TRUST-AWARE INFORMATION RETRIEVAL IN PEER-TO-PEER ENVIRONMENTS

A thesis submitted to the University of Manchester
for the degree of Doctor of Philosophy
in the Faculty of Engineering and Physical Sciences

2011

By
Ye Zhang
School of Computer Science

Contents

List of Tables

List of Figures

Abstract

Information Retrieval in P2P environments (P2PIR) has become an active field of research due to the observation that P2P architectures have the potential to become as appealing as traditional centralised architectures. P2P networks are formed with voluntary peers that exchange information and accomplish various tasks. Some of them may be malicious peers spreading untrustworthy resources. However, existing P2PIR systems only focus on finding relevant documents, while trustworthiness of documents and document providers has been ignored. Without prior experience and knowledge about the network, users run the risk to review, download and use untrustworthy documents, even if these documents are relevant.

The work presented in this dissertation provide the first integrated framework for trust-aware Information Retrieval in P2P environments, which can retrieve not only relevant but also trustworthy documents. The proposed content trust models extend an existing P2P trust management system, PeerTrust, in the context of P2PIR to compute the trust values of documents and document providers for given queries. A method is proposed to estimate global term statistics which are integrated with existing relevance-based approaches for document ranking and peer selection. Different approaches are explored to find optimal parameter settings in the proposed trust-aware P2PIR systems. Moreover, system architectures and data management protocols are designed to implement the proposed trust-aware P2PIR systems in structured P2P networks.

The experimental evaluation demonstrates that P2PIR can benefit from trust-aware P2PIR systems significantly. It can importantly reduce the possibility of untrustworthy documents in the top-ranked result list. The proposed estimated global term statistics can provide acceptable and competitive retrieval accuracy within different P2PIR scenarios.

Declaration

No portion of the work referred to in this thesis has been submitted in support of an application for another degree or qualification of this or any other university or other institute of learning.

Copyright

i. The author of this thesis (including any appendices and/or schedules to this thesis) owns any copyright in it (the "Copyright") and s/he has given The University of Manchester the right to use such Copyright for any administrative, promotional, educational and/or teaching purposes.

ii. Copies of this thesis, either in full or in extracts, may be made only in accordance with the regulations of the John Rylands University Library of Manchester. Details of these regulations may be obtained from the Librarian. This page must form part of any such copies made.

iii. The ownership of any patents, designs, trade marks and any and all other intellectual property rights except for the Copyright (the "Intellectual Property Rights") and any reproductions of copyright works, for example graphs and tables ("Reproductions"), which may be described in this thesis, may not be owned by the author and may be owned by third parties. Such Intellectual Property Rights and Reproductions cannot and must not be made available for use without the prior written permission of the owner(s) of the relevant Intellectual Property Rights and/or Reproductions.

iv. Further information on the conditions under which disclosure, publication and exploitation of this thesis, the Copyright and any Intellectual Property Rights and/or Reproductions described in it may take place is available from the Head of School of Computer Science (or the Vice-President).

Acknowledgements

I would like to take this opportunity to express my gratitude to my supervisor Dr. Rizos Sakellariou, for his invaluable advice, direction, patience, support and insights throughout both my PhD work and my life.

I'm grateful to my parents, my father Mr. Kewen Zhang and my mother Mrs. Limei Wang. My father was passed away by a car accident during my PhD study. Without his strong support and encouragement I cannot achieve this. I also thank my lovely girlfriend Miss. Jie Zhou for her love and support throughout my studies.

I would also like to thank my advisor Dr. Suzanne Embury and her colleagues who generously gave advice and help throughout this work, in particular Dr. Alvaro Fernandes who is one of my second year interview examiners.

I also thank my friends and collegues Shuo, Deyu, Lei, Henan, Tianyi, Yaoyao, Xuejiao, Chao, Jun, Jing, Naikuo, Hongfen, Chenchen, Sky, Wei, Syed, Serafeim and Viktor.

Last but by no means least, this work would not be possible without the sponsorships from the UK Overseas Research Students Awards Scheme (ORSAS) and from the School of Computer Science, University of Manchester, UK.

Chapter 1

Introduction

1.1 Information Retrieval in P2P Networks

A Peer-to-Peer (P2P) network is composed of a number of distributed network computers which can contribute their resources (e.g., storage, processing power, and file) to other network computers without the need for central coordination instances [Sch01]. In a P2P network, any computer can assume multiple roles simultaneously, such as that of a *resource provider, resource consumer*, and *directory service*. Computers form self-organising overlay networks which can communicate with each other through Internet Protocols. Currently, a wealth of P2P applications have been designed, developed and widely used for file sharing (e.g., eMule [eMu], MLDonkey [mld]) and media steaming [1](e.g., Gnustream [JDXB03], MyP2P [myp], SopCast [sop]).

Information Retrieval (IR) is a method to retrieve documents or information within documents to satisfy a user's information need. Unlike *data retrieval* systems (e.g., DBMS) which deal with structured data with well-defined semantics, the term Information Retrieval usually refers to unstructured data (e.g., natural language documents) or semi-structured data (e.g., XML documents) [Bun97] which could be semantically ambiguous [BYRN99] [2]. Clearly, one central problem of IR systems is the issue of predicting which documents satisfy a user's request. Such a decision is usually based on a ranking algorithm which attempts to establish a simple ordering of the retrieved documents by relevance degrees [BYRN99].

[1]Media streaming is when multimedia files start to play by constantly receiving data flows from the network [YV07].

[2]Further discussion of the differences between information retrieval and data retrieval can be found in Section 1.1.1 of the book [BYRN99].

Today, Information Retrieval techniques are extensively used; for example, Web search engines for the Internet (e.g., Google [goo]), or content-based information retrieval systems for digital libraries (e.g., ACM digital libraries [ACM]). Most traditional IR systems typically assume centralised control. They either store all document copies into a central server or collect the directory information of all of the distributed collections of the network into a centralised directory service.

Information Retrieval in P2P networks (P2PIR) became an active field of research in the last decade. This is due to the observation that P2P architectures may have the potential to become an attractive alternative to traditional centralised architectures. Some of the reasons for this are as follows [CAN02, Mal03]:

- Since any peer in a P2P network can be assumed to be an information provider and directory service at the same time, P2P networks provide more opportunities for replication and minimise the chance of having a single point of failure. This indicates that, if one peer breaks down, the rest of peers in the network are still able to work.

- P2P networks are composed of a large number of distributed computers which share part of their own resources (e.g., bandwidth and storage) [LCP+05]. This makes it easier for P2P networks to avoid bottlenecks, such as traffic overload, because they can distribute data and balance requests across the network without using a central server.

