A Multi-dimensional Reputation System Combined with Trust and Incentive Mechanisms in P2P File Sharing Systems *

Mao Yang[21], Qinyuan Feng[1], Yafei Dai[1], Zheng Zhang[2]
Peking University, Beijing, China[1]
Microsoft Research Asia, Beijing, China[2]
(maoyang@microsoft.com, {fqy, dyf}@net.pku.edu.cn, zheng.zhang@microsoft.com)

Abstract

Free-riders and fake files are two important problems in P2P file sharing systems. Previous works have always used incentive mechanisms and trust mechanisms to address them respectively. In real systems however, a trust mechanism without incentive would face lack of users' enthusiasm and thus cause sparse relationship of direct trust while an incentive mechanism without trust could induce users' bad behavior. A novel reputation system is proposed in this paper that combines trust and incentive mechanisms. It uses files' vote and retention time, download volume and users' rank to construct a more extensive direct trust relationship and calculates a user's reputation with a multi-trust algorithm. It can identify fake files and provide service differentiation with reputation. Implementation and some security consideration in DHT are also discussed.

Keywords: trust, incentive, reputation, p2p

1. Introduction

Free-riders and fake files are two important problems in P2P file sharing systems. Earlier research focused on free-riders which result from concern for sharing risk and the lack of motivation to share, so incentive mechanisms are used to encourage users to share. Q. Lian et al. [13] evaluated several incentive mechanisms, proposed an incentive system which is a hybrid between Tit-for-Tat and EigenTrust, and showed its effectiveness of generating preferences for well behaved nodes while correctly punishing colluders. Though it can conquer the pitfalls that exist in private history-based Tit-for-Tat and the EigenTrust algorithm, it still faces the coverage problem. This means it needs more direct trust relationship between users to improve the request coverage and get more accurate results.

Recent research focuses on fake files. J. Liang et al. [4] analyzed the severe pollution which is launched by some companies to protect copyrights in KaZaA, while Q. Feng et al. [3] analyzed the severe pollution which is launched by some users to gain unfair advantages from incentive policies in Maze . Fake files are more and more pervasive in these systems and nearly half of the files of some popular titles are fake. Reputation mechanisms, which collect the evaluation of files and users, are used to identify fake files and attackers. There are mainly two kinds of evaluations.

- Explicit evaluation involves active user participation such as voting [5]. . This method is widely used in electronic commerce but users lack the impetus to vote in P2P file sharing systems. As an example, less than 1% of the popular files on KaZaA are voted on [4]. Therefore, an incentive mechanism is also needed to encourage voting as well as sharing.
- o Implicit evaluation means the system infers a user's evaluation from their behavior. Q. Feng et al. [3] proposed a lifetime and popularity based ranking approach to filter out fake files in P2P file sharing systems. However this method cannot identify the quality of a file accurately when its number of owners is too small. The actually play time of a movie file can also be taken as a user's evaluation for this file, but it depends on the type of file.

A multi-dimensional reputation system combined with trust and incentive mechanisms in P2P file sharing systems is proposed in this paper, its contributions are:

* Supported by the National Natural Science Foundation of China under Grant No. 60673183; the National Grand Fundamental Research 973 Program of China under Grant No. 2004CB318204; the Ph. D. Programs Foundation of Ministry of Education of China under Grant No. 20060001044

1. It combines explicit and implicit evaluations, which include files' vote and retention time, download volume and users' rank, to construct a more extensive direct trust relationship.
2. It combines trust and incentive mechanisms, and uses service differentiation based on users' reputations to encourage users to share and vote on files, rank users and remove fake files quickly, and helps the system identify fake files at the same time.

The road map of this paper is as follows. In Section 2, we cover the related works. We propose a multi-dimensional reputation system combined with trust and incentive mechanisms in Section 3 and discuss its implementations and some security considerations in DHT in Section 4. We draw a brief conclusion and propose some future research possibilities in Section 5.

2. Related Works

Most of the current reputation systems are variations of Tit-for-Tat [6] [7] and EigenTrust [2].

