ISSN (Online): 2455-7838
SJIF Impact Factor (2015): 3.476

EPRA International Journal of

Research & Development (IJRD)

Volume:1, Issue:3, May 2016

Published By : EPRA Journals

SJIF Impact Factor : 3.476 ISSN: 2455-7838(Online)
EPRA International Journal of Research and Development (IJRD)
Volume: 1 | Issue: 3 | May | 2016

A PROXIMITY-AWARE INTEREST-CLUSTERED P2P FILE SHARING USING SOCIAL NETWORK FILE SHARING SYSTEM

Avibala.V[1]
[1]PG Scholar, Department of Information Technology, K.S.Rangasamy College of technology, Tamil Nadu, India.
Priya.V[2]
[2]Assistant Professor, Department of Information Technology, K.S.Rangasamy College of technology, Tamil Nadu, India.

ABSTRACT

Peer-to-peer (P2P) applications have shown their popularity on the Internet for file sharing. In contrast to the traditional client-server model of content distribution, scalable a large number of users and amount of content, greater fault tolerance for content to be shared by multiple sources, and less time for receiving required by a given data file to download , We carry a Proximity-Aware and interest clustered P2P file-sharing system (PAIS) based on a structured P2P that in a cluster and other groups physically-close and common interest node forms physically-close-nodes in a sub cluster on a based hierarchical topology. It drops copies of files that are often requested by a group of physically close nodes in their position. First, it classifies further the interests of sub-cluster to a number of sub-interests and cluster common interest sub nodes in a group for file sharing. Secondly PAIS builds an overlay for each group that connects lower capacity nodes to higher capacity nodes for distributed file query, while the node congestion avoid .third to reduce file search delay. Furthermore, the experimental results show the high effectiveness of intra-sub-cluster file file search efficiency of the search to improve approaches. We propose that a system that integrates a social network in a P2P network, called social P2P, for simultaneous efficient and trusted file sharing. It includes three mechanisms: (1) Search interest confidence-based structure, (2) interest/confidence-based file, and (3) trust setting. By exploiting combines the social interests and relations in the social networks, the interest trust structure based groups common interests more nodes in a cluster and other socially close nodes within a cluster. form a Distributed Hash Table (DHT) The relatively stable nodes in each cluster for inter-cluster file search. In the interest / confidence-based file search mechanism, a file request is forwarded to the cluster of the file first by the DHT routing. Then it is passed along constructed links within a cluster, achieved the high hit rate and reliable routing. In addition, files advises under socially close friends sharing nodes from providing erroneous files, because people are unlikely to risk their reputation in the real world. In the trust relationship adjustment mechanism, each node adaptively takes in a routing path his confidence on the node that has forwarded an incorrect file to visits to downed node to avoid routing later. We have extensive track-driven simulations and implemented a prototype on Planet Lab. The experimental results show that social P2P scored highly efficient and reliable file sharing compared to current file-sharing systems and Trust Management Systems.

KEYWORDS: *P2P networks, online social networks, file sharing, DHT*

1. INTRODUCTION

Peer-to-Peer (P2P) systems are in file-sharing applications such as widespread as Bit Torrent. Downloaded more than 50 percent of the files, and 80 percent of the files are uploaded via the Internetthrough P2P networks [1]. P2P file sharing systems to attract millions of users. Through the large area of P2P systems, a desired file locating an open problem for many years has been. Given the number of users with no pre-existing trust relationships in P2P open platform, trustworthy file-sharing offer has become another important issue. In fact, many users found in the wrong download files with misleading file names and descriptions. Peers flawed with malicious intent upload files, defined as manipulated files or files with malicious code (for example, Trojans and viruses). The research on the issues of efficient and trusted P2P file search was carried out separately. Although many methods to improve the efficiency and reliability, the individual methods are not efficiently been even proposed. In order to improve search efficiency, bundle made some works to increase the node with the same interest rate file. Since these methods with a single node of interest cluster, a node must use multiple interests of several clusters obtained which produces a high overhead. In addition, most of the previous approaches depend on the contents of the user to close the local storage of their file interests. They are not only expensive but also not able to retrieve the entire interests of a user with stored content, such as new users or to users who have shared files deleted. a system that integrates a social network in a P2P network, namely social P2P, while achieving efficient and trustworthy P2P file-sharing through the use of social interests and relations. Two facts lay the foundation for this work. First, people usually share files to which they are interested. [9] Interests that accurately reflect by a user himself in his profile the entire interests of the user. Second, users are unlikely to provide faulty available dismantle files on their socially close friends, because it affects their social relationships with other people and their reputation in their social communities in the real world. Thus can be discouraged by the network P2P Cyber mapping to the restricting social network cyber-services (for example, file sharing and message routing) between socially close nodes misconduct (ie, the provision of faulty files and forwarding messages rejection) ,