- P2P networks can scale better than centralised networks because P2P networks take advantage of the unused resources in each peer, such as processing power and storage. As peers arrive and the demand on the system increases, the total capacity of the system also increases [LCP+05]. This flexibility is not the case in centralised networks.

- Information in the network can be updated effectively and efficiently because data is stored in a highly distributed manner, and can be maintained locally.

- P2P networks are appealing from an economic perspective since they do not need expensive infrastructures or high maintenance costs to manage information.

Considering the above five reasons, P2P platforms may become a possible choice for next-generation search engines, which will be dealing with huge amounts

of *distributed* and *dynamic* data [CF04]. As such, there is interest in studying P2PIR.

To study Information Retrieval in P2P networks, a formalised model is proposed by extending the traditional IR model in [BYRN99]. An Information Retrieval model in P2P networks can be formally defined by a quadruple $/D, Q, P, S(d_j, q_i, p_k)/$, where:

- D is a set composed of all shared documents in a P2P network;
- Q is a set composed of all possible user queries;
- P is a set of peers that are providing documents in the network; and
- $S(d_j, q_i, p_k)$ is a ranking function which represents a score for a document d_j provided by a peer p_k to a given query q_i, where the query $q_i \in Q$, the document $d_j \in D$, and the peer $p_k \in P$. Such ranking defines an order among the retrieved documents provided by different peers in response to the query q_i.

Two major scenarios have been extensively studied in traditional IR environments, which are *cooperative* and *uncooperative* Information Retrieval environments [BYRN99, goo, Cal00, SC03]. These two scenarios are defined by the degrees of cooperation provided by document providers. For example, each document provider in a network can cooperate closely to provide their document copies or content statistics for document retrieval and ranking without access limitations, and this is defined as *cooperative Information Retrieval* (e.g., Web search engines). On the contrary, document providers may not be willing to provide their individual document copies due to copyright issues and access limitations (e.g., payment is required). Moreover, each of these document providers may employ an individual search engine for document indexing, retrieving and ranking, which is often the case in digital libraries. In such an environment, detailed techniques of search engines may not be public, such as ranking algorithms and stemming algorithms. This scenario is defined as *uncooperative Information Retrieval*, and is typically encountered as in distributed digital libraries.

Similar to those problems studied in traditional IR environments, cooperative and uncooperative Information Retrieval scenarios have been explored in P2P networks, and these can be defined as *cooperative Information Retrieval in P2P networks* (cooperative P2PIR) and *uncooperative Information Retrieval in*

P2P networks (uncooperative P2PIR) in this dissertation. For example, P2P Web search [LLH+03, SYYW03, KWTA07, LJT07, JNC06] can be referred to as cooperative P2PIR, and P2P digital library search [ZCLL04, LC06, LC05, NF07a, NFTN05] can be grouped into uncooperative P2PIR. Both cooperative P2PIR and uncooperative P2PIR offer different possibilities and difficulties for distributed search solutions in P2P networks.

1.2 The Problem

1.2.1 Motivation

The primary goal of Information Retrieval systems is to retrieve a list of documents with ranks which are most likely to satisfy selected metrics (e.g., relevance). Then, a few of these documents are selected by users as final answers. Basically, P2P networks open up more ways for document selection metrics, which are fundamentally assumed to be perfect in traditional Information Retrieval environments, but are important factors in P2P environments, such as the *trustworthiness* of documents [LZL06].

P2P networks lack any centralised infrastructure, but rather rely on voluntary peers to exchange information and accomplish tasks [ATS04]. Some of these peers may be malicious peers and spread untrustworthy resources (e.g., documents and surrogates) [Lyn01]. An untrustworthy document could be partially, or entirely, different from the advertising information in the network, and even worse, it might be a newly deployed virus [DGM+03]. Without prior experience and knowledge about the network, users run the risk of reviewing, downloading and using untrustworthy documents, even if those documents are relevant. Thus, it is necessary to integrate the issue of trustworthiness into the P2PIR systems [Lyn01, ZG00]. However, almost all of the existing P2PIR systems only focus on finding relevant documents for given queries, while the trustworthiness of documents and document providers has been ignored. Therefore, the present thesis is motivated to design and develop *trust-aware Information Retrieval systems in P2P networks* (trust-aware P2PIR systems) for both cooperative and uncooperative scenarios because of the need to retrieve not only *relevant* but also *trustworthy* documents on request.

The major purpose of Information Retrieval is to retrieve a number of relevant

documents for a user's information needs. Then, the user selects a set of these as the final answers. The selected documents can be viewed as useful resources for the user's specific request. *Trust* in the proposed trust-aware P2PIR systems denotes the degree of assessment of the reliability and quality of the relevant document to meet the user's specific request. In this case, reliability means that there is no virus in the retrieved document, quality means that the retrieved documents are the same as advertised in a P2P network, and relevance means how well the retrieved document meets the user's information need. Reliability and quality can be identified by the user's review (i.e., feedback) after using the documents.

1.2.2 Challenges

Building trust-aware P2PIR systems for cooperative and uncooperative Information Retrieval in P2P networks involves the following challenges:

- One of the most important characteristics of P2P networks is their dynamic nature, whereby peers highly frequently join and leave the network [TD07, SGG02]. The structure of a P2P network is always changing, which affects the content distribution in the network [TD07]. Most existing algorithms for computing relevance-based scores between documents or document providers and queries rely on global term statistics (e.g., inverse document frequency, IDF [SWY75]), which are either assumed to be available in advance, or generated by sampling documents, using a reference corpus, or setting an arbitrarily big value. All of these methods are not likely to be useful in real P2P networks, which are highly dynamic and distributed. How to design an approach to estimate global term statistics, which can be integrated with existing approaches of document ranking and document provider (i.e., peer) selection to facilitate effective and practical Information Retrieval in real P2P environments is a challenging problem.

