A New Adaptive, Semantically Clustered Peer-to-Peer Network Architecture♣

[1]S. Das [2]A. Thakur [3]T. Bose and [4]N.Chaki

[1]Department of Computer Sc. & Engg, University of Calcutta, India, soumava@acm.org
[2]Department of Computer Sc. & Engg, University of Calcutta, India, avishekt@yahoo.com,
[3]Department of Computer Sc. & Engg., University of Calcutta, India, tanumoyb@yahoo.com
[4]Department of Computer Sc. & Engg, University of Calcutta, India, nabendu@ieee.org

Keyword: *Bloom filter, Peer-to-Peer network, Super Peer, content searching*

Abstract

This paper aims towards designing and implementation of a new adaptive Peer to Peer (P2P) network that cluster itself on the basis of semantic proximity. We also developed an algorithm to classify the nodes to form the semantic groups and to direct the queries to appropriate groups without any human intervention. This is done using Bloom filters to summarise keywords of the documents shared by a peer. The queries are directed towards the appropriate clusters instead of flooding them. The proposed topology supports a system for maintaining a global, omnipresent trust value for each peer in an efficient manner both in terms of decision time and network load.

1. Introduction

Peer-to-Peer (P2P) networks are increasingly gaining acceptance on the internet as they provide an infrastructure in which the desired information and products can be located and shared. Some of the major factors driving the P2P adaptation are anonymity, autonomy, empowerment, censorship-resistance, collaboration and participation.

There are a lot of P2P systems that are in use today. Though P2P systems like the Napster [7,14], Gnutella [10,12], the Freenet [2,9], the Kazaa[11] etc have gained wide user acceptability, technically, there's a lot of scope for improvement. We have done an extensive study of the various P2P systems in section 2.

The existing P2P architecture may have a centralized control or with no central control. However, a third kind of approach that is evolving recently and is certainly looking quite promising, is the hierarchical P2P systems. The distinguishing feature of this type of systems is the presence of super peer – a peer which acts as the representative of a group of peers. Unlike the hybrid model, this model is fault tolerant since any of the peers can become super peer as and when the need arises. The hierarchy allows easy scaling and keeps the network traffic under control. This model is the most promising one but various problems need to be overcome before it can realize its potential.

Given the current state of peer to peer computing we would like to design and implement an Adaptive self-organizing semantic peer-to-peer network. We propose that connection establishment and query routing be guided by the contents stored at the member peers. We propose a mechanism to ensure that semantically related nodes cluster together. A special peer called a super-peer will represent each of the semantic clusters. We achieve the semantic clustering using Bloom filter[1]. Counting Bloom filters are

♣ Partly sponsored under SARAI project of the Centre for the Study of Developing Societies, Delhi, India

constructed containing the keywords of the documents the peer wants to share. These Bloom filters are send to the super-peer of that cluster. The super-peer then constructs a summarized Bloom filter for transmission to the other super-peers. The formation of the semantic clusters will facilitate in efficient search. The queries can be directed towards the appropriate clusters instead of just flooding them. This will reduce network traffic immensely. We would also like the trust management of the system to be such that accessing the trust value of any peer is faster for all the other peers when needed, and the opinions of the nodes who are not presently logged into the system be also taken into account while decision making.

2. REVIEW OF THE EXISTING SYSTEMS

In this section, we would consider some of the widely accepted P2P protocols and discuss the relative merits and de-merits of these existing systems.

2.1 Gnutella

The Gnutella protocol is an open, decentralized group membership and search protocol, mainly used for file sharing. The term Gnutella also designates the virtual network Internet accessible hosts running Gnutella-speaking applications and a number of smaller and often private, disconnected networks.

Gnutella peers provide client side interfaces through which user's can issue queries and view search results, accept queries from other peers, check for matches against their local data set and respond with corresponding results. These nodes are also responsible for managing the background traffic that spreads the information used to maintain network integrity. Every peer is connected dynamically to an average of six to seven peers, depending on the bandwidth of the peer's network connection. The Gnutella network employs a routing concept, known as 'Viral propagation" for the query messages in which a peer searching for content or exploring the network, sends out a query message, to all the neighboring peers it is currently directly connected to via TCP/IP connections in the virtual overlay network. Thus every peer is able to explore in a completely decentralized manner without the need for a central entity, by more or less simply flooding the network.

2.2 Freenet

Freenet is a distributed information storage and retrieval system that addresses the concerns such as - privacy and availability. The system operates as a location independent distributed file system across many peers, that allows files to be inserted, stored and requested anonymously.

