Scale the Active Influence Based Investigation Using Materialized Sub Graphs

1. Amjan. Shaik , *CSE, Ellenki College Of Engineering and Technology(ECET), Patelguda,Hyderabad,India.*
2. Nazeer. Shaik, *-CSE,Moghal College Of Engineering and Technology(MCET),Bandlaguda,Hyderabad,India.*
3. Amtul Mubeena, *Jazan University (Kingdom of Saudi Arabia).*
4. S.V.Achuta Rao , *CSE&IT,DJR Institute of Engineering and Technology(DJRIET),Vijawada,India.*
5. K.Vikram, *CSE,City Womens College of Engineering and Technology(CWCET),Bandlaguda,Hyderabad,India.*

Abstract

The development of Information Technology has generated large amount of databases and huge data in various areas. The research in databases and information technology has given rise to an approach to store and manipulate this precious data for further decision making. Data mining is a process of extraction of useful information and patterns from huge data. It is also called as knowledge discovery process, knowledge mining from data, knowledge extraction or data /pattern analysis. In this paper active authority-based keyword search algorithms, such as ObjectRank and personalized PageRank, leverage semantic link information to provide high quality, high recall search in databases, and the web. Conceptually, these algorithms require a querytime PageRank-style iterative computation over the full graph. This computation is excessively expensive for large graphs and not realistic at query time. Alternatively, building an index of precomputed results for some or all keywords involves very expensive preprocessing. Now we demonstrated BinRank, a system that approximates ObjectRank results by utilizing a hybrid approach inspired by materialized views in traditional query processing. We materialize a number of relatively small subsets of the data graph in such a way that any keyword query can be answered by running ObjectRank on only one of the subgraphs. BinRank generates the subgraphs by partitioning all the terms in the corpus based on their co-occurrence, executing ObjectRank for each partition using the terms to generate a set of random walk starting points, and keeping only those objects that receive non-negligible scores. The perception is that a subgraph that contains all objects and links relevant to a set of related terms should have all the information needed to rank objects with respect to one of these terms. We demonstrate that BinRank can achieve subsecond query execution time on the english wikipedia data set, while producing high-quality search results that closely approximate the results of ObjectRank on the original graph. The Wikipedia link graph contains about 110 edges, which is at least two orders of magnitude larger than what prior state of the art dynamic authority-based search systems have been able to demonstrate. Our experimental evaluation investigates the trade-off between query execution time, quality of the results, and storage requirements of BinRank.

Keywords: Scaling, Subgraphs, Binbank, ObjectRank, PageRank.

1.INTRODUCTION

Data mining is a relatively new technology that has not fully matured. Despite this, there are a number of industries that are already using it on a regular basis. Some of these rganizations include retail stores, hospitals, banks, and insurance companies. Many of these organizations are combining data mining with such things as statistics, pattern recognition, and other important tools. Data mining can be used to find patterns and connections that would otherwise be difficult to find. This technology is popular with many businesses because it allows them to learn more about their customers and make smart marketing decisions. Here is overview of business problems and solutions found using data mining technology. Information computing in various web domains is broadly extracting the web objects of unstructured nature like text objects that convince information need from within large collections using document-level ranking and therefore the structured information about real-world objects which is embedded in static web pages. Online databases exist on the web in huge amounts which are of unstructured nature. Unstructured data refers to the data which does not have clear, semantically obvious structure [7]. In other words information computing constitutes process of searching, recovering, and understanding information, from huge amounts of stored data. The information from the web can be retrieved by implementing searching techniques as Keyword based Searching, Concept-based Searching, Hybrid Search, and Knowledge Base Search. In case of object level information computing, domain based search is required. Every commercial information retrieval systems try to facilitate a user's access to information that is relevant to his information needs. This paper highlights ranking problem for domain based information retrieval, which states that every owner of the document wants to improve ranking of its document for that it can do many manipulations on its document like increasing number of links to the page by the dummy pages [1]. Object based information computing maintain the integrity of the search results based upon various lexicons. As the web contains the contradictions and hypothesis on a huge scale, therefore finding the relevant information using search engines is a tedious job. With the help of object level ranking , various objects on a domain independent of the query that describes the relative trust of the web page can be prioritized. The object rank of a page depends upon various factors associated with the web object In this paper, we discussed about a BinRank system that employs a hybrid approach where query time can be traded off for preprocessing time and storage. BinRank closely approximates ObjectRank scores by running the same ObjectRank algorithm on a small subgraph, instead of the full data graph. Data mining has importance regarding finding the patterns, forecasting, discovery of knowledge etc., in different business domains. Data mining techniques and algorithms such as classification, clustering etc., helps in finding the patterns to decide upon the future trends in businesses to grow. Data mining has wide application domain almost in every industry where the data is generated that's why data mining is considered one of the most important frontiers in database and information systems and one of the most promising interdisciplinary developments in Information Technology.