- A wealth of trust management systems has been developed to establish the trust values of peers and files in P2P networks. Following a distinction between entity trust and content trust, used in [GA07] (albeit in a different context), the trust values of peers and files are referred to as *entity trust* in this dissertation. For example, a system may assign the entity trust value of 0.7 to a certain document in the network. This is insufficient in some

application areas which require to select final answers from a large number of resources [GA07], because a number of useful and useless resources could be assigned the same entity trust values. P2PIR is such an environment, the primary goal of which is to select relevant information from a ranked resulting list to satisfy users' information needs. Without taking into account queries and document contents, many relevant and irrelevant documents may have the same entity trust values. It is hard for users to select "useful resources" (i.e., relevant documents in P2PIR) with the same entity trust values. Consider another case, with two queries q_1 and q_2; the relevance scores of q_1 and q_2 for the Document A are 0.99 and 0.01, respectively. Although, the entity trust value of Document A is 0.7 regardless of different queries, in P2PIR systems, the Document A may be much more useful for q_1 than for q_2, since users are interested in relevant rather than irrelevant information. Therefore, the existing entity trust is not enough for information retrieval in P2P networks, and a new trust mechanism is needed. *Content trust* can be proposed to address this problem in this dissertation, which is a trust judgement on the contents of a document, or document provider, for a given query. Essentially, this builds upon both entity trust and relevance. Considering the previous example, it would be better for a content trust model to assign the trust value of 0.8 to the Document A for q_1 and 0.2 for q_2. The challenge is how to identify the content trust factors to evaluate the trustworthiness of a document provided by a peer for a given query, and the trustworthiness of a document provider for a given query. Further, how to combine the content trust factors into coherent schemes to compute the content trust values of documents and document providers.

- In the envisaged trust-aware P2PIR of uncooperative environments, the query process is that, when a user submits a query to the network, a set of peers is selected, which may contain relevant and trustworthy documents for a given query. The query is then forwarded to the selected peers. In response, each of the selected peers process the query and generate a ranked list. Afterwards, the different ranked lists from various peers are merged, re-ranked and then presented to the user. The question is how to select relevant and trustworthy peers to search, and what number of documents should be retrieved from these selected peers for a given query. Moreover, how to merge and re-rank the documents returned from these selected

peers, because the document scores computed by each selected peer are not directly comparable.

- The proposed trust-aware P2PIR systems combine both *relevance* and *trustworthiness* to retrieve documents upon request. The objectives of the systems are: *(i)* to retrieve more relevant documents and *(ii)* to prevent more untrustworthy documents in the results set. However, these objectives are often in conflict. For a given query, some documents may be relevant, but untrustworthy, and some may be trustworthy, but irrelevant. When giving a bigger weight to relevance, more untrustworthy documents may be obtained. Conversely, weighting trustworthiness more, more irrelevant documents may be involved. The trust-aware P2PIR systems either lose retrieval accuracy or the effectiveness of trust to prevent untrustworthy documents, in return for meeting the other objective. Therefore, a compromising weight to strike a balance between relevance and trustworthiness is needed. The question is how to find the compromising weights between relevance and trustworthiness.

- Since P2P networks lack any centralised infrastructure, applications in P2P networks should organise distributed peers into autonomous and collaborative manners to exchange information and accomplish tasks. The challenge is how to design and develop new implementation strategies for the proposed trust-aware P2PIR systems which can organise peers in a collaborative manner, so the factors to compute relevance and trustworthiness of documents, and document providers, for given queries can be collected and computed by any user in a P2P network.

- Currently, there are no standard metrics and testbeds for evaluating the performance of trust-aware P2PIR systems. Therefore, the question of how to develop the evaluation methodologies, metrics and experimental data to evaluate trust-aware P2PIR systems should be addressed.

1.3 Aims and Contributions

The aim of the present work is to design trust-aware P2PIR systems, which can be used to retrieve relevant and trustworthy documents for a given query in both cooperative and uncooperative P2PIR scenarios. The challenges listed in the

previous section and contributions are one-to-one mappings. In this context, the dissertation can contribute the following:

- An effective and practical method to estimate global term statistics based on the characteristics of structured P2P networks (e.g., routing table size). The estimated global term statistics can be integrated within existing algorithms to compute relevance-based document scores for a given query in cooperative P2PIR, and within a heuristic-based peer selection algorithm to compute relevance-based peer (i.e., document provider) scores in uncooperative P2PIR.

- A set of content trust factors has been identified, which is related to the evaluation of trustworthiness for a document provided by a peer for a given query, and the evaluation of trustworthiness of a document provider (i.e., peer) for that query. Moreover, a document trust metric combining these factors is proposed for calculating the trustworthiness of documents provided by a peer for a given query, which extends the peer trust model in PeerTrust [XL04].

- In uncooperative P2PIR, a theoretical-based peer selection model is proposed which can compute clear cut-off values for which peers to search and the number of documents to be retrieved from these selected peers. Moreover, a heuristic-based estimation function for result merging is designed by modifying the INQUERY result merging function [CCB95], which is able to calculate the merged document score by combining the document score provided by a peer and that peer's score for a given query.

- A method is developed to find a set of compromising weights between relevance and trustworthiness in the proposed trust-aware P2PIR system. Afterwards, a ranking approach is designed to sort different compromising weights by the ratios of changing the percentages between relevance and trustworthiness. Finally, preferred solutions can be selected in various situations.

- To implement the proposed trust-aware P2PIR systems in cooperative (or uncooperative) P2PIR and P2P networks, system architectures and data management protocols are designed. These are extensions of the PeerTrust

architecture [XL04] and data management protocols of structured P2P networks in the context of trust-aware Information Retrieval in P2P networks.

- A number of experiments for trust-aware P2PIR systems have been performed on both retrieval accuracy and the effectiveness of trust. In these, several common approaches have been compared, such as existing ranking algorithms in P2PIR (e.g., K-L [XC99]) and trust models in P2P (e.g., PeerTrust [XL04]). Furthermore, a rank-based evaluation metric is proposed to assess the effectiveness of the proposed trust-aware P2PIR system to protect untrustworthy documents in the top-ranked resulting list.

1.4 Thesis Structure

The central focus of the proposed work is trust-aware Information Retrieval in P2P networks. Accordingly, the remainder of the present dissertation is organised as follows:

- **Chapter 2: Literature Survey** provides a comprehensive review of the literature related to trust-aware Information Retrieval in P2P networks, including issues of P2P network architectures, Information Retrieval in P2P networks and trust management systems. Firstly, an overview of different P2P network architectures is described. Then, the existing techniques of two P2PIR scenarios (cooperative and uncooperative) are surveyed individually. Along the way, the related work of trust management systems in P2P networks and Information Retrieval are described. A discussion concludes this chapter.

- **Chapter 3: Trust-Aware P2PIR System in Cooperative P2PIR Environments** represents a trust-aware P2PIR system in cooperative environments, which includes trust-based document description, document ranking, and the content trust metrics designed to evaluate a document provided by a peer for a given query. Moreover, implementation issues of the proposed trust-aware P2PIR system in cooperative P2PIR are discussed, including a system architecture and data management protocols. The effectiveness of trust-aware P2PIR system in protecting untrustworthy documents, retrieval accuracy and scalability, are evaluated in this chapter.

