Companion algorithm. In Cocitation algorithm, the running time is O (nlogn), n is the number of siblings examined for Cocitation. Also, the running times of Cocitation and companion algorithms are generally correlated, since URLs which have a large number of siblings to consider in the Cocitation algorithm also generally produce a large neighborhood graph for processing in the companion algorithm [1].

Advantages of Companion Algorithm over others

- Weight computation is an intrinsic feature from collection of linked pages
- Provides a densely linked community of related authorities and hubs
- Pure link-based computation once the root set has been assembled, with no further regard to query terms
- Provides surprisingly good search result for a wide range of queries [2].

Disadvantages of Companion Algorithm

- Limit On Narrow Topics
  - Not enough authoritative pages
  - Frequently returns resources for a more general topic
  - Adding a few edges can potentially change scores considerably
- Topic Drifting
  - Appear when hubs discuss multiple topics [2].

Advantages of Similarity Clustering

- *Crawling*: A crawler can finish its job faster if it skips replicated pages and entire collections that it has visited elsewhere.
- *Ranking*: A page that has so many copies will have a higher rank, meaning it is an important page.
- *Archiving*: As the archive can not store the whole web, it may give priority to the known replicated pages and do not archive them again and again, avoiding duplication in archives.
- *Caching*: Knowledge about collection replication is used for storage saving, no cache needed for replicated ones.

Difficulties of Similarity Clustering

- *Update Frequency*: The primary copy of a collection may be constantly updated but mirror copies are updated only daily, weekly or monthly. So this disrupts the similarity of pages, detecting them.
- *Mirror partial coverage*: Only a subset of a collection may be mirrored, and hyperlinks point to uncopied portions in another mirror site, or at the primary.
- *Different formats*: Documents in a collection may not appear as original ones. Some are in the format of HTML, word or postscript. It makes difficult to cluster them as similar ones.
- *Partial crawls*: We need to identify replicated collections from a snapshot of the web that has been crawled or cached at one site, not examining the original data.
- *Similarity of Link Structure:* They may involve different links and structures. For example, lack ness of one to one mappings, existence of break points, different collection sizes may exist.

Contributions of Similarity Clustering

- *Computable similarity measures:* Efficient heuristic algorithms are identified to detect replicated collections.
- *Improved crawling:* Web crawling is improved with the help of this method, by clustering replicated ones.
- *Reduce clutter from search engines:* Search engines' results are shown in a clearer way, Replica and Collection links.

Discovering Web communities on the web

There are two different web communities, explicit and implicit ones. Explicit communities are easy to identify such as Yahoo. Implicit ones are detected

with the help of web graph. To detect the similarity, if there is a hyperlink from one to other, they are said to be similar. The other methods that come from bibliometrics are Cocitation and bibliographic coupling. All of them can discover meaningful communities. But these methods are very expensive to the whole World Wide Web with billions of web pages. However, there is a cheaper method found by Ravi Kumar, Prabhakar Raghavan, Sridhar Rajagopalan, and Andrew Tomkins by IBM Almaden Research Center. It is communities trawling (CT), not included in this paper.

V. RELATED WORK

Many researchers are working on clustering techniques on web data. One of them is Chakrabarti [3] that use the links and their order to categorize web pages and they show that the links that are near a given link in page order frequently point to pages on the same topic.

Some authors suggested using Cocitation and other forms of connectivity to identify related web pages. Spertus [4] observed that Cocitation can detect that two pages are related. Pitkow and Pirolli [5] cluster web pages according to Cocitation algorithms.

Our companion algorithm is extended from HITS defined by Kleinberg. He defined it as a way of using connectivity structure to identify the most authoritative sources of information on a particular topic, where the topic was defined by the combined link structure of a large number of starting nodes on the topic [1].

Previous link-based research for identifying collections of related pages includes bibliometrics methods such as co-citation and bibliographic coupling [6], the Page Rank algorithm [7], the HITS algorithm [8], bipartite sub graph identification [9], Spreading Activation Energy (SAE) [10], and others.