- Tit-for-tat [1] is based on private history and a peer giving higher priority to those from who he has successfully downloaded more. Q. Lian et al. [13] showed that even with a long private history it is difficult to improve request coverage. A one month download log only enforces Tit-for-Tat to only 2% of a peer's uploads and the other 98% are blind uploads.
- EigenTrust works similarly to the PageRank [8] algorithm used by Google. The page link in the PageRank algorithm becomes traffic flow in EigenTrust. It assigns each peer a global EigenRank value by computing the left principle eigenvector of the trust matrix. Q. Lian et al.[13] also found that it suffers from both false negatives and false positives.

Q. Lian et al. [13] proposed a multi-trust solution to achieve a balance between the two reputation mechanisms. It uses direct trust relationship to construct a one-step trust matrix which is tit-for-tat and uses this matrix to construct a two-step or n-step trust matrix which is similar to EigenTrust. The insight is that when deriving the incentive metric for service differentiation, it needs to ideally consider all these matrices instead of just one. The immediate friends form the first tier, friends' friends form the next and so on. Each matrix precisely represents the trust that a peer imposes on others at a different level and as the levels go deeper, the trust becomes more global and less private. This multi-tier incentive scheme essentially imposes service differentiation by looking at which tier U_j falls into when its downloading request arrives at U_i. The smaller level the user belongs to, the higher priority they are given. Within the same tier, two peers will be ranked according to their values in the matrix of that tier. However, it does not solve the one-step sparse matrix problem which means that if the one-step matrix is too sparse, it will need a lot of steps to get adequate request coverage.

There are also many other works such as P2Prep [9], XRep[10], PeerTrust [11], TrustGuard [12], Credence [5] and LIP [3]. Most of these woks discuss incentive and trust mechanisms separately. In a real situation, a trust mechanism without incentive would face lack of users' enthusiasm and sparse relationship of direct trust while an incentive mechanism without trust could induce users' bad behavior. Our work combines trust and incentive mechanisms and creates a more active and trustworthy P2P network. Our work also combines explicit and implicit evaluations to construct a more extensive direct trust relationship between users.

3. Design of Reputation System

From the analysis above, a novel reputation system is designed in the following way. We first use files' vote and retention time, download volume and users' rank to construct a denser one-step trust matrix and calculate users' reputations with n-step multi-trust. Based on the reputations, we identify fake files and use service differentiation to encourage users' active participation which will also increase the denseness of the one-step trust matrix.

3.1. Direct Trust Relationship

File, download volume and user will be used separately to construct a multi-dimensional direct trust relationship.

3.1.1. File Based Direct Trust Relationship

In P2P file sharing systems, users exchange files with others and evaluate the files they have. The evaluation can be mapped into [0,1] with 1 means the best and 0 means the worst.

A file can be evaluated explicitly and implicitly.

- Explicit evaluation can be collected by users' votes on files. Most of the traditional vote-based reputation mechanisms use this method. It can reflect

a user's evaluation of files more accurately but requires users' participation and imposes a burden on them.

- Implicit evaluation can be collect by a file's retention time in a user's computer. It doesn't need users to participate but its error might be large because the evaluation is only inferred from a user's behavior.

Our work calculates a file's evaluation by an integration of the two evaluations. U_i's evaluation of file j is calculated by Equation (1), while IE_{ij} means U_i's implicit evaluation of file j, EE_{ij} means U_i's explicit evaluation of file j, η and ρ are weight values, and $\eta+\rho = 1$. However we still need to use an incentive mechanism to encourage users to vote because it is more accurate.

$$E_{ij} = \begin{cases} IE_{ij} & \text{if a user doesn't vote} \\ IE_{ij} \cdot \eta + EE_{ij} \cdot \rho & \text{if a user votes} \end{cases} \quad (1)$$

A file's evaluation generally means a user's opinion so if two users have similar evaluations of files, we infer that they have some direct trust relationship. If U_i and U_j both have evaluated some files and the intersection of these files is F with size m, we define file based direct trust relationship with Equation (2)[1].

$$FT_{ij} = 1 - \frac{\sum_{k \in F} |E_{ik} - E_{jk}|}{m} \quad (2)$$

It is obvious that $FT_{ij} \in [0, 1]$. A large FT_{ij} means the two users' opinions are similar; a small FT_{ij} means the two users' opinions are different. If the size of intersection is zero, the two users will not have file based direct trust relationship.