- **Chapter 4: Trust-Aware P2PIR System in Uncooperative P2PIR Environments** describes a trust-aware P2PIR system in uncooperative environments, including trust-based peer description, peer selection and result merging. Furthermore, a system architecture and data management protocols are designed to implement the proposed trust-aware P2PIR system into uncooperative P2PIR and structured P2P networks. The effectiveness of the trust-aware P2PIR system in protecting untrustworthy documents, retrieval accuracy and scalability, are evaluated in this chapter.

- **Chapter 5: A Theoretical-Based Peer Selection Approach in Uncooperative P2PIR Environments**. Peer selection is an important problem for P2PIR in uncooperative environments. In contrast to the heuristic-based peer selection approach in Chapter 4, this chapter proposes a theoretical-based model for optimal peer selection which is inspired by the decision-theoretical approach [Fuh99]. The proposed precision-risk $PrRi$ model can estimate the precision and risk value of the results set when users specify the number of documents to be retrieved. A clear cut-off can be computed for the number of peers to search, and the number of documents to be retrieved from each of the selected peers.

- **Chapter 6: An Analysis of the Trade-off Study between Relevance and Trustworthiness** describes an approach to find a set of near optimal solutions of the relative weights between relevance and trustworthiness in the proposed trust-aware P2PIR systems. Then, the near optimal solutions are visualised to represent a trade-off surface. A ranking approach is developed to sort the different near optimal solutions by the ratios of changing percentages in relevance related to the changing percentage of trustworthiness. This may help users to select a preferred solution as the final answer.

- **Chapter 7: Conclusions and Further Work** reviews the thesis contents, contributions, and potential future work is discussed.

Chapter 2

Literature Survey

Trust-aware Information Retrieval in P2P networks is motivated by the need to retrieve not only relevant but also trustworthy (e.g., reliable and high quality) documents for a given query. Trust-aware P2PIR systems are built upon three key elements: *(i) network architecture*; *(ii) relevance-based (or similarity-based) Information Retrieval in P2P*; and *(iii) trust management*. A network architecture defines the functionality and responsibility of each peer, as well as data-location schemes and message-routing mechanisms. The function of relevance-based Information Retrieval systems in P2P is to locate documents which are likely to be relevant for a given query in P2P networks, and the function of trust management is to collect and analyse evidence from a network to make an assessment of the trustworthiness of a file or peer.

This chapter seeks to provide a comprehensive review of the literature related to each of the above three subjects. It begins with general background information of P2P network architectures in Section 2.1. The section provides an overview of different P2P network architectures by issues of data location schemes and search mechanisms. Section 2.2 discusses the advantages and disadvantages of P2P networks for Information Retrieval, and then describes the existing techniques of Information Retrieval in P2P networks (P2PIR), with a focus on relevance. Two major scenarios of P2PIR are surveyed individually, namely, *cooperative P2PIR* and *uncooperative P2PIR*. Section 2.3 introduces the definition and categories of trust. Along the way, the related work to trust management systems in P2P networks and Information Retrieval is surveyed. The limitations of the existing P2PIR systems and trust management systems are discussed in Section 2.4, and this chapter is summarised in Section 2.5.

2.1 Peer-to-Peer Network Architectures

P2P network architectures determine the required functionality and responsibility of each peer, as well as a data-location schema and a message-routing mechanism. Design and implementation strategies of applications in P2P networks extensively rely on network architectures. In other words, a certain application which is employed in different P2P networks should have different design and implementation strategies. For example, in a hierarchical P2P network, only super peers take the responsibility to forward queries to other peers, which does not happen in other P2P network architectures. Design and implementation of trust-aware P2PIR systems should be determined by the selected P2P network architecture. In this section, different P2P network architectures are introduced and one P2P network architecture is chosen for the design of our proposed trust-aware P2PIR systems.

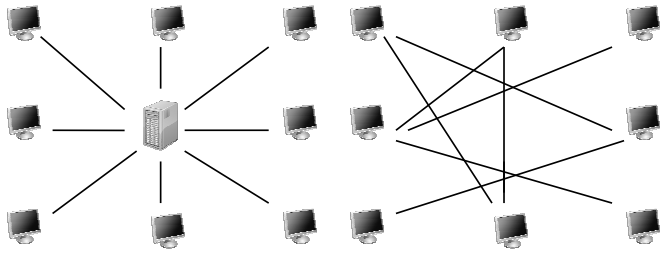
Nowadays, a large number of P2P networks have been designed, developed and widely used. P2P networks are *"distributed systems in nature, without any hierarchical organisations or centralised control. Peers form self-organising overlay networks that are overlayed on the Internet Protocol networks, offering a mix of various feature such as robust, wide-area routing architecture, efficient search of data items, selection of nearby peers, redundant storage, permanence, hierarchical naming, trust and authentication, anonymity, massive scalability and fault tolerance* [LCP+05]". The architectures of P2P networks determine the data location schemes and message routing mechanisms to be supported. A wealth of surveys of P2P networks [GMH04, ATS04, LCP+05, CC05] studied different aspects, such as overlay network schemes and search mechanisms. Among these works, the existing P2P network architectures can be typically split into two categories, namely, *unstructured* and *structured*. An unstructured P2P network is a randomly-established network overlay. On the contrary, a structured P2P network is established with pre-defined rules and data management protocols. This categorisation is adopted by this thesis, since it best describes current P2P networks and has been widely used. In the remainder of this section, P2P networks are described by issues of data-location schemes and query-routing mechanisms.

2.1.1 Unstructured Peer-to-Peer Networks

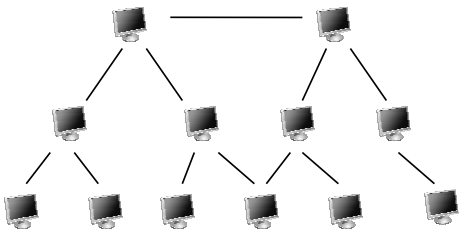
An unstructured P2P network is a randomly-established network overlay, in which new peers join the network without any pre-defined rules [LCP+05]. In general,

there are three types of unstructured P2P network architectures available, namely, *brokered P2P networks, completely decentralised P2P networks* and *hierarchical P2P networks*.

In a brokered P2P network [nap], such as the one shown in Figure 2.1 (a), a set of peers with strong computation capabilities and storage is selected to serve as a single centralised directory service. The remaining peers serve as information providers and users simultaneously. Data (e.g., music, files or documents) is actually stored locally, and the descriptions of data (e.g., file name) are sent to the centralised directory service. To conduct a search, users issue queries to the centralised directory service to retrieve peers which contain the required data for given queries. When users have obtained the contact information of peers (e.g., IP address and port) from the centralised directory service, they will communicate with the peers directly to download the data.



