EFFICIENT SEARCHING FOR FILE SHARING IN PEER TO PEER WITH REPOSITORY FUNCTION &SUPER PEER

R.Satheesh
ME. Computer and Communication Engineering
Department of Information Technology, SNS College of Technology,
Coimbatore, Tamil Nadu
sthheesh.mr@gmail.com

G.Selvavinayagam
Asst.Professor, Department of Information Technology, SNS college of Technology,
Coimbatore,Tamil Nadu
ohmselva@gmail.com

***Abstract*- Optimizing the search based on multikeyword using Bloom Filter minimizes the traffic cost. Existing scheme gives the optimal setting of Bloom Filter to reduce the search traffic. Search traffic is further minimized by implementing repository function on server search side. The involved inverted list information, during search is temporally stored. Gathered information from stored results of query logs minimizes the search traffic, instead of whole query search throughout the network. Super Peer based P2P architecture minimize the processing time and latency by providing server for a set of clients. Cluster of server search makes the gathering of information fast to the main search server. The construction of repository function and Super Peer reduce the search traffic and latency.**

Keywords* — *Repository function, Super Peer, Bloom Filter, multikeyword search,P2P

I. INTRODUCTION

Peer-to-Peer makes the sharing of files or information through the network. Distribute information throughout the network Peers are gathered and listed to make the establishment of Peer connection with the searching of files based on multikeyword. The keywords are mapped to all Peer's in the network and retrieved results are listed in inverted index in the form of table. With multikeyword search lot of search traffic occurs, for example- when two keywords "Bloom Filter Setting". The word is split into "Bloom Filter" and "setting", keywords are searched separately and results are listed, this cause lot of data traffic.

Bloom Filter is used to reduce search traffic. Bloom Filter is to find whether an element is present in a set or not. BF is a bit vector with m bits. BF with multikeyword search is done with distributed intersection and union operation. "AND" and "OR" queries maximize the optimal setting of Bloom Filter.BF is a lossy but succinct and efficient data structure to represent a set S, which can efficiently process the membership query such as "is the element x in the set S."By transmitting the encoded sets instead of raw sets among peers, the communication cost can be effectively saved

In this work the search traffic is minimized further by implementing repository function in search server side which acts as a temporary storage of the query log results. Repository function makes the search traffic by mapping the keyword given by Peer to Repository Database. Process of searching is increased by constructing Super Peer based P2Parchitecture. Super Peer acts as a server for a set of clients and distributing the keyword to the cluster of Peers. By Super Peer the latency is minimized and makes the processing result fast.

With the help of NS2 (Network Simulator 2) the processing speed and the latency can be calculated. Tracegraph shows the decrease in latency and the processing speed.

A. Table Inverted Index

The searched keywords *are* retrieved by Text Information Retrieval System. This allows efficient retrieval of files based on set of keywords searched. The query from a Peer is distributed and gathered information's are listed in the form of table. Required file is selected from the table and retrieved by downloading it.

B. Ranking of list

Ranking depends on the multikeyword searching. The retrieved result is ranked based on file size, relationship to keyword given by Peer and based on type of file.

C. Mulikeyword Search

User behaviour is calculated by measuring the query logs used to search for file. Multikeyword search is common in all web searches. Bloom Filter is merged in multikeyword search. This filter makes the search traffic less by optimal setting of the filter with numerical analysis of the function.

II. APPROACHES IMPLEMENTED IN MINIMIZING SEARCH TRAFFIC

Two approaches are implemented for further minimizing the search traffic and latency

- Repository function
- Super Peer-based P2P architecture

These two approaches make P2P searching with keyword fast and decrease the latency. Repository function is implemented in the server search side, this act as a temporary storage of all the query log results and the results are kept in the repository database. Repository function minimizes the search traffic and the search gets completed if the keyword is found in the repository database, instead of distribution of keyword to entire Peer's.The repository function is directly connected to the peer and the results are mapped in the repository database. Repository function is to limit the search traffic.

Super Peer based P2P architecture increase the retrieval of search result fast. Super Peer acts as a server for a set of clients and search is done only among those clusters of Peer's a Super peer is responsible for. This architecture merges with the server search and the retrieval results are listed to the Peer which is searching a file with the keyword.

