DSphere: A Source-Centric Approach to Crawling, Indexing and Searching the World Wide Web

Bhuvan Bamba, Ling Liu, James Caverlee, Vaibhav Padliya, Mudhakar Srivatsa,
Tushar Bansal, Mahesh Palekar, Joseph Patrao, Suiyang Li and Aameek Singh
College of Computing, Georgia Institute of Technology, Atlanta, Georgia - 30332
{bhuvan, lingliu, caverlee, vaibhav, mudhakar, tusharb, mpalekar, jpatrao, suiyang, aameek}@cc.gatech.edu

Abstract

We describe DSPHERE[1] *– a decentralized system for crawling, indexing, searching and ranking of documents in the World Wide Web. Unlike most of the existing search technologies that depend heavily on a page-centric view of the Web, we advocate a source-centric view of the Web and propose a decentralized architecture for crawling, indexing and searching the Web in a distributed source-specific fashion. A fully decentralized crawler is developed to crawl the World Wide Web where each peer is assigned the responsibility of crawling a specific set of documents referred to as a* ***source collection****. Link analysis techniques are used for ranking documents. Traditional link analysis techniques suffer from problems like slow refresh rate and vulnerabilities to Web Spam, to counter which, we propose a* ***source-based*** *link analysis algorithm which computes fast and accurate ranking scores for all crawled documents.*

1. Introduction

Most current search systems manage Web crawlers with a centralized client-server model in which the assignment of crawling jobs is managed by a centralized system using centralized repositories. Such systems suffer from a number of problems, including link congestion, low fault tolerance, low scalability and expensive administration. DSPHERE performs crawling, indexing, searching and ranking using a fully decentralized computing architecture. Each peer in the DSphere network is responsible for crawling a specific set of documents, referred to as the **source collection**, and maintaining an index over its crawled collections to facilitate a full-text keyword search over these collections. A source collection may be defined as a set of documents belonging to a particular domain. For example, a peer may be assigned the responsibility of crawling all or a subset of the documents in the *www.cc.gatech.edu* domain. The DSPHERE crawler uses geographical proximity of peers to Web resources for faster crawling and distributes the crawling task among a network of peers which exchange information amongst themselves to avoid URL duplication and content duplication. We have developed two different versions of the crawler, the first [5] uses the structured P2P approach based on a Distributed Hash Table protocol and the other [3] uses the unstructured P2P approach based on the Gnutella protocol. The design of our decentralized crawlers aims at providing extremely fast crawling functionality with built-in scalability and fault tolerance. For ranking purposes, the traditional TF-IDF based measures are supplemented by link analysis based techniques. Several search engines have successfully deployed such techniques, as exemplified by the popularity of Google's PageRank algorithm [4]. However, PageRank and other existing link-based approaches suffer from a number of known problems, such as slow update time, vulnerability to Web Spam, and lack of support for higher levels of abstraction of the Web beyond the flat page-based view. Our *DSphere* approach enhances the link-based ranking mechanisms through collection-based link analysis [1]. Motivated by the strong locality-based nature of Web links, we view the Web graph as a set of interlinked *collections* on top of the interlinked Web pages. The source-based approach significantly speeds up the update time of link analysis based scores and provides higher resilience to a number of known Web Spam attacks [2].

2. Decentralized Crawler

A decentralized crawler consists of a network of peers located in geographically disparate regions across the World. Each peer is responsible for crawling a certain portion of the Web based on the geographical proximity between peers and the collections to be crawled. We have developed two different versions of the crawler, one using a structured P2P topology based on a Distributed Hash Table protocol (called Apoidea) and another using an unstructured P2P system based on Gnutella-like topology (called PeerCrawl).

Due to space constraints, we describe only the most important feature of the DSPHERE decentralized crawler – the *division of labor* among the peers. Interested readers may refer to [5, 3] for more detailed descriptions. The structured decentralized crawler [5] uses the DHT protocol for distributing the World Wide Web space among all peers in the network.

[1] DSPHERE stands for *decentralized information sphere*

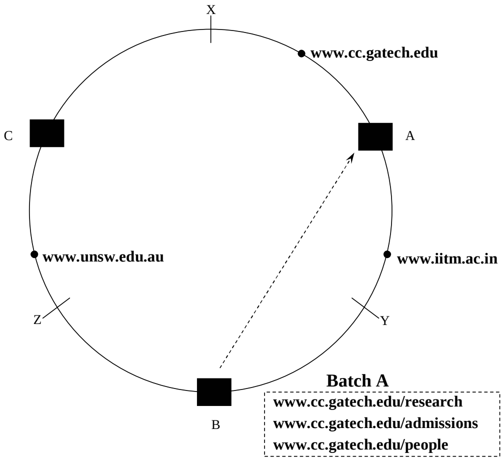

Fig. 1. Division of Labor for PeerCrawl

Peers and domains to be crawled are hashed to a common *m* bit space. A peer is responsible for those URLs whose domain name hashes onto it.