(a) a brokered P2P network (b) a completely decentralised P2P network



(c) a hierarchical P2P network

Figure 2.1: Three different types of unstructured P2P networks

In a completely decentralised P2P network [gnu], such as the one shown in Figure 2.1 (b), all the peers in the network simultaneously act as information providers, users and directory services. A peer stores its data locally and send descriptions of it to all of its neighbouring peers. When locating the required data, a user sends a query to all of its neighbouring peers. When all of the neighbouring peers have received the query, they process it and forward it to all

of their neighbouring peers until the required data has been found or the pre-defined TTL (Time-to-Live) [1] is reached. This search mechanism is referred to as *flooding*. Alternatively, the query may be forwarded to a random neighbour instead of all of the neighbours to save network costs, and this is called a *random walk*.

In a hierarchical P2P network [GEBR+03, HZM+08, Lua05], such as the one shown in Figure 2.1 (c), networks consist of super peers and leaf peers. Peers are organised into a hierarchy, including a super-peer level (or called a hub peer level in some papers, for example, in [LC07a]) and a leaf peer level. Super peers are directory services which store data descriptions of their leaf peers and neighbouring super peers. In a hierarchical P2P network, super peers need to cooperatively process the searching process. For example, to conduct a search, a user sends a query to a super peer it belongs to and the super peer processes and forwards the query to its leaf peers and neighbouring super peers until the required data has been found or the predefined TTL is reached.

While the flooding and a random walk search mechanisms are effective for locating highly replicated items, they are poorly suited for locating rare items. Moreover, unstructured P2P networks do not scale well when handling a high rate of aggregate queries and a sudden increase in system size [ATS04, LCP+05].

2.1.2 Structured Peer-to-Peer Networks

To address the limitations of unstructured P2P networks, a new generation P2P network has been developed, which is a structured P2P network, such as the one shown in Figure 2.2. A structured P2P network is an established network overlay with pre-defined rules, in which new peers join the networks, not randomly, but in specified locations according to various structured P2P network data management protocols [LCP+05]. Structured P2P networks employ distributed hash tables (DHTs) to store and retrieve data. Differently structured P2P networks have specific key spaces and message routing mechanisms. In structured P2P networks, all the peers serve as information providers, users and directory services at one time, which means that there is no peer with a special or administrative role. Each peer provides equal functionalities as the peers in completely decentralised unstructured P2P networks. Currently, a large number of structured P2P networks have been developed [SMK+01, RFH+01, RD01, MM02, ACMD+03, MBR03,

[1] TTL is a limit on the number of steps for a query in the network before it expires [LCC+02].

ZKJ01] and applied for a widely variety of Internet-scale applications, for example, eMule [eMu], aMule [aMu] and MLDonkey [mld]. Structured P2P networks are regarded as being a significant improvement over unstructured P2P networks in terms of scalability, efficiency and reliability [RV03, LLH+03, LCP+05]. In this dissertation, structured P2P networks are chosen as a basic P2P network architecture to design our proposed trust-aware P2PIR systems.

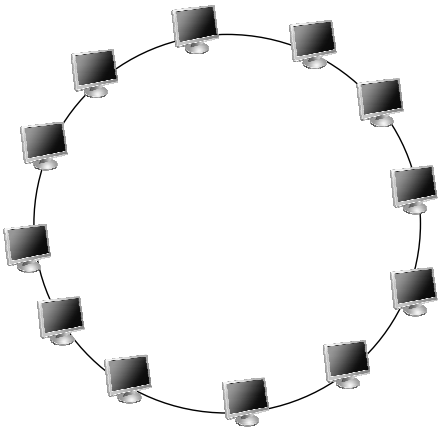


Figure 2.2: System architecture of structured P2P networks

2.2 Information Retrieval in Peer-to-Peer Networks

The primary goal of trust-aware P2PIR is to retrieve not only trustworthy but also relevant documents upon request. One of the key purposes of trust-aware P2PIR is to locate relevant information for given queries. This, referred to as Information retrieval (IR) in P2P environments (P2PIR), has been an active field of research in the last decade, and a number of papers have been published so far.

The reason for employing P2P as the basic infrastructure for IR is that P2P networks may be an attractive alternative to centralised search engines (e.g., Google) for both technical and economic reasons [ZS06]. Similar to the problems studied in traditional IR environments [BYRN99, goo, Cal00], two major scenarios have been extensively explored in P2PIR, namely, *cooperative P2PIR* and

uncooperative P2PIR. These two scenarios are defined by the degrees of cooperation provided by document providers (i.e., peers). For example, each document provider in the network can cooperate closely to provide its document copies or content statistics for document retrieval and ranking, and this can be defined as *cooperative P2PIR* such as P2P Web Search [LLH+03, SYYW03, KWTA07, LJT07, JNC06]. This problem normally corresponds to the Web search engine case in traditional IR environments such as Google. In summarising the state-of-art cooperative P2PIR [CJL+09, NYF08a, XSD+08, MM09, PLPZR08, RGKK09, NYF08b, ZRL+07, KWTA07, LJT07, JNC06, YDRC06, LKZ+06, JYF07, Che05, LLQ+04, TDX04, KJ03, TD04, LC04b, TXD03, TXM03, RV03, LLH+03, CAN02, KJ04, SLZ+07, SLZ+09, GKN09, YMA09, MMKA04, MKA06, SYYW03, PMB06], three characteristics can be extracted and summarised: *(i)* each document provider needs to publish full document copies or document descriptions (e.g., statistics such as term frequency) to the network; *(ii)* document copies or descriptions can be obtained, assessed and downloaded by any user without any limitations upon request; and *(iii)* users should employ the same search engine to retrieve the required documents in a network.

In another case, document providers may not be willing to provide their individual document copies or descriptions to the network due to copyright issues and access limitations. Moreover, each of these document providers may employ an individual search engine for document indexing and retrieving, which is often the case in digital libraries. In such an environment, it may not be possible to know which document provider is using which type of search engine. In addition, it is unlikely to expect document providers to inform other users about detailed techniques of search engines, such as ranking algorithms and stemming algorithms. This scenario is defined as *uncooperative Information Retrieval in P2P networks*; a typical example is such as a P2P Digital Library search [ZCLL04, LC06, LC05, NF07a, NFTN05]. This problem normally corresponds to Distributed Information Retrieval in traditional IR environments such as searching through digital libraries. In summarising the literature of uncooperative P2PIR [LC04a, ZCLL04, LC06, LC05, WB05, LC07a, ZTW07, Pap08, ZL06, RP08, RPTW08, ZHTW08, CSB+05, Wit08a, ZYKG05, ZYKG07, WM09, NF07a, NFTN05, DMP+09, RP08], three characteristics can be concluded: *(i)* documents cannot be directly reviewed or downloaded by users due to issues of access limitation, payment or copyright; *(ii)* document providers may or may not

publish their descriptions (e.g., statistics of peers) to the network; *(iii)* each document provider employs an individual search engine with unknown techniques.