1.1 Problem Definition

The idea of approximating ObjectRank by using Materialized SubGraphs (MSGs), which can be precomputed offline to support online querying for a specific query workload, or the entire dictionary. Use of ObjectRank itself to generate MSGs for bins of terms. A greedy algorithm that minimizes the number of bins by clustering terms with similar posting lists. Extensive experimental evaluation on the Wikipedia data set that supports our performance and search quality claims. The evaluation demonstrates superiority of BinRank over other state-of-the-art approximation algorithms.

1.1.1 Existing Methodology

PageRank algorithm utilizes the Web graph link structure to assign global importance to Web pages. It works by modeling the behavior of a random Web surfer who starts at a random web page and follows outgoing links with uniform probability. The PageRank score is independent of a keyword query. Personalized PageRank (PPR) for web graph data sets and ObjectRank for graph-modeled databases results. Therefore, the issue of scalability of PPR has attracted a lot of attention. ObjectRank extends Personalized PageRank to perform keyword search in databases. ObjectRank uses a query term posting list as a set of random walk starting points and conducts the walk on the instance graph of the database.

1.1.2 Proposed Methodology

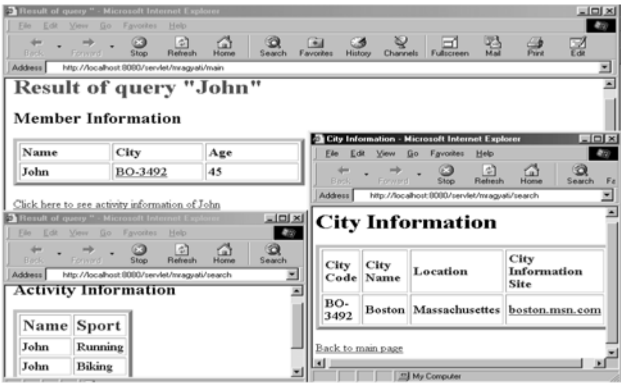
BinRank system that employs a hybrid approach where query time can be traded off for preprocessing time and storage. BinRank closely approximates ObjectRank scores by running the same ObjectRank algorithm on a small subgraph, instead of the full data graph. BinRank query execution easily scales to large clusters by distributing the subgraphs between the nodes of the cluster. We are proposing the BinRank algorithm for the trade time of search. This alogorithm solves the time consuming problem in query execution. Time will be reduced because of cache storage and redundant query handling method.

2. RESEARCH BACKGROUND

The web, through many search engine sites, has popularized the keyword-based search paradigm, where a user can specify a string of keywords and expect to retrieve relevant documents, possibly ranked by their relevance to the query. Since a lot of information is stored in databases (and not as HTML documents), it is important to provide a similar search paradigm for databases, where users can query a database without knowing the database schema and database query languages such as SQL.

The problem of extracting information scattered over the web is well known. Web search tools usually extract information pulled out from semi-structured documents. However, as information over web increasingly comes out of a database, it is crucial to provide searching of databases on the web directly. Since these databases serve applications that also update the data, it is not feasible to convert the database contents into a searchable (set of) HTML documents for use by search engines. A free-form search utility is required which can construct SQL queries from the keyword-based query (using metadata and other information from the database as transparently as possible) and present search results in a structured form. Furthermore, such a utility should be general so that it can be used with any database.