III. Bloom Filter

Bloom Filter optimal setting is constructed in the Peer side. BF withmultikeyword searching is minimized with numerical analysis of the filter by BF(X∩Y∩Z). When three sets XYZ is taken

X={a,b,c,d,e}, Y={c,d,e,f,g}, Z={c,f,h}

First query x,y,z is send to distributed hash table which consist of set X={a,b,c,d,e}. Bloom Filter of X is taken and send to distributed hash table which consist of Y set Y={c,d,e,f,g}, numerical equation for calculating between X and Y is Y∩BF(X)={c,d,e|f}. When set Y∩BF(X) is again send to hash table which consist of X set.

X∩(Y∩BF(X) = {c,d,e}

BF(X∩Y) = {c,d,e}

Bloom Filter with multikeyword give the optimal result. BF (X∩Y) is send to distributed hash table consisting of Z set

Z = {c,f,h}

Z∩BF(X∩Y) = {c|h}

BF {X∩Y∩Z} = {c}

Bloom Filter in Peer side works when some other Peer is searching a file with keyword. Keyword is distributed to all Peers' and the keyword is mapped to the files with the similar keywords. Search is done with Bloom Filter. Results are gathered in the form of list.

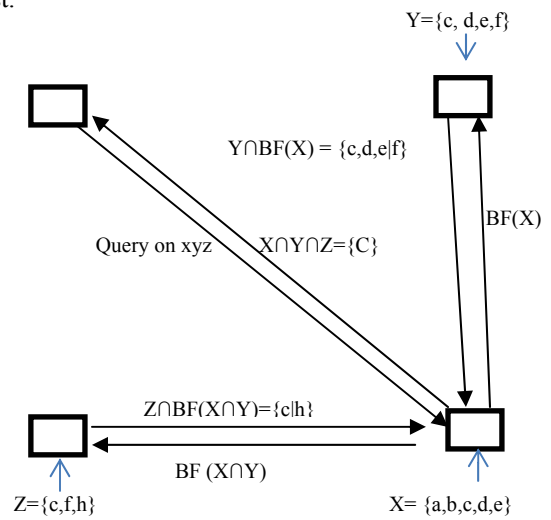


Figure1. Bloom Filter function

Working of Bloom filter is optimized with multikework searching. Search traffic is minimized and communication cost is also minimized. Bloom Filter used efficiently computes the intersection between two sets. BF results in compression of the keyword of the keyword search result. The hashing of all elements is minimized.

Two Queries are carried out in Bloom Filter

- AND query
- OR query

Multikeyword search is done with both the "AND" and "OR" query. Communication cost and search traffic is minimized with this numerical analysis. False Positive Rate is minimized by setting Bloom Filter.

IV. SYSTEM DESIGN

In a P2P multikeyword search network we focus mainly on minimizing search traffic and latency. The Bloom Filter optimal setting with "AND" query and "OR" query to get efficient search result list. Query logs are collected to show the results of search engine.

A. Architecture for a P2P Searching

P2P searching for a file in this paper is done with the distribution of multikeyword through the network with search server engine. The gathering of information and distribution is done by using Gossiping Algorithm. This algorithm is used to gather global information in an unstructured P2P network. The query is passed into the repository in the server search side. The keyword is mapped to the repository database to find out whether the keyword is present in the table consisting of results of query logs. If the keyword is not present in the repository database, then the keyword is distributed to the Super Peer based P2P Architecture.

Super Peer acts as a server for a set of peers. After query search the results are listed in the form of table and displayed in the peer side. The architecture of peer to peer makes the analysis of the working in searching of the file and the query response is displayed. The searching of the query is collected in the format of table within the server search and is displayed to peer. The super peer provides the load balance between the peers and manages the peer which it is responsible for. When the super peer fails the server assume itself as super peer and gathers the query information. The P2P architecture shows the overall working of super peer and the P2P communication.