We then define file based one-step matrix(FM) with Equation (3), while U_{all} means all the users in the system.

$$FM_{ij} = \frac{FT_{ij}}{\sum_{k \in U_{all}} FT_{ik}} \quad (3)$$

3.1.2. Download Volume Based Direct Trust Relationship

In P2P file sharing systems, if a user downloads some real file from another user, it means he can trust this user so valid download volume can be used to construct implicit direct trust relationship. We define VD_{ij} as valid download volume that U_i has downloaded from U_j with Equation (4), while D_{ij} means all the files that U_i has downloaded from U_j, E_{ik} means U_i's evaluation of file k, and S_k means the size of file k. We then define download volume based direct trust matrix(DM) with Equation (5).

$$VD_{ij} = \sum_{k \in D_{ij}} E_{ik} \cdot S_k \quad (4)$$

$$DM_{ij} = \frac{VD_{ij}}{\sum_{k \in U_{all}} VD_{ik}} \quad (5)$$

3.1.3. User Based Direct Trust Relationship

In P2P file sharing systems, users can also evaluate each other and we use UT to represent this relationship. This can be realized by assigning a user with a value. This can also be done by friend lists and blacklists, which means a user's friends should be more trustworthy and they should be assigned with a large UT, while the users in the blacklist are not as trustworthy and they should be assigned with zero. We then define user based one-step matrix(UM) with Equation (6).

$$UM_{ij} = \frac{UT_{ij}}{\sum_{k \in U_{all}} UT_{ik}} \quad (6)$$

3.1.4. Integration of Direct Trust Relationship

We can get an integrated direct trust relationship from the three relationships above and form one-step direct trust matrix (TM) with Equation (7), while α, β and γ mean the weight value of each direct trust relationship respectively and $\alpha + \beta + \gamma = 1$. When there are more methods to get direct trust relationship, this equation can be extended easily.

$$TM = \alpha \cdot FM + \beta \cdot DM + \gamma \cdot UM \quad (7)$$

3.2. The Calculation of Reputation

We use multi-trust to calculate the reputation between users. We define reputation matrix(RM) with Equation (8).

$$RM = TM^n \quad (8)$$

In order to choose n, we do an experiment in Maze [15], which is a large deployed P2P file sharing system with more than 2 million registered users and more than 10,000 users online at any given time. A log server is used to record every downloading action and each log contains uploading user-ID, downloading user-ID,

1 There are also many other equations to define the distance between two vectors, such as Kullback-Leibler distance and Euclid distance.

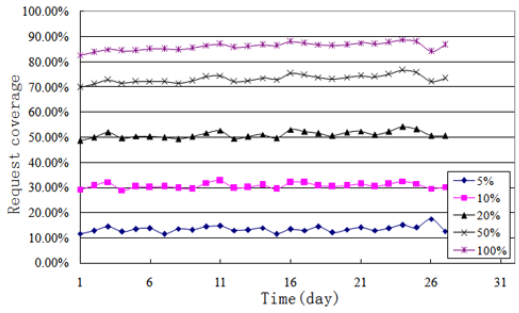

Figure 1. Request coverage with different evaluation coverage

global time, files content hash, and filename. We collect the logs of 30 days, which include 103825 users, 24,656,024 downloading actions and 395218 different files.

We first set the evaluation coverage to be k%, meaning each user will evaluate k percent of his files randomly, then replay the downloading actions to see how many download requests will be covered. A download request is covered means a file based direct trust relationship can be constructed from the uploader to the downloader with the files they have evaluated.

In Figure 1, the x-axis means the time, the y-axis means the request coverage's change with time when users evaluate different percentage of the files. We can conclude from the figure that when users only evaluate 5% of the files, the request coverage is small;when users evaluate 20% of the files, the request coverage reaches 50%. Because we use a file's retention time to collect implicit evaluation, users will evaluate 100% of the files they have, resulting in the request coverage being above 80%. In addition, download volume and user based direct trust relationships can also increase request coverage. We can choose n as 1 in Maze, which means the one-step direct trust matrix is enough for Maze. However, multi-trust can be easily extended to an n-step direct trust matrix to adapt to other P2P networks.

We can also find that the request coverage will not change significantly with time. It originates from the churn of users and files so we only need to store the evaluations within an interval.

3.3. Identification of Fake Files

Most of the current reputation systems consider two trust values; one is about a user's performance while the other is about a user's feedback trustworthiness. In our reputation system, only the one who performs well and gives honest feedback can get a high reputation, the reputation between users can be used to identify fake files directly.