Google is a prototype of a large-scale search engine that makes heavy use of the structure present in hypertext[1]. Google is designed to crawl and index the web efficiently and produce much more satisfying search results than existing systems. Link Analysis Ranking [16]emphasize that hyperlink structures are used to determine the relative authority of a web page and produce improved algorithms for the ranking of search results. The prototype with a full text and hyperlink database of web pages is available at [8]. In the current era there is much concern in using random graph models for the web. The Random Surfer model [9] and the Page Rank-based selection model[11] are described as two major models [10]. Page Rank-based selection model tries to capture the effect that the search engines have on the growth of the web by adding new links according to Page Rank. The Page Rank algorithm is used in the Google search engine [12] for ranking search results. PageRank is a link analysis algorithm used by the Google Internet search engine that assigns a numerical weighting to each element of a hyperlinked set of documents, such as the World Wide Web (WWW), with the purpose of "measuring" its relative importance within the set. Google is designed to be a scalable search engine with primary goal to provide high quality search results over a rapidly growing WWW[18].


Result of query "" - Microsoft Internet Explorer
File Edit View Go Favorites Help
Back Forward Stop Refresh Home Search Favorites History Channels Fullscreen Mail Print Edit
Address

http://localhost:8080/servlet/mragyati/main

Result of query "John"

Member Information

Name	City	Age
John	BO-3492	45

Click here to see activity information of John

City Information - Microsoft Internet Explorer
File Edit View Go Favorites Help
Back Forward Stop Refresh Home Search Fa
Address

http://localhost:8080/servlet/mragyati/search

City Information

City Code	City Name	Location	City Information Site
BO-3492	Boston	Massachusettes	boston.msn.com

Back to main page

My Computer
Result of query "" - Microsoft Internet Explorer
File Edit View Go Favorites Help
Back Forward Stop Refresh Home Search
Address

http://localhost:8080/servlet/mragyati/search

Activity Information

Name	Sport
John	Running
John	Biking

Figure 1: Keyword-based Searching in Databases

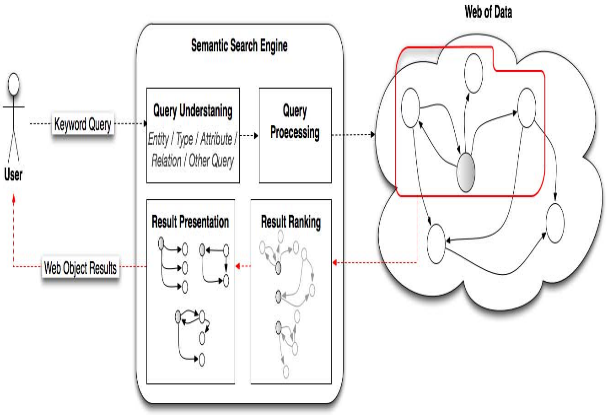


Figure 2: Overview of the semantic search process. A keyword query is used to identify relevant web objects. Relevant objects are ranked, and for each object a set of connected objects is selected as the result.

2.1 Page Rank

The PageRank algorithm utilizes the web graph link structure to assign global importance to web pages. It works by modeling the behavior of a random web surfer who starts at a random web page and follows outgoing links with uniform probability. The PageRank score is independent of a keyword query. Recently, dynamic versions of the PageRank algorithm have become popular, they are characterized by a query-specific choice of the random walk starting points.

2.2 Personalized Page Rank

In particular, two algorithms have got a lot of attention: Personalized Page Rank (PPR) for web graph data sets and ObjectRank for graph-modeled databases. PPR is a modification of PageRank that performs search personalized on a preference set that contains web pages that a user likes. For a given preference set, PPR performs a very expensive fixpoint iterative computation over the entire web graph, while it generates personalized search results. Therefore, the issue of scalability of PPR has attracted a lot of attention. ObjectRank extends Personalized PageRank to perform keyword search in databases. ObjectRank uses a query term posting list as a set of random walk starting points and conducts the walk on the instance graph of the database. The resulting system is well suited for high recall search, which exploits different semantic connection paths between objects in highly heterogeneous data sets.

2.3 Object Rank

ObjectRank has successfully been applied to databases that have social networking components, such as bibliographic data and collaborative product design. However, ObjectRank suffers from the same scalability issues as Personalized PageRank, as it requires multiple iterations over all nodes and links of the entire database graph. The original ObjectRank system has two modes: online and offline. The online mode runs the ranking algorithm once the query is received, which takes too long on large graphs. For example, on a graph of articles of english wikipedia1 with 3.2 million nodes and 110 million links, even a fully optimized in-memory implementation of ObjectRank takes 20-50 seconds to run. In the offline mode, ObjectRank precomputes top-k results for a query workload in advance. This precomputation is very expensive and requires a lot of storage space for precomputed results. Moreover, this approach is not feasible for all terms outside the query workload that a user may search for all terms in the data set dictionary. For example, on the same wikipedia data set, the full dictionary precomputation would take about a CPU-year.