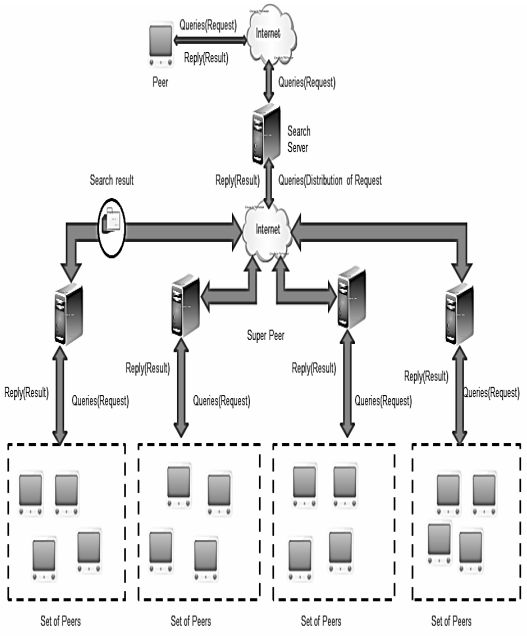


Figure2. P2P Architecture

The multikeyword which is given by the Peers send to the search server and it is distributed to the Super Peer. Theresults are listed in the form of table and are displayed in the peer side. Query results are ranked according to its size and most commonly searched words. This minimizes the false positive rate which occurs during the keyword search in the Peer side. Communication cost is reduced with this architecture.Latency of search is the main drawback which is overcome by implementing repository function. Latency is also based on time propagation and the link bandwidth. Latency is reduced with splitting the Peer and allocating a Super Peer of a set of clients. Thus reduce the latency and performance is increased.The Peer-to-Peer architecture shows the communication between peers and the reduce in search traffic. This shows the decrease in peer side processing.

B. Algorithm Used :

This system makes use of two types of algorithms namely,

- Gossiping Algorithm
- Lookup Algorithm
- Distributed Algorithm

The distribution of keyword and gathering of information is done with gossiping algorithm. The location of the search result in the repository is found with lookup algorithm. Lookup algorithm also performs the identification of the Peer's potential and location in Super Peer.

The lookup algorithm is to provide regular keyword to each peer with the entry of id among the cache entries. This algorithm showshow peer can remove the O(log n)-state-period restriction on existing DHT topologies to achieve significantly better lookup performance. Distributed Algorithm is for "AND" query and "OR" query for minimizing communication cost and to get optimal results based on the keywords given by the peer.

C. Repository Function

Repository Function is implemented in the server search side. The main working of repository function is to send result of the query logs for a particular time. The keyword search is first done in the repository, if present the query log result is send to the peer and further search will not take place. This function is to minimize the search traffic but getting searched keyword result without entering into whole network search.The repository function is directly connected to the peer and the results are mapped in the repository database. Repository is emerged in the sever search when given keyword is mapped and stops the search when it is found. The table consisting of the query result is displayed to the Peer.

D. Super Peer Based P2P

Super Peer acts as a server for a set of clients and makes the searching process faster. The keyword from the main search server is send to the Super Peer and the search results are gathered in the main server and transferred to the peer. Super Peer has been proposed to improve but structured and unstructured Peer-to-Peer network. Lookup algorithm exploits the Peer's involved in Super Peer and identifies the potential of the Peer in the network. Peers are then in Super Peer that a Peer is reliable and it done not leave the P2P network frequently.

E. P2P search engine

When a query is requested a query integrator first retrieves the locations of the relevant inverted lists and similar data's through DHT lookups, and then uses this data to design a good query execution plan that specifies how peers communicate with each other during request of query. Queries are issued by peer with the P2P system which acts has a query integrator for showing better connected node within the P2P system. Threshold algorithm is used Peer might also act as intermediaries (search servers) that forward queries issued by web clients and relay back the results. With this search engine as the base the P2P architecture is improved and the repository function is implemented in the sever search side the minimize in communication cost. This search makes the gathering of request fast and minimizes the false positive rate. In peer side the Bloom Filter setting takes place.

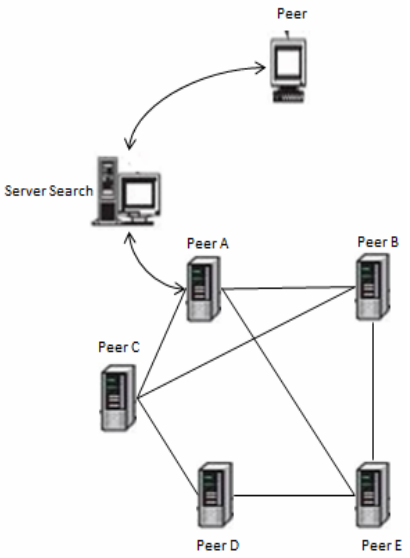


Figure3. P2P Search Engine

F. Working of data flow in P2P Network

Flow of data is shown when a peer gives request for retrieving the query from repository database.

Peer query is given as input to the repository database and the output is send to the search list and to the peer. The peer can view the result which is displayed in the form list.