Both cooperative P2PIR and uncooperative P2PIR offer different possibilities and difficulties for a distributed search solution in P2P networks. The advantages and disadvantages of the P2P paradigm for Information Retrieval, compared to the Client-Server paradigm are discussed in Section 2.2.1. Then, prior work related to cooperative and uncooperative P2PIR is surveyed in Sections 2.2.2 and 2.2.3, respectively.

2.2.1 Peer-to-Peer vs. Client-Server for Information Retrieval

Nowadays, there are two major types of network paradigms for information sharing, namely, the *Client-Server* (CS) model and the *Peer-to-Peer* (P2P) model. In the CS paradigm, all users communicate with a single, centralised server. Sometimes the centralised server can be replaced by a set of hierarchical servers to improve scalability. The servers of one level act as clients to the higher level servers in a hierarchical model, which is a tree structure. In the CS model, all documents are stored in one central server, and users can retrieve the required documents from that server. The advantages of the CS paradigm for Information Retrieval are as follows [Mal03]:

- Document management is much easier, because all the documents are stored in one location.
- Document retrieval is efficient.
- Searching is comprehensive.

The disadvantages of the CS paradigm are as follows:

- The central server can easily become a bottleneck during periods of high demand.
- It has a single point of failure.
- Maintenance costs are high.

- It exhibits poor scalability, because all the actions rely on the central server or a fixed number of servers. When new documents are published to the network, requests (e.g., storage, processing power) to the server increase, and the server performance is diminished.

- It is expensive to frequently update the global index in the central server.

The Peer-to-Peer paradigm may offer an attractive alternative to the traditional Client-Server paradigm for Information Retrieval. A P2P network is a distributed network composed of a large number of distributed, heterogeneous, autonomous and highly dynamic peers, in which participants share parts of their own resources such as processing power, storage capability and software [ATS04, LCP+05]. The advantages of P2P architecture for Information Retrieval are as follows [CAN02, Mal03]:

- Since any peer in a P2P network can be assumed to be an information provider and directory service at the same time, P2P networks provide more opportunities for replication and minimise the chance of having a single point of failure. This indicates that, if one peer breaks down, the rest of peers in the network are still able to work.

- P2P networks are composed of a large number of distributed computers sharing part of their own resources (e.g., bandwidth, storage) [LCP+05]. This makes it easier for P2P networks to avoid bottlenecks, such as traffic overload, because they can distribute data and balance requests across the network without using a central server.

- P2P networks can scale better than centralised networks because P2P networks take advantage of the unused resources of each peer, such as processing power and storage. As peers arrive and demand on the system increases, the total capacity of the system also increases [LCP+05]. This flexibility is not true of centralised networks.

- Information in the network can be effectively and efficiently updated because data is stored, highly distributed and can be maintained locally.

- P2P networks are appealing from an economic perspective since they do not need to build expensive infrastructures or bear high maintenance costs to manage information.

On the other hand, P2P architectures have a few of disadvantages for Information Retrieval:

- Document retrieval is not efficient, because multiple distributed peers have to work collaboratively to accomplish the retrieval task.
- High network traffic on messages exchanging between peers.
- No guarantee of the reliability and quality of documents, because P2P networks depend on the voluntary participation of peers to contribute and maintain their documents in a distributed way.
- Searching may not be comprehensive because only part of a network may be retrieved.

Both types of network paradigms offer a number of advantages and disadvantages, which is why they exist simultaneously. When considering better scalability, low maintenance cost, no single point of failure and bottlenecks, P2P network architectures may provide a better solution for Information Retrieval than the CS paradigm.

2.2.2 Cooperative Information Retrieval in Peer-to-Peer Networks

Extensive research has been undertaken into cooperative P2PIR in the last few years. Prior work on cooperative P2PIR is surveyed in this subsection, and four sub-problems of cooperative P2PIR are also analysed. Before discussing cooperative P2PIR, the following assumptions must be made: *(i)* peers should provide any required information for document retrieval and ranking, such as document statistics; *(ii)* any peer in the network needs to employ an integrated public search engine which can run unstructured text queries and return a list of documents. In current cooperative P2PIR systems, four sub-problems need to be addressed:

- *Document Selection Criteria.* Document selection criteria typically refer to users' requirements for document retrieval. Almost all of the cooperative P2PIR systems only take *retrieval quality* (e.g., relevance for a given query) as the selection criterion.

- *Document Description.* The form of representing a document provided by a peer in a network, such as the frequency of a term in the document.
- *Document Retrieval and Ranking.* How to retrieve documents from a network which can satisfy the selection criterion. A ranking algorithm will establish an ordering of the retrieved documents by how well each document matches or satisfies a given query.
- *Implementation.* A mechanism for organising peers into a cooperative manner so that document copies and descriptions can be collected by each peer in a network.

Retrieval quality is the most common selection criterion in all of the IR areas, which considers the relevance degrees between documents and given queries. Since all of the cooperative P2PIR systems choose relevance as the document selection criterion, the remainder three sub-problems in cooperative P2PIR will be discussed, including issues of document description, document retrieval and ranking, and implementation.

2.2.2.1 Document Description

To decide which documents are likely to satisfy users' queries, document retrieval and ranking methods require information of documents. This information is called document description (or document representation is some literature), which is assumed to be available in cooperative P2PIR. In distributed networks, document description normally consists of three problems [Cal00, CC01]: *(i)* how to represent documents; *(ii)* how to compress the size of document descriptions; and *(iii)* how to obtain document descriptions in a distributed way.

The first problem is how to represent a document. Currently, four approaches have been developed to represent documents in P2P networks, which are: *(i) full-text based description; (ii) semantic-based description; (iii) query-driven based description*; and *(iv) link-based description.* Typical full-text based descriptions include a list of terms with corresponding term frequencies in a document, as well as the total number of terms and the length of the document. Full-text based document description is the most widely used approach, not only for cooperative P2PIR but also all of the IR area. In cooperative P2PIR, a large number of approaches [CJL+09, NYF08a, XSD+08, MM09, PLPZR08, RGKK09, NYF08b,

ZRL+07, KWTA07, LJT07, JNC06, YDRC06, LKZ+06, JYF07, Che05, LLQ+04, TDX04, KJ03, TD04, LC04b, TXD03, TXM03, RV03, LLH+03, CAN02, KJ04, SLZ+07, SLZ+09] employ full-text based descriptions for document retrieval and ranking. Compared to other document description methods, full-text based descriptions provide much more comprehensive descriptions for text documents. However, the size of a full-text based description is significantly large.