3. MODULES

3.1 User Registration:

We are providing the facility to register new users. If anyone wants use our application, they should become a member of our application. To getting the membership login the users should made registration with our application. In registration we will get all the details about the users and it will be stored in a database to create membership.

3.2 Authentication Module:

This module provides the authentication to the users who are using our application. In this module we are providing the registration for new users and login for existing users.

3.3 Search Query Submission:

Users query will be submitted in this module. Users can search any kind of things in our application when we connect with Internet. Users query will be processed based on their submission, and then it will produce the appropriate result. Result will be produced based on our algorithm.

3.4 Index Creation:

Index is something like the count of search and result which we produced while searching. Based on the index we will create the rank for the results, such like pages or corresponding websites. This will be maintained in background for future use like cache memory. By the way we are creating the index for speed up the search efficient and fast with the help of implementing BinRank algorithm.

3.5 BinRank Algorithm Implementation:

We generate an MSG for every bin based on the intuition that a subgraph that contains all objects and links relevant to a set of related terms should have all the information needed to rank objects with respect to one of these terms. Based on the index creation we need to generate the results for the users query. BinRank algorithm will use the indexing and ranking techniques to produce the efficient results in short time.

3.6 Graph based on Rank:

Graph will be generated based on the users queries submitted. This graph will represent the user search keyword, number of websites produced for their search, how many times that websites occurred in the search result and the Rank for websites based on the user clicks. User may search the same keyword again and again, so result may also produce as same URLs. At that user will click some of the URLs; based on their clicks the Rank will be calculated. Based on the Number of times URL occurrence, Rank and Keyword the Graph will generate.

4. DESIGN AND VALIDATIONS

4.1 Database Design

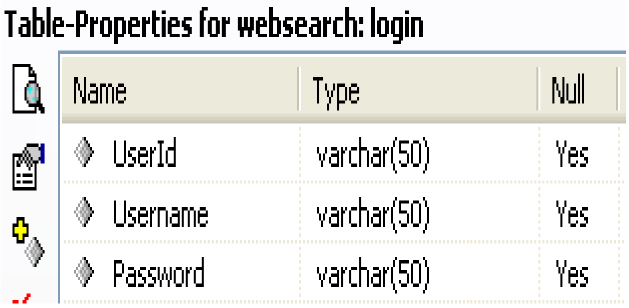


Table-Properties for websearch: login

Name	Type	Null
UserId	varchar(50)	Yes
Username	varchar(50)	Yes
Password	varchar(50)	Yes

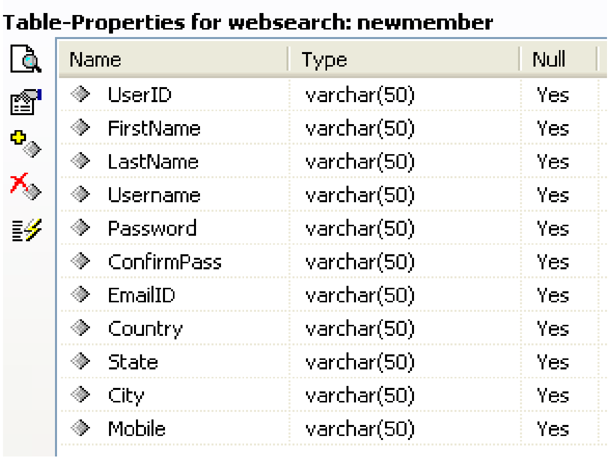


Table-Properties for websearch: newmember

Name	Type	Null
UserID	varchar(50)	Yes
FirstName	varchar(50)	Yes
LastName	varchar(50)	Yes
Username	varchar(50)	Yes
Password	varchar(50)	Yes
ConfirmPass	varchar(50)	Yes
EmailID	varchar(50)	Yes
Country	varchar(50)	Yes
State	varchar(50)	Yes
City	varchar(50)	Yes
Mobile	varchar(50)	Yes