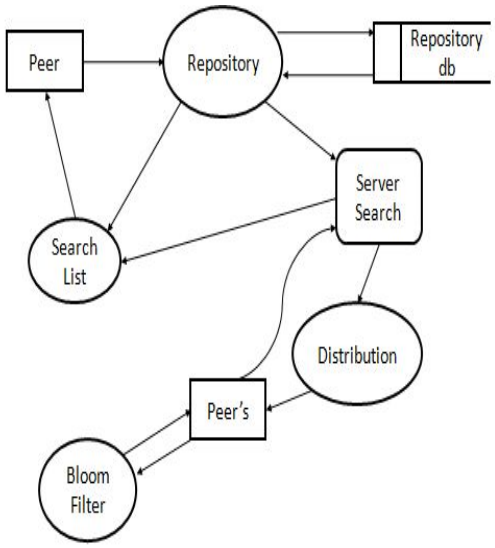


Figure4. Data Flow in P2P Network with Repository Function

The input is further passed to the server search and is distributed to peers, if the input is not found in the database. The bloom filter setting is present in the peer side for minimizing communication cost.

V. SIMULATION TOOL

In this system we make use of NS2 tool for showing the process speed and the latency.

NS-2 is used for many purposes including:

1. To evaluate the performance of existing network protocols.
2. To evaluate new network protocols before use.
3. To run large scale experiments not possible in real experiments.
4. To simulate a variety of ip networks

Latency is calculated with the help of trace graph. By setting Super Peer the processing speed is increased and the latency is decreased. With NS2 stimulation tool, reduce in search traffic is shown and the communication cost is minimized. The false positive rate is minimized with optimal setting of Bloom Filter.

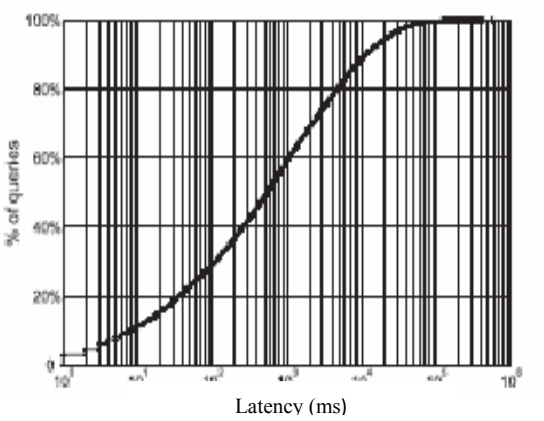
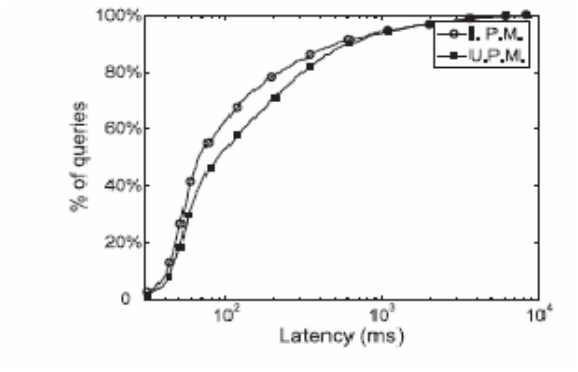


Figure5. Latency result of the previous work



Results of implementing Super Peer

In this, the two graphs show the latency result of the search query result. The first graph is the result of the previous work result. The second graph show the difference by showing the increase in performance of query result.

VI. RESUTLS

Based on above graph, we would be able to compare the performance results of the existing work and time latency difference is observed from this paper. The gathering of data is make fast with the repository and latency is reduced with increasing the performance of search.

VII. CONCLUSION

In this paper, by using NS2 Tool it's easy to show how the search traffic is minimized and the

latency is decreased. The P2P searching is made faster by using Super Peer concept and repository function. Repository function limits the search traffic. It is clear that with repository function the keyword search is made fast and the communication cost is reduced Stimulation results show the advance from the existing work. The false positive rate is also minimized with Bloom filter settings.

For Future work, the processing of search can be further increased with Hierarchical Bloom Filter Arrays (HBF) to map filenames to the metadata servers holding their metadata faster and more reliable

ACKNOWLEDGEMENT

Like to express gratitude to G.Selvavinayagam, Asst.Professor SNS College of Technologyfor the detailed comments and numerous suggestions which led to improvements in the paper and to all the reference for developing this paper

REFERENCES

[1] Hanhua Chen, Hai Jin, Lei Chen "Optimizing Bloom Filter Settings in Peer-to-Peer Multikeyword Searching "IEEE TRANSACTIONS ON KNOWLEDGE AND DATA ENGINEERING, VOL. 24, NO. 4, APRIL 2012

[2] J. Zhang and T. Suel, "Efficient Query Evaluation on Large Textual Collections in a Peer-to-Peer Environment," Proc. IEEE Int'l Conf. Peer-to-Peer Computing (P2P), 2005.