A semantic-based document description represents a document by a propositional Semantic Web language, such as Description Logic [BCM+03]. Fausto *et al.* [GKN09] analysed the meanings of words and phrases in a document, and then represented a document by a list of terms and phrases with an enumerated sequence of conjunction components $\sqcap A^d$ or the disjunction symbol $\sqcup$. These symbols are defined in Description Logic. Yu *et al.* [YMA09] developed a Web content capability description (WCD) language to represent documents, which is a metadata of Web contents and the composition of knowledge domains. The knowledge domain ontologies are described by OWL [Hor05] and the metadata is represented in RDF [rdf]. The advantage of a semantic-based document description is that it can address the problem of which term is semantically independent in full-text based descriptions. However, generating a semantic-based document description requires analysing every single word and phrase in a document, which can be time consuming and computationally expensive. Since efficiency is an important issue in IR [BYRN99], semantic-based document descriptions are not widely used in cooperative P2PIR.

A query-driven based document description stores information about queries satisfied in the past. Mine *et al.* [MMKA04, MKA06] represented a document by a content file which consists of *Content* and two histories in terms of *Q-RDH* and *Q-SAH*. Content is a metadata-based description, including the title of the document, the abstract of the document, the address of the document provider, and the range across which it is allowed to be distributed (e.g., ALL, Community, Agent). Q/RDH is a list of pairs of a query and the address of the document provider which provided the relevant documents for that query in the past. Q/SAH is a list of pairs of a query and the address of a document requester. Unlike full-text based document descriptions, the size of a query-driven based document description is relatively small, which is effective for transmission through P2P networks. The major problem of applying query-driven based document descriptions is that the relevant judgement of a document for a given query should require manual

effort [CC01]. This cannot be consistent because different people may have various standards of judgement, even for the same query and document. Further, it is expensive and difficult to apply query-driven based document descriptions when document providers frequently update their documents.

In terms of link-based document descriptions, a document is represented by a number of different types of links. For example, Shi *et al.* and Parreira *et al.* [SYYW03, PMB06] represent Web documents (e.g., Web pages) by various links such as *inner link, virtual link, afferent link* and *efferent link*. Similar to query-driven based document descriptions, the size of a link-based document description is relatively small, which is effective for transfer over a network. However, the range of applications for link-based document descriptions is limited because it is only useful when the link information is available, as in the case of Web pages.

The most commonly used document description so far is full-text based document descriptions. In unstructured P2P networks, a peer will send full copies of documents or document statistics to its neighbouring peers or super peers, while in structured P2P networks, document statistics are routed to specific peers by data location policies and DHTs. The main problem of full-text based descriptions is that the full copies of documents or statistics are significantly large in size. When publishing or retrieving documents, the copies or statistics have to be transferred over the network, which may produce a large amount of network traffic. This cannot be scalable [LLH+03].

The second problem is how to compress the size of a document description. Zhang *et al.* and Yang *et al.* [ZS05, YDRC06] measured the bandwidth cost and search latency of full-text based document descriptions of publishing and retrieving documents in both unstructured and structured P2P networks. Their results show that a full-text based document description is bandwidth consumption, ineffective and unscalable for both types of P2P networks. Different methodologies of compressing the size of full-text based document descriptions have been explored. To address this problem, five approaches have been studied so far, including *bloom filter, top-ranked term selection, feature extraction, query history* and *stemming*. Some approaches employ one method only such as [LLH+03, CJL+09, RV03], while some combine several methods together such as [RGKK09]. These methods will be described in the following paragraphs.

A Bloom filter [Blo70] is a hash-based data structure which can represent a set

compactly. Li *et al.* [LLH+03], Chen *et al.* [CJL+09], and Reynolds *et al.* [RV03] reduced the size of a full-text based document description by using a Bloom filter.

Top-ranked term selection means choosing some top-ranked terms in a document to represent the document contents. Not all of the terms in a document are important for representing the document contents. An approach has been developed to remove the low-weighted terms in a document by ranking all of the terms. Only the top-ranked terms are selected and published to the network instead of the whole number of terms listed in a document. Tang *et al.* [TD04] adopted Okapi [RWHB95] to calculate the term weights, and subsequently, they [TDX04] applied a Random Projection [FM87] to filter the low-weighted terms in a different paper. Rosenfeld *et al.* [RGKK09] computed a threshold of the number of maximum important terms in a document, and declare that the threshold can be adapted according to the system overload. ALVIS [SLZ+07, SLZ+09, ZRL+07, PLPZR08, LKZ+06] developed a highly-discriminative key (HDK) to publish the selected terms or term sets with the corresponding DF_{max} documents, where DF is the document frequency and DF_{max} is a threshold defined by the system.

Feature extraction is an approach to represent documents by a number of features and corresponding values. In Chen's approach [Che05], terms appearing in the different frames usually carry different weights. For example, terms appearing in the title are assigned more weights than the abstract. Skobeltsyn *et al.* [SLZ+07, SLZ+09] took into account the popularity of terms and term combinations from the user query history to remove unpopular terms. This approach keeps retrieval quality at an acceptable level.

Stemming is a method to simplify a term in its root form and it has been extensively studied in traditional IR. Rosenfeld *et al.* [RGKK09] employed stemming algorithms to reduce the size of a full-text based document description. Although compressing the size of a full-text base document description can reduce the cost of transferring descriptions among peers, this method scarifies the retrieval quality in terms of precision and recall [NYF08a, NYF08b].

The last problem is how to obtain document descriptions in P2P networks, which is one of the most challenging problems in P2PIR. In full-text based P2PIR, document descriptions include two types of statistics, namely, *global term statistics* and *local term statistics*. Local term statistics can be easily obtained from peers in the network. Since P2P networks are highly distributed and there is no

central server, peers lack a global view of term statistics for underlying documents in a network. It is challenging to collect global term statistics in a P2P network, but without them, it is difficult to measure the importance of a term in the network. In addition, peers in a P2P network are highly dynamic, and frequently join and leave the network [TD07, SGG02]. The structure of a P2P network is always changing, and this affects the content distribution in the network, as well as the global term statistics [TD07]. This makes it hard to keep global term statistics up-to-date. Some cooperative P2PIR systems assume that global term statistics are available in advance [LLH+03, Che05, RV03, LLQ+04, KJ04]. Alternatively, Cuenca-Acuna *et al.* [CAN02] assumed that each peer can act as a central server to collect and maintain global term statistics about other peers in a network. Both of these assumptions are not realistic, and in order to address this problem, several strategies have been explored to obtain global term statistics for P2PIR. A few approaches [XSD+08, KWTA07, LJT07] just set some global values (e.g., the number of terms in a network) to be a maximum unsigned integer. Moreover, Tang *et al.* [TDX04, TD04, TXM03], Klampanos *et al.* [KJ04] and Klemm *et al.* [KA05] sample documents from a part of the network and use merged statistics to replace global term statistics. Then, these estimated global term statistics are disseminated to the network and updated periodically [PLPZR08, TDX04, TD04, KJ04, TXM03, KA05]. The limitations of these approaches will be discussed in Section 2.4.