2. RELATED WORK

Chord: A Scalable Peer-to-peer Lookup Protocol for Internet Applications

Peer-to-peer systems and applications are distributed systems. Without centralized control or hierarchical organization, in which each node executing software with equivalent functionality. A review of the features of the latest peer-to-peer applications results in a long list: redundant storage, durability, choice of nearby servers, anonymity, search, authentication and hierarchical name. Despite this variety of functions, the operation core in most peer-to-peer systems is efficient location of data elements. The contribution of this work is a scalable protocol for searching in a dynamic peer-to-peer system with frequent nodes arrival and departure. A fundamental problem that applications Peer-to-peer are facing the efficient location of the node that stores a desired data element. It turns Chord, a distributed lookup protocol that addresses this issue. Chord supports only one operation: As a key, it is the key to a node. Data location can be easily implemented on top Chord by a key with any data item associating and storing the key / value pair at the node to which the key card. Chord fits efficiently connect as nodes and leave the system, and can answer questions, even if the system changes continuously. Results from theoretical analysis and simulations show that Chord is scalable: communication costs and the state of each node scale held logarithmically with the number of Chord nodes.

Pastry: Scalable, decentralized object location and routing for large-scale peer-to-peer systems

The design and evaluation of pastries, a scalable, distributed object location and routing substrate for wide-area peer-to-peer applications. Bakery performs application-level routing and the location of the object in a potentially very large overlay network of nodes that are connected via the Internet. It can be used to support a variety of peer-to-peer applications, including global data storage, data exchange, communication in the group and designation. Each node in the Pastry network has a unique identifier (Node ID). When presented with a message and a key, the message to the node with node ID pastries node efficient routes that is numerically closest to the key that currently live nodes under all pastries. Each Pastry node pursues its immediate neighbors in the node ID space and Notify their applications of new node arrivals, node failures and rebuilds. Pastries considered network locality; It seeks to distance travel news, how to minimize the number of IP routing hops according to a scalar to close metric. Pastry is fully decentralized, scalable and self-organizing; it automatically adapts to the arrival, departure and loss of nodes. The experimental results with a prototype implementation on an emulated network of receiving up to 100,000 nodes con firm Pastry scalability and efficiency, the ability to organize themselves and adjust defaults to the node, and its good network locality properties.

Tapestry: A Resilient Global-Scale Overlay for Service Deployment

Tapestry, a peer-to-peer overlay routing infrastructure offering efficient, scalable, location-independent routing of messages directly to nearby copies of an object or service using only localized resources. Tapestry supports a generic decentralized object location and routing applications programming interface using a self-repairing, soft-state-based routing layer. It presents the Tapestry architecture, algorithms, and implementation. It explores the behavior of a Tapestry deployment on

Planet Lab, a global test bed of approximately 100 machines. Experimental results show that Tapestry exhibits stable behavior and performance as an overlay, despite the instability of the underlying network layers. Several widely distributed applications have been implemented on Tapestry, illustrating its utility as a deployment infrastructure. Tapestry uses adaptive algorithms with soft state to maintain fault tolerance in the face of changing node membership and network faults. Its architecture is modular, consisting of an extensible up call facility wrapped around a simple, high-performance router. This applications programming interface (API) enables developers to develop and extend overlay functionality when the basic DOLR functionality is insufficient.