Before U_i downloads a file, he can get some other users' evaluations of this file. He can then calculate the reputation of this file with Equation (9), while U means the set of users from whom he gets the file's evaluations. Then he can judge whether to download this file by the threshold set by himself. More details will be described in the next section.

$$R_f = \frac{\sum_{j \in U} RM_{ij} * E_{jf}}{\sum_{j \in U} RM_{ij}} \tag{9}$$

3.4. Trust Based Incentive Mechanism

The system features service differentiation based on reputation. It is designed to give downloading preference to users with high reputations. These users add to their request time a negative offset whose magnitude grows with their reputation. In contrast, a bandwidth quota is applied to downloads of users with lower reputations. Different from other reputation systems, uploading real files, voting on files and ranking other users honestly and even deleting fake files quicker can increase a user's reputation and give him better service. This reputation system does not only identify fake files but also prevents users sharing fake files intentionally and unintentionally.

4. Implementation in P2P File Sharing Systems

In P2P networks, the most important problem is how to store and retrieve users' evaluations of files effectively. It is easy implemented with central servers so we only discuss the situation without central servers.

4.1. Implementation in DHT

DHT is used in P2P networks to store index object and other information. It provides a basic operation: Lookup(key, list<HostInfo>& Hosts, int num = 1). When a user publishes a file, he will use DHT to lookup one or more users who are next to the key and store this file's information in them. When other users need this file, they can also use lookup to find the information about this file. We can store and retrieve a file's evaluation in DHT in the same way, showen in Figure 2.

1. Publication of a file's evaluation: In DHT, we can include a file's evaluation with its publication so

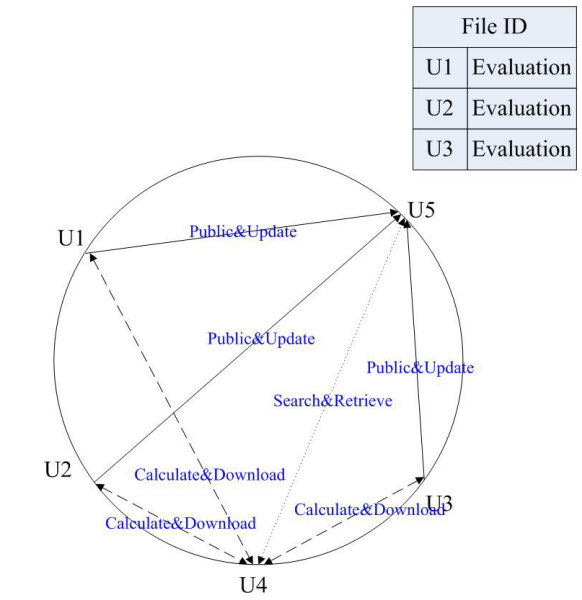

Figure 2. Framework in DHT based overlay

we do not only publish a file's metadata to index peer but also this user's evaluation of the file. The message is EvaluationInfo = <FileID, OwnerID, Evaluation, Signature>.

2. Update of a file's evaluation: This can be done with the regular republication.

3. Retrieval of a file's evaluations: When a user wants to download a file, he finds this file's index peer first and then retrieves the information of this file's owners. He will also get an array of evaluation information.

4. Calculation of a user's reputation: A user can contact the user he wants to evaluate to get his evaluation list and calculate TM, he can then calculate RM with multi-trust.

5. Calculation of a file's reputation: A user can use the reputations and evaluations of the users in the array of evaluation information to calculate a file's reputation.

6. Service differentiation: A user can give different service to users with different reputations. In Figure 2, U_4 requests other users to download a file from them. Other users can calculate U_4's reputation and give U_4 a suitable service which includes a bandwidth quota and the position in the waiting queue.

4.2. Security Issues

There are some possible attacks:

1. A user may forge or distort other user's evaluation in step 1, 2 and 3: This can be solved by digital signature.
2. U_5 may remove or does not answer any query in step 3: This is analyzed in [14] and it is generally treated as a problem of routing security and is beyond the scope of this paper.
3. U_4 may forge his files' evaluations as the same as U_1 to get U_1's trust in step 6: G. Swamynathan et al. [14] suggested a virtual user examine other users' evaluations randomly. If there are great differences between two examinations, it means this user has forged his evaluations and he should be punished.
4. Some users may collaborate to increase their reputation: This was analyzed in [13].