The system is designed to respond adaptively to usage patterns, transparently moving replicating and deleting files as necessary to provide efficient service without resorting to broadcast searches and centralized indexes. It is not intended to guarantee permanent file storage, although it is hoped that a sufficient number of nodes will join with enough storage capacity that most files will be able to remain indefinitely. In addition, the system operates in the application layer and assumes the existence of a secure transport layer, although it is transport independent. It does not seek to provide anonymity for general network usage, only for Freenet file transactions.

Maintaining privacy for creating and retrieving files means little without also protecting the files themselves in particular, keeping their holders hidden from attack. Freenet thus makes it hard to discover location of files across the nodes. Together with redundant replication of data, holder privacy makes it extremely difficult for censors to block or destroy files on the network.

2.3 The Kazaa

Kazaa is a peer-to-peer file sharing application that allows its users to share files with other Kazaa users. It also allows users to search for and download files from other users who are sharing them. Kazaa has implemented a reputation management system consisting of two components, *Integrity Rating* and *Participation Levels*.

3. SCOPE OF THE WORK

The growth and spread of P2P networks has been dominated by one killer application: file-swapping over the Internet. The metadata associated with the files normally exchanged on these networks is minimal consisting mainly of the artist's name, title, etc.

In most of the P2P networks like Gnutella or Kaaza, the nodes connect to a limited number of other nodes and use these nodes as gateways to the network as a whole. These connections are either randomly established or based on network proximity or such other considerations as vector space representations. These either lead to inefficient search based on flooding or impose restrictions on where a particular document or file has to be stored [15, 18].

It would be more efficient in terms of searching if the connection establishment is be guided by the content stored at that peer. For example, a peer publishing or sharing files on 'Rock' should connect to the super-peer of a group of related peers. However, the classification is not exclusive. The same peer might be sharing files on 'Football' also and hence will connect to a super-peer representing nodes sharing files on 'Football'. Thus semantically related nodes tend to cluster together. The formation of semantic clusters facilitate efficient search as queries can be directed to the appropriate clusters instead of just flooding them. This enables the queries to be answered quickly and with minimum network load as done in SON[4]. However, SON classified the peers and queries manually and the number and scope of each cluster was pre-determined and static.

One of the concerns of P2P systems is the method of keeping a track about the trustworthiness of the peers. In both FreeNet and Gnutella, the reputation of the peers is not addressed, this makes the system vulnerable to attacks by malicious peers. There are a few *reputation management systems* like the XRep[6] or EigenRep[13] which do not suffice to our requirements simply because either the decision making process in them is too difficult (like in XRep) or some of them do not take into account, the opinion of the peers which are logged off, which eventually leads to false trust values for any peer. TrustMe[16] is a system however where some of these aspects are very well handled, but TrustMe depends very heavily on broadcasting.

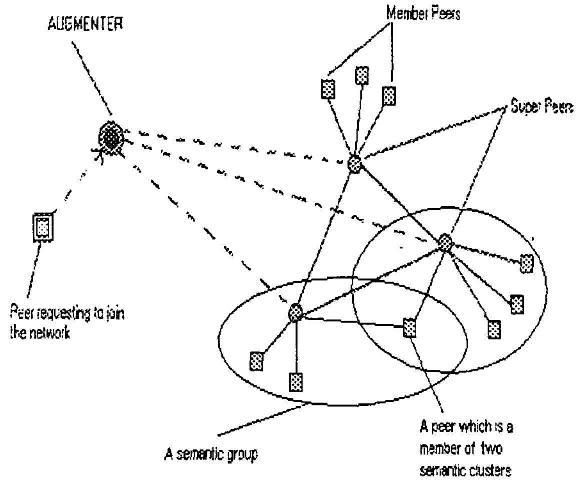
Thus there is a need to develop a P2P network that supports efficient keyword based search by exploiting the semantic relationships among the peers. The network needs to efficiently support a system for maintaining a global, omnipresent trust value for each of the peers, which can be accessed by any other peer.

4. THE PROPOSED P2P TOPOLOGY

The present work aims towards assimilation of the best features of the various existing networks eliminating their performance bottle-necks in a significant manner. We have adopted a hierarchical model like in Kazaa but extended it to form semantic overlays like in SON. However, unlike SON, we have proposed a method to classify the nodes and queries and to form the semantic clusters without the need for any human intervention. This was done using PlanetP's[5] concept of using Bloom filters to summarise the keywords of the documents shared by a peer. But, in order to ensure efficiency and scalability, we do not propagate these filters to every peer. Also, we use a modification of the Bloom filter in order to maintain a weight for each of keywords in the filter. Thus we have been able to assimilate the best of the existing networks while overcoming their shortcomings. In the section below we will describe our proposed architecture in detail.