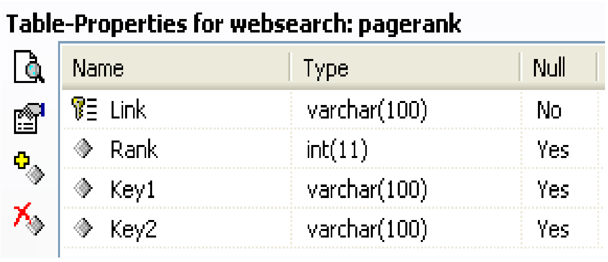


Table-Properties for websearch: pagerank

Name	Type	Null
Link	varchar(100)	No
Rank	int(11)	Yes
Key1	varchar(100)	Yes
Key2	varchar(100)	Yes

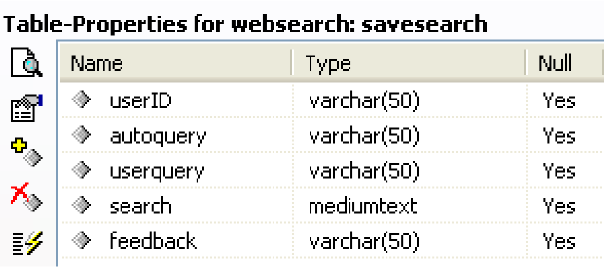


Table-Properties for websearch: savesearch

Name	Type	Null
userID	varchar(50)	Yes
autoquery	varchar(50)	Yes
userquery	varchar(50)	Yes
search	mediumtext	Yes
feedback	varchar(50)	Yes

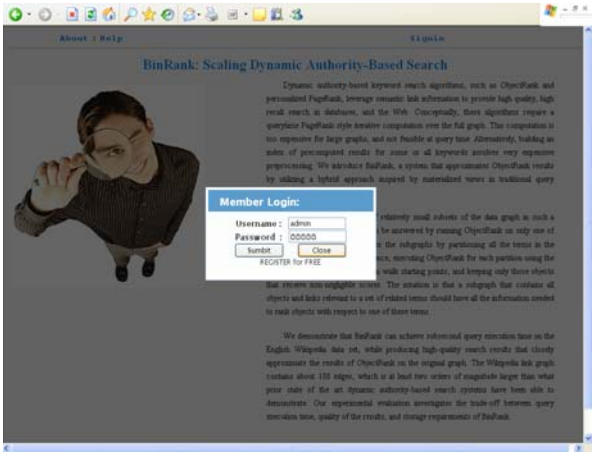
4.2 Modules Validations

Module 1:
Input : User details registration
Output : Authorized user to access application
Module 2:
Input : Username and password
Output : Access to application
Module 3:
Input : Search query
Output : Query to database
Module 4:
Input : Query result from database
Output : Results to user
Module 5:
Input : Binrank implementation
Output : Binrank for output
Module 6:
Input : Binrank
Output : Graph based on binrank

4.3 Sample Source Code

```
package org.websearch.Action;
public class Search
{
    public static void main(String args[])
   {
     System.out.println("Search File");
     long start = System.nanoTime();
     System.out.println("Start: " + start+" nano
seconds");
      try {
      URL url = new
URL("http://www.google.com/search?q=example");
      URLConnection conn = url.openConnection();
conn.setRequestProperty("User-Agent","Mozilla/5.0 (X11;
U; Linux x86_64; en-GB; rv:1.8.1.6) Gecko/20070723
Iceweasel/2.0.0.6 (Debian-2.0.0.6-0etch1)");
BufferedReader in = new BufferedReader(
new InputStreamReader(conn.getInputStream())
     );
          String str;
          while ((str = in.readLine()) != null) {
          System.out.println(str);
            }
                  in.close();
             }
             catch (MalformedURLException e) {}
             catch (IOException e) {}
      long end = System.nanoTime();
System.out.println("End : " + end+" nano seconds");
long elapsedTime = end - start;
System.out.println("The process took approximately: "
 + elapsedTime + " nano seconds");
   }
}
```

5. EXPERIMENTAL RESULTS



BinRank: Scaling Dynamic Authority-Based Search

Dynamic authority-based keyword search algorithms, such as ObjectRank and personalized PageRank, leverage semantic link information to provide high quality, high recall search in databases, and the Web. Conceptually, these algorithms require a query-time PageRank-style iterative computation over the full graph. This computation is too expensive for large graphs, and not feasible at query time. Alternatively, building an index of precomputed results for some or all keywords involves very expensive preprocessing. We introduce BinRank, a system that approximates ObjectRank results by utilizing a hybrid approach inspired by materialized views in traditional query

relatively small subsets of the data graph in such a
be answered by running ObjectRank on only one of
the subgraphs by partitioning all the terms in the
ce, executing ObjectRank for each partition using the
walk starting points, and keeping only those objects
that receive non-negligible scores. The intuition is that a subgraph that contains all objects and links relevant to a set of related terms should have all the information needed to rank objects with respect to one of these terms.