Hash-based proximity clustering for efficient load balancing in heterogeneous DHT networks

Distributed Hash Table (DHT) networks based on consistent hashing functions have an inherent load uneven distribution problem. The aim of DHT load balancing is to balance the workload of the network node in relation to their capacity to eliminate as TRAF bottleneck. It is difficult because the dynamics, the proximity and heterogeneity natures of DHT networks and time-varying load characteristics. We present a hash-based clustering approach close to the load in heterogeneous DHT compensation. In approach DHT nodes are classified as regular nodes and super nodes according to their computer and network capacity. Regular nodes are grouped and associated with super nodes on consistent hashing of their physical proximity information via the Internet. The super node forms a self-organized and churn elastic support network for load balancing. The hierarchical structure facilitates the design and implementation of a site-aware randomized (LAR) load balancing algorithm. The algorithm provides a factor of randomness in the load balancing method in a range of ambient to deal with both the proximity and dynamism. Simulation results show the superiority of clustering approach with LAR, compared to a number of other DHT load-balancing algorithms. The approach takes no worse than existing proximity aware algorithms and shows a strong resistance to the action of the churn. It also reduces significantly the effort for elastic randomized load balancing through the use of proximity information.

Topologically-Aware Overlay Construction and Server Selection

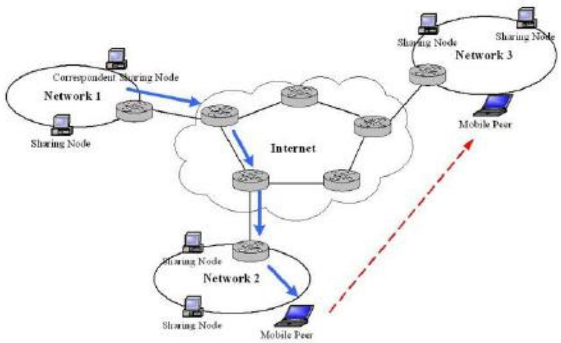
A number of large-scale distributed Internet applications could potentially benefit from some level of knowledge about the relative proximity between its participating host nodes. For example, the performance of large overlay networks could be improved if the application-level connectivity between the nodes in these networks is congruent with the underlying IP-level topology. Similarly, in the case of replicated web content, client nodes could use topological information in selecting one of multiple available servers. For such applications, one need not find the optimal solution in order to achieve significant practical benefits. Thus, these applications, and presumably others like them, do not require exact topological information and can instead use sufficiently informative hints about the relative positions of Internet hosts. In this paper, we present a binning scheme whereby nodes partition themselves into bins such that nodes that fall within a given bin are relatively close to one another in terms of network latency. Our binning strategy is simple (requiring minimal support from any measurement infrastructure), scalable (requiring no form of global knowledge, each node only needs knowledge of a small number of well-known landmark nodes) and completely distributed (requiring no communication or cooperation between the nodes being binned). We apply this binning strategy to the two applications mentioned above: overlay network construction and server selection. We test our binning strategy and its application using simulation and Internet measurement traces. Our results indicate that the performance of these applications can be significantly improved by even the rather coarse-grained knowledge of topology offered by our binning scheme.

3. PROPOSED WORK

We propose a system that integrates a social network into a P2P network, namely Social-P2P, to simultaneously achieve efficient and trustworthy P2P file sharing by leveraging social interests and relationships. Two facts lay the foundation for this work. First, people usually share files that they are interested in peer to peer networks. Interests indicated by a user himself in his profile can more accurately reflect the complete interests of the user. Second, users are unlikely to provide faulty files to their socially close friends because it will impair their social relationships with others and degrade their reputation in their social communities in the real world. Thus, by mapping the P2P cyber network to the social network and restricting cyber services (e.g., file sharing and message routing) between socially close nodes, misbehaviors (i.e., providing faulty files and rejecting forwarding messages) can be discouraged. we develop the following three components for Social-P2P: 1) Interest/trust-based structure construction. 2) Interest/trust-based file searching. 3) Trust relationship adjustment.

4. SYSTEM DESIGN

The system design represents how files are sharing in interested peer to peer networks and also to provide faulty files to their socially close friends to degrade their reputation in the real world.