4.1 The proposed architecture

In conformation with the recent trend we will adopt a hierarchical nature for our proposed news distribution system. We propose a self-organizing, adaptive network. Three kind of nodes are used in the network, an Augmenter (AUG), a Super Peer (SP), and the Member Peers (MP). In the proposed network, a hierarchy of the nodes will be formed like this: a group of nodes which stores or publishes related items will form a peer group. The nodes are grouped together based on the semantics of the materials stored in them. These nodes are called the member peers (MP). These peers form one level of the hierarchy. Each of these semantic groups will have a Super Peer (SP) representing them. The SPs represent another level in the hierarchy. So, the total system will have many SPs, each representing a particular semantic group. The MPs are members of these SPs. Any MP can be a member of more than one semantic groups i.e., we allow the overlapping of semantic groups. The task of the AUG is to assist the nodes that want to join the network. The sole work of the AUG is to introduce the node into the system.



(Fig. 1: The propose P2P network topology)

We also propose that connection establishment and query routing be guided by the contents stored at the member peers. For example, a peer publishing or sharing files based on "Rock music" should be the member of a SP whose other MPs also deal with "Rock music". However, this classification is not exclusive. The same member peer might be sharing substantial amount of files on "football" also, and hence will also be a member of a SP who deals with "football". i.e. we propose overlapping of the semantic groups be possible. The formation of the semantic clusters will facilitate in efficient search. The queries can be directed towards the appropriate clusters instead of just flooding them.

We further propose an efficient system for maintaining a global, omnipresent trust value for each of the peers, which can be accessed easily by any other peer.

4.2 Operations and Algorithms

In this section we take a brief look at how the network is supposed to perform its various tasks. We present a detailed step by step view of the functionalities of the system.

4.2.1 Peer joining: When a new node wants to join the network, it requests the AUGMENTER (AUG) that it wants to join the network. Since there can be more than one AUG in the systems, the node chooses an AUG based on network proximity. The AUG now assigns a global ID to the node and the AUG asks the node to contact a particular SP. The node now contacts the SP and sends its public key, global ID, its capabilities to the SP. The SP accepts the node as one of its MP and updates its own MP records. The MP record table in the SP contains the fields for MPs global ID, its current IP, its public key, its trust value, and a field to store the counting bloom filter of the node. The SP now sends its public key to the MP.

4.2.2 Rejoining: Whenever a MP leaves a network and rejoins its group later (this is a regular scenario in case of a MP using a dial up line), it has to let the new IP address made known to its super peer. The SP will store the mapping between the global name and the current valid IP address of that MP in its MP records.

4.2.3 Publishing news items: When a peer is interested in publishing any document, it produces a counting Bloom filter[1] with the key words of all the documents it wants to publish and sends that bloom filter to its SP. The SP now updates the record of this MP by storing the new bloom filter. Hence, when a new item is to be published by this MP, it recalculates the counting bloom filter in the same manner and sends it to its SP for update.

4.2.4 Peer leave: When a particular MP wants to leave the system, they can just notify its SP and leave the network. When super peers want to leave, they will have to determine a new super peer who will take up their task. The SP will decide on a new SP for all its MPs. This decision is taken based on whether a particular MP is interested in being a SP, the capabilities of that SP, and the current trust value of the MP.

4.2.5 Query routing: We have seen that all the SPs store the counting bloom filters of its MPs, these bloom filters reflect the content stored at each MP. Each SP also calculates a summarized bloom filter. This summarized bloom filter reflects the contents of all the bloom filters of the SP's member peers. This summarized bloom is send to all the other SPs at regular intervals. The querying member peer sends its query, which is a single key word or a set of key words, to its SP. The SP matches this key set to the stored summarized blooms. Then the query is routed to the SPs who's summarized bloom filters are the closest match to the set of key words. This SP is now the serving SP. It selects the MP with the closest match and obtains information on the files that these MPs store regarding the queried keywords. The serving SP sends this information to the querying SP along with the trust rating of the serving MPs. The querying SP sends this information to the querying MP. The querying MP now selects a particular file from the information list of files send to it. It asks its SP for the file and following the same chain through the serving SP, the query is served.