[3] Yingwu Zhu, Honghao Wang, Yiming Hu "A Super-Peer Based Lookup in Structured Peer-to-Peer Systems" IEEE Int'l Conf. Peer-to-Peer Computing (P2P), 2007

[4] ShrideepPallickara, HasanBulut and Geoffrey Fox, "Fault-Tolerant Reliable Delivery of Messages in Distributed Publish/Subscribe Systems"Proceedings of ACM/IEEE Supercomputing 2002 (SC2002)

[5] B.Jalender, Dr A.Govardhan, "Breaking the Boundaries for Software Component Reuse Technology" IEEE TRANSACTIONS ON DATA ENGINEERING, VOL.13, NO. 4, APRIL 2006

[6] R. Boichat, et al. Effective Multicast programming in Large Scale Distributed Systems. CCPE, 2000.

[7] D. Kempe, A. Dobra, and J. Gehrke, "Gossip-Based Computation of Aggregation Information," Proc. Ann. IEEE Symp. Foundations of Computer Science (FOCS), 2003.

[8] S. Nath, P.B. Gibbons, S. Seshan, and Z.R. Anderson, "Synopsis Diffusion for Robust Aggregation in Sensor Networks," Proc. Int'l Conf. Embedded Networked Sensor Systems (SenSys), 2004.

[9] A. Broder and M. Mitzenmacher, "Network Applications of Bloom Filters: A Survey," Internet Math., vol. 1, no. 4, pp. 484-509, 2005.

[10] M. Bender, S. Michel, P. Triantafillou, G.Weikum, and C. Zimmer, "P2P Content Search: Give the Web Back to the People," Proc. Int'l Workshop Peer-to-Peer System (IPTPS), 2006.

[11] G. Skobeltsyn, T. Luu, I.P. Zarko, M. Rajman, and K. Aberer, "Web Text Retrieval with a P2P Query-Driven Index," Proc. Ann. Int'l ACM SIGIR Conf. Research and Development in Information Retrieval (SIGIR), 2007.

[12] C. Tang and S. Dwarkadas, "Hybrid Global-Local Indexing for Efficient Peer-to-Peer Information Retrieval," Proc. Networked Systems Design and Implementation (NSDI), 2004.

[13] J. Lu and J.P. Callan, "Content-Based Retrieval in Hybrid Peer-to- Peer Networks," Proc. Conf. Information and Knowledge Management (CIKM), 2003.

[14] N.F. Huang, R. Liu, C.H. Chen, Y.T. Chen, and L.W. Huang, "A Fast Url Lookup Engine for Content-Aware Multi-Gigabit Switches," Proc. Int'l Conf. Advanced Information Networking and Applications (AINA), 2005.

[15] P. Reynolds and A. Vahdat, "Efficient Peer-to-Peer Keyword Searching," Proc. Int'l Conf. Distributed Systems Platforms and Open Distributed Processing (Middleware), 2003.

[16] T. Suel, C. Mathur, J. wen Wu, J. Zhang, A. Delis, M. Kharrazi, X. Long, and K. Shanmugasundaram, "Odissea: A Peer-to-Peer Architecture for Scalable Web Search and Information Retrieval," Proc. Int'l Workshop Web and Databases (WebDB), 2003.

[17] Riener. A, "Online UPnP A/V Device Database for Quick and easy Capability Checking" Frontiers of Information Technology(FIT) Conference Publications, 2011..

AUTHORS PROFILE

R.Satheesh has completed his Bachelor of Technology in Ranganathan engineering college in 2011 under Anna University Coimbatore. Pursuing Mater of Engineering in Computer and Communication Engineering in SNS College of

Technology 2011-2013 under Anna University Chennai.

G.Selvavinayagam has completed Bachelor of Engineering in Computer Science and Engineering in 2003 under Bharathiar University, Master of Engineering in Computer and Communication Engineering in 2009 under Anna University Chennai, Master of Science in Psychology in 2009 under University of Madras, Master of Business Administration in Human Resource in 2010 under Bharathiar University Chennai and pursuing Ph.D in Computer Science under Anna University Chennai. He has published many research paper in National and International Journals. Currently he is a member of Professional Bodies like ISTE, IACSIT,etc. His area of interest includes Theoretical Computer science, Automata Theory,Crytography and Network Security.