Co-citation, bibliographic coupling, and bipartite sub graph identification are localized approaches in the sense that they seek to identify well-defined graph structures that exist inside of a narrow region of the web graph. Page Rank, HITS, and SAE, are more global since they work by iteratively propagating weights through a significant portion of the web graph. The weights reflect an estimate of page importance (Page Rank), how authoritative or hub-like a web page is (HITS), or how "closes" a candidate page is to a starting region (SAE). Both HITS and Page Rank are relatively insensitive to their choice of parameters, unlike spreading activation energy, which yields results that are extremely sensitive to the choice of parameters [11].

VI. CONCLUSION AND FUTURE WORK

During this paper we tried to analyze finding the related web pages and finding the replicated ones by using different algorithms. Companion which is an extension of HITS model performed the best result in finding related pages. Cocitation is also used in detecting related web pages. We compared these algorithms with Netscape algorithm, and resulted that Companion and Cocitation are better than Netscape's algorithm. These algorithms may extend to take more than one input URL. Another good work will be not taking the URL, taking the content that is searched as input.

We also looked at finding replicated web pages by using cluster growing algorithm. We resulted that by doing such a clustering, search engine result representation and crawling will improve. We achieved it by comparing the links and their similarities in web pages.

We compare these algorithms by information usage, performance and precision and concluded that Companion is better than others.

REFERENCES

[1] Jeffrey Dean, Monika R. Henzinger, *Finding related pages in the World Wide Web,* 1999

[2] Sanjay Kumar Madria, Department of Computer Science, University of Missouri-Rolla, MO 65401, *Web Mining : A Bird's Eye View,* 2008

[3] S. Chakrabarti, B. Dom and P. Indyk, *Enhanced hypertext categorization using hyperlinks, in:*

Proc. of the ACM SIGMOD International Conference on Management of Data, pp. 307–318, 1998.

[4] E. Spertus, ParaSite: *Mining structural information on the Web, in: Proc. of the Sixth International World Wide Web Conference*, pp. 587–595, Santa Clara, CA, April 1997.

[5] J. Pitkow and P. Pirolli, *Life, death, and lawfulness on the electronic frontier, in: Proc. of the Conference on Human Factors in Computing Systems (CHI 97)*, pp. 383–390, March 1997.

[6] E. Garfield. *Citation Indexing: Its Theory and Application in Science. Wiley, New York*, 1979.

[7] S. Brin and L. Page. *The anatomy of a large-scale hypertextual Web search engine. In Proc. 7th Int. World Wide Web Conf.*, 1998.

[8] Jon M. Kleinberg. *Authoritative sources in a hyperlinked environment. In Proceedings of the Ninth Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 668–677, 1998.

[9] Ravi Kumar, Prabhakar Raghavan, Sridhar Rajagopalan, and Andrew Tomkins. *Trawling the web for emerging cyber-communities*. In Proc. 8th Int. World Wide Web Conf., 1999.

[10] Peter Pirolli, James Pitkow, and Ramana Rao. *Silk from a sow's ear: Extracting usable structures from the web.* In Proc. ACM Conf. Human Factors in Computing Systems, CHI. ACM Press, 1996.

[11] Gary William Flake, Steve Lawrence, C. Lee Giles, Frans M. Coetzee, NEC Research Institute, 4 Independence Way, Princeton, NJ 08540, *Self-Organization and Identification of Web Communities.*

[12] Junghoo Cho, Narayanan Shivakumar, Hector Garcia-Molina Department of Computer Science Stanford, CA 94305, *Finding replicated web collections.*

[13] http://www.cs.uic.edu/~liub/WebMiningBook.html

[14] http://64.233.183.104/search?q=cache:htaSlQrZoo8J:eprints.cs.vt.edu/archive/00000693/01/GP5.pdf+comparison+cocitation+and+companion&hl=en&ct=clnk&cd=12

[15] http://linkanalysis.wlv.ac.uk/5.htm