4.2.6 Propagation of trust values: Each peer is allowed to integrity rate its shared files. This information is send along with file information whenever information about the file is asked for. The super peers store the trust value of each of its members. Initially all the peers start with a predefined trust value. As the peer continues to stay in the system and starts to interact and share documents in the system, its trust value gradually starts reflecting the behavior of that peer. The trust value depends on opinions of peers about each other. Each peer sends an opinion after an interaction. Decisions are based on both the file rating and the opinion about a peer. The opinions are routed to the SPs of the serving peers, since the SPs store the trust value of all its MPs.

5. CONCLUSION

In this section we would like to summarize the advantages of our proposed P2P network over the existing P2P networks. First and most importantly, in the proposed system, the queries are highly directed. The queries are always directed towards the appropriate semantic cluster hence reducing network traffic considerably. The proposed query servicing mechanism helps maintaining a low average network traffic comparing the other P2P systems. The decision making for the MP, about which file to download, is faster as compared to some other existing techniques like the XRep.

In our work, we have used a counting Bloom filter consisting of an array of bytes. Instead of using eight different hash functions to obtain the indices, we have obtained the SHA1 hash of the keyword and divided the resulting 160 bit long hash in to 8 equal portions of 20 bits each. These portions act as the indices and the byte at the position indicated by these addresses are incremented by one. In order to determine the membership and the weight of a word, we calculate the indices in a similar way. The weight of the word in the given Bloom filter is the minimum of the value of bytes at those index positions. If the weight is equal to 0 then the word is not in the set represented by the Bloom filter.

The system also preserves anonymity of the users. Every node interacts with just the next node. Even the sender does not know about who is the consumer, it just interacts with its SP.

Unlike Freenet and Kazaa, reputation of the peers is addressed in the proposed P2P architecture. Though EigenRep addressed this issue as well, the drawback in EigenRep was that, while accessing the trust value for a peer, EigenRep did not take into account the opinions of the peers who are logged off at that time. In our topology, trust value stored is global, so all the opinions are reflected in the trust values all the time.

References

[1] B. H. Bloom, "Space/time trade-offs in hash coding with allowable errors", Comm. ACM, 13(7), pp:422–426, 1970. ISSN 0001-0782.

[2] I Clarke, O. Sandberg, B Wiley, T. W. Hong, "Freenet: A distributed anonymous information storage and retrival system", LNCS, 2001.

[3] E. Cohen, A. Fiat, H. Kaplan, "A case for associative peer to peer overlays", SIGCOMM Comput. Commun. Rev., 33(1), pp:95–100, 2003.

[4] A Crespo, H G Molina, "Semantic overlay networks for p2p systems", Technical report, Santford University, January 2003.

[5] F. Acuna, T. D. Nguyen, "Text-based content search and retrieval in ad-hoc p2p communities", Proc. of the Networking Workshops on Web Engineering and Peer-to-Peer Computing, pp. 220–234, 2002.

[6] E. Damiani, et.al, "A Reputation based approach for choosing reliable resources in peer-to-peer networks", Proc. of the 9th ACM conf. on Computer and communications, USA, pp. 207-216, 2002.

[7] http://opennap.sourceforge.net/napster.txt.

[8] Fasttrack web site. URL http://www.fasttrack.nu.

[9] URL http://freenet.sourceforge.net/.

[10] Gnutella web site. URL http://www.gnutella.com.

[11] The Kazaa web-site. URL http://www.kazaa.com.

[12] Patrick Kirk. RFC-Gnutella, 2003.

[13] S. Kamvar, M. Schlosser, and H. Garcia-Molina, "Eigenrep: Reputation management in p2p networks", Proc. of 12th Intl. World Wide Web Conference, 2003.

[14] Napster home page. URL http://www.napster.com.

[15] S. Ratnasamy, P. Francis, M. Handley, R. Karp, S. Schenker, "A scalable content-addressable network", Proceedings of the 2001 conference on Applications, technologies, architectures, and protocols for computer communications, pp. 161–172. ACM Press, 2001.

[16] Ameek Singh, Ling Liu, "TrustMe, Anonymous management of trust relationship in decentralized P2P systems", College of computing, Georgia Tech.

[17] A. Singla, C. Rohrs, "Ultrapeers:Another step towards Gnutella scalability", Gnutella Developer Forum,http://groups.yahoo.com/group/the_gdf/files/Proposals/Ultrapeer/Ultrapeers.html.

[18] I. Stoica, R. Morris, D. Karger, M. F. Kaashoek, H. Balakrishnan, "Chord: A scalable peer-to-peer lookup service for internet applications", Proc. of the conf. on Applications, technologies, architectures, and protocols for computer comm., pp. 149–160. ACM Press, 2001