We demonstrate that BinRank can achieve subsecond query execution time on the English Wikipedia data set, while producing high-quality search results that closely approximate the results of ObjectRank on the original graph. The Wikipedia link graph contains about 108 edges, which is at least two orders of magnitude larger than what prior state of the art dynamic authority-based search systems have been able to demonstrate. Our experimental evaluation investigates the trade-off between query execution time, quality of the results, and storage requirements of BinRank.

Member Login:
Username : admin
Password : ●●●●●

Figure 1: User authentication

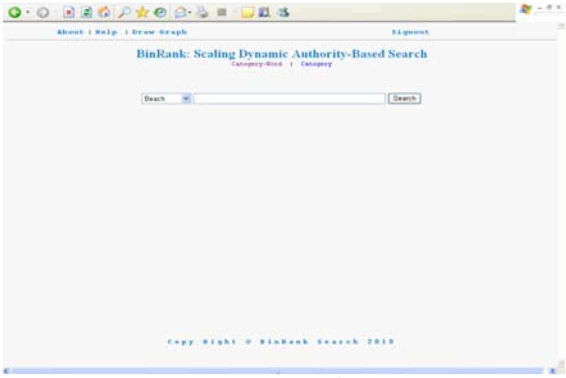
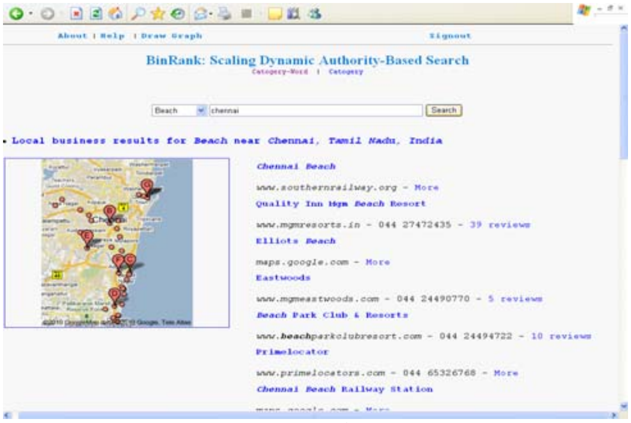

About | Help | Draw Graph
Signout
BinRank: Scaling Dynamic Authority-Based Search
Category-Word | Category
Beach
Search
Copy Right © BinRank Search 2010

Figure 2 : Search for a website

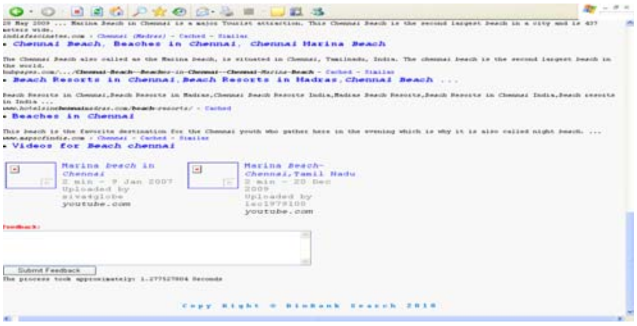

About | Help | Draw Graph
Signout
BinRank: Scaling Dynamic Authority-Based Search
Category-Word | Category
Search

Beach chennai

- Local business results for Beach near Chennai, Tamil Nadu, India

Chennai Beach
www.southernrailway.org - More
Quality Inn Mgm Beach Resort
www.mgmresorts.in - 044 27472435 - 39 reviews
Elliots Beach
maps.google.com - More
Eastwoods
www.mgmeastwoods.com - 044 24490770 - 5 reviews
Beach Park Club & Resorts
www.beachparkclubresort.com - 044 24494722 - 10 reviews
Primelocator
www.primelocators.com - 044 65326768 - More
Chennai Beach Railway Station
maps.google.com - More