5. METHOD

1. NETWORK INFRASTRUCTURE

i) Interest/trust-based structure:-

It groups common interests more nodes in a cluster of interest (O1) and forms relatively stable nodes in a Distributed Hash Table (DHT) to connect clusters for efficient inter-cluster data sharing (O4). within each Interest cluster nodes connected to their socially close nodes as P2P overlay neighbors (O3). In addition, social P2P uses anonymous routing of malicious nodes to selectively prevent attack socially remote node and protects the privacy of the node.

ii) Interest/trust-based file searching:-

The trust between the nodes weighted and a node tends to transmit a file search to trusted neighbors (O2). Since higher popularity files have more copies of files in the system, divided odyssey considering in intra-cluster file search for high hit employs more popular files more copies in the system (O5) have. We propose more algorithms to improve routing of random walk-based routing.

iii) Trust relationship adjustment:-

Each node in a routing path takes confidence to the next hop when a malformed file accessed in order to avoid routing requests to downed node later (O2). Social With these three components reached P2P highly efficient and reliable data sharing with low overhead.

2. CLUSTER ROUTING ALGORITHM

i) Intra-Cluster Routing Algorithm:-

A queried file has one or more interests. If at least one query is one concern for the interests of the file selection, it uses the intra-cluster routing algorithm, which submits a query to trusted nodes. We define the social distance between two nodes, such as the number of hops in the shortest path between them on the social network. A reduction in the social distance clearly increases confidence between the nodes. So we use an exponential model to reflect the relationship between trust and social distance.

ii) Inter-Cluster Routing Algorithm:-

Inter-cluster inquiry is required if users have files outside when the query in the current cluster can not query satisfied their interests or. In this case, based on the diverse interests of the requested file, the request generates a request vector to generate an interest vector: calculate their value Hilbert and get assigned to the ID of the cluster with the Hilbert value. Since the cluster ID of the direct More represents interests of the nodes in the cluster, the cluster is associated with the destination cluster that holds the requested file. After I6, the applicant asks first in his friends list his friends, whether they belong to the destination cluster. If so, the file request is sent to the cluster through the friend. Otherwise, the requestor is to transmit the file request to the Ambassador in his current cluster. In this way, the traffic from AMBASSADOR can be greatly reduced.

iii) Enhanced Intra-Cluster Routing Algorithm:-

In the improved routing algorithm, when a node receives a file request, it first checks whether an entry for the requested file in its CRT. If so, the node will furthermore examine whether the confidence weight of the next hop node is greater than a predefined threshold, which is high enough to determine that the node is trustworthy. If so, the request is forwarded to the node. If there is no entry for the requested file or next hop node is not trustworthy, the original random walk routing algorithm is used.

3. DISTRIBUTED HASH TABLE (DHT) FOR INTER-CLUSTER FILE SEARCHING

With DHT routing the request to the Ambassador in the destination cluster is passed. After the file request arrives at the destination cluster, it is routed through the intra-cluster routing algorithm. Two similar vectors can be divided into two adjacent clusters. Therefore, if a query can not be met within the target cluster, it is forwarded in both clockwise to the neighboring cluster and counterclockwise with a cluster TTL (CTTL). Similar inter-cluster routing, attempts a node to send the query about his friends on the social network, rather than ambassadors in the DHT. 6¼ 0 continue to hold Units of the query with TTL ¼ 0 and CTTL the request to its adjacent clusters. If the CTTL expires, the request message is sent to the server to make the file holder identified. Each node in the system reports files that are to rarely queried by other to the server, the file to ensure availability. Such files are the files with attendance rates less than a predetermined threshold.