4.3. Other Issues

In a real P2P network, users may join and leave the system frequently and churn may affect data's availability. As we have described, a file's evaluation information can be stored and published with index information so the designer only needs to consider how to optimize the publication of index information. This will not increase the complexity of the system. There are many techniques to reduce the effect of churn. Take eMule for example, a user will publish index information to multi-users regularly.

In addition, the system will not need more lookup messages when a user publishes and retrieves a file's evaluation with this file's index information, though it will increase the size of the information slightly.

In a real P2P file sharing system, most of the messages are publication information while search and request messages only occupy a small proportion. However, if a user has evaluated a lot of files, he may send more messages when he exchanges files' evaluations. Most files' numbers of owners are small and most files have a small life cycle which is also shown in 1. So users only need to preserve the evaluations within an interval when they have evaluated so many files.

5. Conclusion and Future Works

A multi-dimensional reputation system combined with trust and incentive mechanisms in P2P file sharing systems is proposed. It can not only identify fake files but also encourage users to upload real files, vote

on files and rank users honestly as well as removing fake files quickly. Implementation and some security considerations are also discussed.

In the future, we need to do more experiments to improve the equations and choose the weight values in our work and deploy this framework in a real system.

References

[1] R. Axelrod, "The Evolution of Cooperation", New York: Basic Books, 1984.

[2] S. D. Kamvar, M. T. Schlosser and H. Garcia-Molina, "The EigenTrust Algorithm for Reputation Management in P2P Networks" In Proceedings of WWW, May 2003

[3] Q. Feng, Y. Dai, LIP: A Lifetime and Popularity Based Ranking Approach to Filter out Fake Files in P2P File Sharing Systems, In Proc. of IPTPS, Bellevue, Washington, 2007.

[4] J. Liang, R. Kumar, Y. Xi, and K. Ross, Pollution in P2P file sharing systems, Proc. IEEE INFOCOM'05, Miami, FL, 2005.

[5] K. Walsh and E. G. Sirer, Experience with an Object Reputation System for Peer-to-Peer Filesharing, USENIX NSDI, 2006.

[6] B. Cohen, "Incentives Build Robustness in BitTorrent," In Proc. of P2P-Econ, June 2003.

[7] Y. Kulbak, and D. Bickson, "The eMule Protocol Specication", eMule project, http://sourceforge.net

[8] S. Brin, L. Page, "The anatomy of a large-scale hypertextual Web search engine," In Proc. of WWW, April 1998.

[9] F. Cornelli, E. Damiani and S. De Capitani, Choosing Reputable Servents in a P2P Network, In Proceedings of the 11th World Wide Web Conference, Hawaii, USA, 2002.

[10] E. Damiani, S. Paraboschi, P. Samarati and F. Violante, A reputation-based approach for choosing reliable resources in peer-to-peer networks, Proceedings of the 9th ACM conference on Computer and communications security, 2002, pp. 207-216.

[11] L. Xiong and L. Liu, Peertrust: Supporting reputation-based trust for peer-to-peer electronic communities, IEEE Transactions on Knowledge and Data Engineering, 16 (2004), pp. 843-857.

[12] M. Srivatsa, L. Xiong and L. Liu, TrustGuard: countering vulnerabilities in reputation management for decentralized overlay networks, Proceedings of the 14th international conference on World Wide Web (2005), pp. 422-431.

[13] Q. Lian, Z. Zhang, M. Yang, B. Y. Zhao, Y. Dai, and X. Li, "An Empirical Study of Collusion Behavior in the Maze P2P File-Sharing System". Microsoft Research Technical Report, MSR-TR-2006-14, 2006.

[14] G. Swamynathan, Ben Y. Zhao and Kevin C. Almeroth, Exploring the Feasibility of Proactive Reputations. of IPTPS, Santa Barbara, CA, USA. February 2006.

[15] M. Yang, H. Chen, B. Y. Zhao, Y. Dai, and Z. Zhang, "Deployment of a Large-scale Peer-to-Peer Social Network," In Pro-ceedings of WORLDS, San Francisco, CA, Dec. 2004.