Figure 3: Results for searched website



29 May 2009 ... Marina Beach in Chennai is a major Tourist attraction. This Chennai Beach is the second largest beach in a city and is 437 meters wide.
indiafascinates.com › Chennai (Madras) - Cached - Similar

- Chennai Beach, Beaches in Chennai, Chennai Marina Beach

The Chennai Beach also called as the Marina beach, is situated in Chennai, Tamilnadu, India. The chennai beach is the second largest beach in the world.
hubpages.com/.../Chennai-Beach--Beaches-in-Chennai--Chennai-Marina-Beach - Cached - Similar

- Beach Resorts in Chennai,Beach Resorts in Madras,Chennai Beach ...

Beach Resorts in Chennai,Beach Resorts in Madras,Chennai Beach Resorts India,Madras Beach Resorts,Beach Resorts in Chennai India,Beach resorts in India ...
www.hotelsinchennaimadras.com/beach-resorts/ - Cached

- Beaches in Chennai

This beach is the favorite destination for the Chennai youth who gather here in the evening which is why it is also called night beach. ...
www.mapsofindia.com › Chennai - Cached - Similar

- Videos for Beach chennai

Marina beach in Chennai
2 min - 9 Jan 2007
Uploaded by sivatglobe
youtube.com

Marina Beach-Chennai,Tamil Nadu
2 min - 20 Dec 2009
Uploaded by lsc1979100
youtube.com

Feedback:

The process took approximately: 1.277527804 Seconds

Figure 4: Timing validation for searched website

ACKNOWLEDGEMENTS

The authors thankful to the Ellenki College of Engineering and Technology-Research and Development Cell for collecting the data and preparing the paper.

CONCLUSIONS

In this paper, we discussed about BinRank as a practical solution for scalable dynamic authority-based ranking. It is based on partitioning and approximation using a number of materialized subgraphs. We showed that our tunable system offers a nice trade-off between query time and preprocessing cost. We initiate a greedy algorithm that groups co-occurring terms into a number of bins for which we compute materialized subgraphs. Note that the number of bins is much less than the number of terms. The materialized subgraphs are computed offline by using ObjectRank itself. The intuition behind the approach is that a subgraph that contains all objects and links relevant to a set of related terms should have all the information needed to rank objects with respect to one of these terms. Our extensive experimental evaluation confirms this intuition. For future work, we want to study the impact of other keyword relevance measures, besides term co-occurrence, such as thesaurus or ontologies, on the performance of BinRank. By increasing the relevance of keywords in a bin, we expect the quality of materialized subgraphs, thus the top-k quality and the query time can be improved. We also want to study better solutions for queries whose random surfer starting points are provided by Boolean conditions. And ultimately, although our system is tunable, the configuration of our system ranging from number of bins, size of bins, and tuning of the ObjectRank algorithm itself (edge weights and thresholds) is quite challenging, and a wizard to aid users is desirable.

REFERENCES

[1] S. Agrawal, S. Chaudhuri, and G. Das. DBXplorer: A System For Keyword-Based Search Over Relational Databases.*ICDE*, 2002.

[2] A. Balmin, V. Hristidis, and Y. Papakonstantinou. ObjectRank: Authority-Based Keyword Search in Databases (extended version). *UCSD Technical Report*, 2004.

[3] G. Bhalotia, C. Nakhey, A. Hulgeri, S. Chakrabarti, and S. Sudarshan. Keyword Searching and Browsing in Databases using BANKS. *ICDE*, 2002.

[4] Y. Chen, Q. Gan, and T. Suel. I/O-efficient techniques for computing PageRank. *CIKM*, 2002.

[5] R. Fagin, A. Lotem, and M. Naor. Optimal Aggregation Algorithms for Middleware. *ACM PODS*, 2001.

[6] X. Gu, K. Nahrstedt, W. Yuan, D. Wichadakul, and D. Xu.An XML-based Quality of Service Enabling Language for the Web. *Journal of Visual Languages and Computing 13(1): 61-95*, 2002.

[7] L. Guo, F. Shao, C. Botev, and J. Shanmugasundaram. XRANK: Ranked Keyword Search over XML Documents. *ACM SIGMOD*, 2003.

[8] T. Haveliwala. Efficient computation of PageRank.*Technical report, Stanford University (http://www.stanford.edu/ taherh/papers/efficient-pr.pdf)*,1999.