4. PERFORMANCE EVALUATION

i) Evaluation of File Sharing Efficiency:-

Traffic distribution "intra" refers to the share of traffic in inner Clustering search. "Inter-one-hop" and "Inter-Two-hop" denote the percentage of traffic in inter-cluster search, if the target clusters one and two cluster hops away from the source cluster, respectively. The number of cluster hops describes the distance between two clusters by the number of clusters measured. A cluster is a cluster from its adjacent cluster Hop. The inter-cluster user traffic if the target is> 2 Cluster hops away and the traffic on the server are reported under "Other".

ii) Evaluation of File Sharing Trustworthiness:-

We studied the inquiring trustworthiness of social P2P compared with Pure-P2P and own Trust. Social-P2P is investigated by two mechanisms: (1) The routing is not anonymous. Post Malicious nodes no faulty files on their socially close nodes, but they send erroneous files to the requester three hops away on the social network. This mechanism is called "social P2P NA". (2) The routing is anonymous. Malicious nodes are informal and respond to each request with a malformed file. This mechanism is called "social P2P". To make the methods similar to, use all systems of direct interest More Cluster mechanism for file sharing.

7. CONCLUSION

Social-P2P that synergistically integrates a social network into a P2P network for efficient and trustworthy file sharing. Taking advantage of the interest information in the social network, the socially close nodes with similar multi-interests are clustered together. Nodes are connected with their friends within a cluster. Within each cluster, trust-based routing algorithms are proposed to forward a query message along trustworthy links, enhancing file searching efficiency and trustworthiness. Comparatively stable nodes from clusters form a DHT for inter-cluster communication. Nodes also decrease the trust weights of links to their neighbors which have high probability to forward messages to misbehaving nodes. The experimental results from trace driven simulations and the prototype on PlanetLab demonstrate the efficiency and trustworthiness of file sharing in Social-P2P in comparison with other file sharing systems.

8. FUTURE WORK

As a new method to ensure low-cost and secure file sharing in P2P networks, Social-P2P still has some shortcomings. First, the establishment and maintenance of weighted transaction network consumes high resources. Thus, a future work is to reduce the maintenance cost. Second, calculating valid paths from a server to a client in the weighted transaction network to check suspicious transactions may take a long time. We attempt to reduce this calculation time. Third, implementation of the centralized trust center may cause some problems such as DoS attacks. A powerful malicious node can compromise the trust center and launch DoS attacks to make the entire system mulfunction. Therefore, making Social-P2P completely distributed is another direction to improve the system. This is a formidable challenge because the trust center not only replies to reputation queries between non-friend nodes, but also maintains sensitive personal information (i.e., user's ID, name, relationships with other users). It is quite difficult to remove the trustable centralized trust center since participating nodes are not totally reliable for protecting information and reporting their weighted links honestly. Fourth, our experiments show that including friend transactions to the weighted transaction network may mislead clients to select malicious nodes as servers since they still provide high-QoS files to their friends. Therefore, we will explore a method to solve this problem.

9.REFERENCES

1. *Stoica. I, Morris. R, Liben-Nowell. D, Karger. D. R, Kaashoek. M. F, Dabek. F, and Bala krishnan. H. (Feb.2003), "Chord: A scalable peer-topeer lookup protocol for internet applications," IEEE/ACM Trans. Netw., vol. 11, no. 1, p.p: 17–32.*
2. *Rowstron. A and Druschel. P. (2001), "Pastry: Scalable, decentralized object location and routing for large-scale peer-to-peer systems," in Proc. IFIP/ACM Int. Conf. Distrib. Syst. Platforms Heidelberg, , p.p: 329–350.*
3. *Zhao. B. Y, Huang. L, Stribling. J, Rhea. S. C, Joseph. A. D, and Kubiatowicz. J. (2004), "Tapestry: A resilient global-scale overlay for service deployment," IEEE J. Sel. Areas Commun., vol. 22, no. 1, p.p:41–53.*
4. *Shen. H, Xu. C and Chen. G. (2006), "Cycloid: A scalable constant-degree P2P overlay network," Perform. Eval, vol. 63, p.p: 195–216.*
5. *Li. Z, Xie. G and Li. Z. (Dec.2008), "Efficient and scalable consistency maintenance for heterogeneous peer-to-peer systems," IEEE Trans. Parallel Distrib. Syst., vol. 19, no. 12, p.p:1695–1708.*
6. *Shen. H and Xu. C. Z.(2008), "Hash-based proximity clustering for efficient load balancing in heterogeneous DHT networks," J. Parallel Distrib. Comput., vol. 68, p.p: 686–702.*
7. *FastTrack. (2003) [Online]. Available: http://www.fasttrack.nu/ index_int.html*
8. *Ratnasamy. S, Handley. M, Karp. R, and Shenker. S. (2002), "Topologically-aware overlay construction and server selection," in Proc. IEEE INFOCOM, p.p: 1190–1199.*
9. *Waldvogel. M and Rinaldi. R. (2002), "Efficient topology-aware overlay network," in Proc. ACM Workshop Hot Topics Netw., p.p: 101–106.*
10. *Zhu. Y and Shen. H. (2008), "An efficient and scalable framework for content-based publish/subscribe systems," Peer-to-Per Netw. Appl., vol. 1, p.p: 3–17.*
11. *Hang. C and Sia. K. C. (2002), "Peer clustering*