Similarly, the unstructured decentralized crawler (PeerCrawl) performs the division of labor by introducing a hash-based URL *Distribution Function* that determines the domains to be crawled by a particular peer. The IP address of peers and domains are hashed to the same *m* bit space. A URL U is crawled by peer *P* if its domain lies within the range of peer *P*. The range of Peer *P*, denoted by $Range(P)$, is defined by $h(P) - 2^n$ to $h(P) + 2^n$, where h is a hash function (like MD5) and n is a system parameter dependent on the number of peers in the system. In our first prototype of DSphere, we use the number of neighbor peers of P as the value of n. Thus whenever a peer joins or leaves the network, the range of all peers will change dynamically. Consider the example shown in Fig.1. The points X, Y, and Z in Fig.1 are computed based on the neighbor count of peers A, B, and C respectively. The Peer *A* in this case is responsible for all domains mapped between points *X* and *Y*, Peer *B* is responsible for all domains mapped between points *Y* and *Z* and Peer *C* is responsible for the rest of the domains. A peer broadcasts all URLs for which it is not responsible to other peers in the system. Interested readers may refer to [3] for further details.

3. Source-Based Link Analysis

To provide augmented relevance ranking and search capabilities, DSPHERE relies on source-based link analysis. Rather than providing a single global authority score to each Web page – much like Google's PageRank does – DSPHERE assigns two scores: (1) First, each source is assigned an importance score based on an analysis of the inter-source link structure; (2) Second, each page within a source is assigned an importance score based on an analysis of intra-source links. This approach is motivated by the strong locality-based nature of Web links, which is often centered around organizational responsibility (e.g., pages within one domain tend to link to pages within the same domain) [1]. SourceRank calculations are significantly faster than flat page-level approaches like PageRank, which is especially important considering the size and growth rate of the Web. We have additionally incorporated a suite of spam-resilient countermeasures into the source-based ranking model to support more robust rankings that are more difficult to manipulate than traditional page-based ranking approaches [2]. In practice, we treat each domain (like *www.cc.gatech.edu*) as a source. So, in Fig.1, Peer A will calculate a local PageRank-based score for each page in the *www.cc.gatech.edu* domain. Using source-based link analysis, we will then calculate a source score for the *www.cc.gatech.edu* domain by accumulating information from other peers regarding all sources that link to this domain. This can be done by providing all the source-based links to a single peer. This peer will be responsible for maintaining the source-based Web graph and calculating the SourceRank for all the sources from this graph. Since peers on the Web have different capabilities, we select more powerful peers for maintaining this source-based Web graph. The source-based Web graph is replicated across multiple peers for handling peer exits from the network.

4. Demonstration Overview

During the demonstration, a network of peers will be set up remotely (if the network connection is provided) to illustrate the features of the system.

- We display the features of the DSPHERE decentralized P2P crawlers, including the methodology employed to resolve URL duplicates, distribution of crawl jobs, crawling speed and quality of the crawled pages.
- We will provide a detailed analysis of our source-based link analysis approach and demonstrate how it outperforms existing algorithms in terms of refresh rates and robustness to Web Spam.

Acknowledgement. This work is partially supported by NSF CISE ITR, CSR, and CyberTrust program, an AFOSR grant, an IBM SUR grant and an IBM Faculty Partnership award.

References

[1] J. Caverlee, L. Liu, and W. B. Rouse. Source-based link analysis of the web: A parameterized approach. Technical report, Georgia Institute of Technology, 2006.

[2] J. Caverlee, S. Webb, and L. Liu. Countering web spam attacks using link-based analysis. Technical report, Georgia Institute of Technology, 2006.

[3] V. J. Padliya and L. Liu. Peercrawl: A decentralized peer-to-peer architecture for crawling the world wide web. Technical report, Georgia Institute of Technology, May 2006.

[4] L. Page, S. Brin, R. Motwani, and T. Winograd. The pagerank citation ranking: Bringing order to the web. Technical report, Computer Science Department, Stanford University, 1998.

[5] A. Singh, M. Srivatsa, L. Liu, and T. Miller. Apoidea: A decentralized peer-to-peer architecture for crawling the world wide web. In *SIGIR 2003 Workshop on Distributed Information Retrieval*. ACM, August 2003.