[9] V. Hristidis and Y. Papakonstantinou. DISCOVER: Keyword Search in Relational Databases. *VLDB*, 2002.

[10] V. Hristidis, Y. Papakonstantinou, and A. Balmin. Keyword Proximity Search on XML Graphs. *ICDE*, 2003.

[11] M. Richardson and P. Domingos. The Intelligent Surfer:Probabilistic Combination of Link and Content Information in PageRank. *Advances in Neural Information Processing Systems 14, MIT Press*, 2002.

[12] G. Salton. Automatic Text Processing: The Transformation,Analysis, and Retrieval of Information by Computer. *Addison Wesley*, 1989.

[13] J. Savoy. Bayesian inference networks and spreading activation in hypertext systems. *Information Processing and Management*, 28(3):389–406, 1992.575.

[14]. G. Bhalotia, 'Keyword searching in Databases using BANKS', B.Tech. project report, I.I.T. Bombay, April 2001.

[15]. M. Carey, L. Haas, V. Maganty and J. Williams, 'Pesto : An Integrated Query/Browser for Object Databases', Proc. VLDB '96, Bombay, p. 203-214, 1996.

[16] F. Gey et.al., 'Advanced Search Technologies for Unfamiliar Metadata', Third IEEE Metadata Conference (Meta-Data 99), April 6-7, 1999, Bethesda, Maryland

[17] R. Goldman, N. Shivakumar, S. Venkatasubramanian, H. Garcia-Molina, 'Proximity Search in Databases', Prof. VLDB '98, pp.26-37, 1998.

[18].U.Masermann and G. Vossen, 'Design and Implementation of a Novel Approach to keyword Seraching in Relational Databases', Prof. Of ADBIS-DASFAA Symp. On Advances in Databases & Information Systems, Sept. 5-8, 2000, Prague, Czech Republic

[19]. U. Masermann and G. Vossen, 'SISQL : Schema Independent Database Querying (on and off the Web)', Proc. Of IDEAS2000, Sept. 18-20, 2000, Yokohoma, Japan

[20]. J.C. Shafer and R. Agrawal, 'Continuous Querying in Database Centric Web Applications', Proc. Of 9th Intl. WWW Conference, Elsivier Science, May 15-19, 2000, pp. 519-531

ABOUT THE AUTHORS

Amjan Shaik is working as a Professor and Head, Department of Computer Science and Engineering at Ellenki College of Engineering and Technology (ECET), Hyderabad, India. He has received M.Tech. (Computer Science and Technology) from Andhra University. Presently, he is a Research Scholar of JNTUH Hyderabad. He has been published and presented 34 Research and Technical papers in International Journals , International Conferences and National Conferences. His main research interests are Software Engineering, Data Mining, Software Metrics, Software Quality and Object Oriented Design.

Nazeer. Shaik is working as an Asst Professor, Department of Computer Science and Engineering at Moghal College of Engineering and Technology (MCET), Hyderabad, India. He has received M.Tech. (Computer Science and Engineering) from Bharath University, Chennai..He has presented number of technical papers in International and National Conferences. His research interests are Software Engineering, Mobile Computing, and Information Security.

Amtul Mubeena is working as a Lecturer in Jazan University (KSA). She has received M.Tech. (Information Technology) from ASTRA, Affiliated to JNTUH Hyderabad,India. She has presented number of Technical papers in International and National Conferences. Her research interests are Data Mining, Information Security and Software Engineering.

S.V. Achuta Rao is working as a Professor and Head, Department of CSE and IT at DJR Institute of Engineering and Technology (DJRIET), Vijayawada, India. He has received M.Tech. (Computer Science and Engineering) from JNTU, Kakinada, India. Presently, he is a Research Scholar of Rayalaseema University (RU), Kurnool, India. He has been published and presented good number of Research and technical papers in International and National Conferences. His main research interests are Data Mining, Networking, Image Processing, Software Engineering and Software Metrics.

K.Vikram is working as a Professor and Head, Department of Computer Science and Engineering at City Womens College of Engineering and Technology ,Hyderabad, India. He has received M.E from Anna University, Chennai,India. Presently he is a Research Scholar in JNTUH Hyderabad. He has published and presented good number of Technical and Research Papers in National and International Conferences. His research Interests are Software Testing, Image Processing , Computer Organization and Information Security.