and firework query model," in Proc. Int. World Wide Web Conf.

12. *Crespo. A and Garcia-Molina. H. (2002), "Routing indices for peer-topeer systems," in Proc. 22nd Int. Conf. Distrib. Comput. Syst., p.p: 23–32.*
13. *Nejdl. W, Wolpers. M, Siberski. W, Schmitz. C, Schlosser. M, Brunkhorst. I, and Loeser. A. (2004), "Super-peer-based routing and clustering strategies for RDF-based peer-to-peer networks," J. Web Semantics, vol. 1, no. 2, p.p: 137–240.*
14. *Bernstein. P. A, Giunchiglia. F, Kementsietsidis. A, Mylopoulos. J, L. Serafini. L, and Zaihrayeu. I. (2002), "Data management for peer-to-peer computing: A vision," in Proc. 5th Int. Workshop World Wide Web Databases, p.p: 89–94.*
15. *Halevy. A. Y, Ives. Z. G, Mork. P, and Tatarinov. I.(2003), "Piazza: Data management infrastructure for semantic web applications," in Proc. 12th Int. Conf. World Wide Web, p.p: 556–567.*
16. *Aberer. K, Cudr e-Mauroux. P, and Hauswirth. M. (2003), "The chatty web: Emergent semantics through gossiping," in Proc. 12th Int. Conf. World Wide Web, p.p: 197–206.*
17. *Morpheus. (2008) [Online]. Available: http://www.musiccity. Com*
18. *Garbacki. P, Epema. D. H. J, and van Steen. M. (2007), "Optimizing peer relationships in a super-peer network," in Proc. Int. Conf. Distrib.Comput. Syst., p.p: 31.*
19. *Mitra. B, Dubey. A. K, Ghose. S , and Ganguly. N. (2010), "How do superpeer networks emerge?" in Proc. INFOCOM, p.p: 1–9.*
20. *Hofstatter. Q, Zols. S, Michel. M, Despotovic. Z, and Kellerer. W. (2008), "Chordella—A hierarchical peer-to-peer overlay implementation for heterogeneous, mobile environments," in Proc. 8th Int. Conf. Peer-to-Peer Comput., p.p: 75–76.*
21. *Sachez-Artigaz. M, Garcia-Lopez. P, and Skarmeta. A. F. G. (2008), "On the feasibility of dynamic superpeer ratio maintenance," in Proc. 8th Int. Conf. Peer-to-Peer Comput., p.p: 333–342.*
22. *Liu. H, Thomas. J, and Khethavath. P. (2013), "Moving target with load balancing in P2P cloud," in Proc. IEEE 6th Int. Conf. Cloud Comput., p.p: 359–366.*
23. *Garbacki. P, Epema. D. H. J, and Steen. M. V. (2010), "The design and evaluation of a self-organizing superpeer network," IEEE Trans. Comput., vol. 59, no. 3, p.p: 317–331.*
24. *Ratnasamy. S, Francis. P, Handley. M, Karp. R, and Shenker. S. (2001), "A scalable content-addressable network," in Proc. ACM SIGCOMM, p.p: 329–350.*
25. *Genaud. S and Rattanapoka. C. (2008), "Large-scale experiment of coallocation strategies for peer-to-peer supercomputing in P2P MPI, "inProc.IEEEInt.Symp.ParallelDistrib.Symp, p.p: 1–8.*