2.2.2.2 Document Retrieval and Ranking

Document retrieval means to locate documents which may satisfy the selection criteria. Since in cooperative P2PIR the retrieval quality is the only selection criterion, document retrieval means finding documents containing relevant contents for a given query. In traditional IR, document retrieval relies on a centralised server, in which document providers publish their document copies or statistics to a central server and users send their queries to the central server to retrieve the relevant documents. Contrarily, information in P2P networks is highly distributed and there is no central server to manage the information of a network. Two types of approaches have been developed for document publication and retrieval in structured and unstructured P2P networks, respectively.

Currently, a wealth of cooperative IR systems have been developed in structured P2P networks [PMB06, SYYW03, SLZ+07, PLPZR08, ZRL+07, LLH+03,

LKZ+06, GKN09, TXM03, TXD03, LC04b, KJ03, JYF07, LJT07, KWTA07, XSD+08, CJL+09, RV03, NYF08a, NYF08b, Che05, TDX04, TD04, SLZ+09]. For document retrieval and publication in structured P2P networks, DHT can provide a solution. In structured P2P networks, DHTs provide an interface to put and get pairs of *key/value*. When publishing documents to a network, DHTs store an inverted index. The *keys* inserted into DHTs are the terms, and the *values* in DHTs are the corresponding posting lists including term frequency, document name and document location. The *key/value* pairs are stored in distributed peers over a network based on the data location policies of the structured P2P networks. To retrieve documents containing query terms in structured P2P networks, systems perform lookup messages for each query term to retrieve its posting list, and then intersect the posting lists to obtain the relevant documents containing the query terms.

A number of cooperative P2PIR systems employ unstructured P2P networks to publish and retrieve documents [KJ04, LLQ+04, JNC06, CAN02, MKA06, MMKA04, CJL+09]. In unstructured P2P networks, peers either publish/retrieve document copies or statistics to/from super peers or their neighbouring peers. The process of document publication and retrieval is the same as the process of data publication and retrieval in unstructured P2P networks (see Section 2.1.1), in which flooding and a random walk are employed. Rosenfeld *et al.* [RGKK09] proposed using a hybrid approach which employs DHTs to publish and locate infrequent terms, and employs a random walk or flooding to publish and find popular terms.

Both document publication and retrieval methods in structured and unstructured P2P networks suffer several drawbacks for such as high network transmission cost and poor retrieval quality. For example, in structured P2P networks, the main problem is that the traffic cost of publishing and retrieving posting lists is extremely high and inefficient, which has been studied by Li *et al.* [LLH+03] and Yong *et al.* [YDRC06]. In unstructured P2P networks, flooding will cost high network bandwidth and it is poor scalability. Moreover, in hierarchical unstructured P2P networks, many super peers can be heavily overloaded [YDRC06]. A random walk is able to provide better scalability than flooding [ALPH01, LCC+02, GMS06] for document publication and retrieval in unstructured P2P networks. However, the quality of retrieval results is not guaranteed. The searching process can be stopped when reaching TTL but without

finding any relevant documents for a given query.

To address these problems, several optimal approaches have been designed for efficient searching and improvement of retrieval quality. For example, *content-based network overlay, query expansion, query history* and *document replication policy*. Content-based network overlay (also called semantic overlay network (*SON*)) is a logical network overlay where document providers with similar contents are grouped closer around their semantics. Documents are analysed by clustering algorithms to extract semantics. Li *et al.* [LLH+03] applied Probabilistic Latent Semantic Analysis (PLSA) [Hof99] to cluster documents in 100 groups. Nguyen *et al.* [NYF08a, NYF08b] developed a greedy algorithm to group documents based on term distributions derived from user queries and document contents. Tang *et al.* [TDX04] employed a hierarchical version of spherical k-means [DM01] to cluster documents. In this approach, a cluster of similar documents with a single vector represents the centroids of this cluster and uses the centroids of the clusters as representations of the documents. In content-based network overlay, to conduct a search, the system analyses the query and then forwards it to the areas with similar contents which could potentially contain the most relevant documents [TXM03, TXD03, LC04b, KJ04, KJ03, NYF08a, NYF08b, Che05, TDX04, JNC06].

Query expansion is a method to reformulate queries to improve retrieval quality. Normally, queries are expanded with additional terms to obtain more relevant documents for a given query. Xu *et al.* [XSD+08] and Tang *et al.* [TD04] analysed the top ranked documents and extract the high weight terms as the query expansion terms for a given query.

Search history is another efficient way to improve retrieval quality. Mine *et al.* [MMKA04, MKA06] and Xu *et al.* [XSD+08] compared similarities between current queries and previous queries to find relevant documents with queries satisfied in the past. In their approaches, relevance can be inferred from feedback derived from document reading times, downloading times and top-ranked documents. Since flooding and random walk are effective and efficient to locate popular documents in unstructured P2P networks, Lv *et al.* [LC04b], Chen *et al.* [CJL+09], Kurasawa *et al.* [KWTA07] and Cuenca-Acuna *et al.* [CAN02] replicate many copies of documents and then disseminate them to the network.

After retrieving relevant documents containing query terms, one central problem regarding information retrieval systems is the issue of predicting which documents are more likely to satisfy users' queries. Such a decision is usually dependent on a ranking algorithm which attempts to establish an ordering of the retrieved documents. Several traditional ranking algorithms have been adopted in cooperative P2PIR. Most of approaches [LKZ+06, LJT07, KJ03, CAN02, LLQ+04] employed the Vector Space Model (VSM) [BYRN99] to rank relevant documents for a given query. Kurasawa *et al.* [KWTA07] used a probabilistic ranking algorithm [BYRN99]. These ranking algorithms require full-text based document descriptions. On the other hand, Li *et al.* [LLH+03], Parreira *et al.* [PMB06] and Shi *et al.* [SYYW03] employed link-based document descriptions and PageRank [BP98] for document ranking. To apply the traditional document ranking algorithms in cooperative P2PIR, the most challenging problem is how to obtain term statistics to calculate relevance-based document scores, which has been described in the document description section.

2.2.2.3 Implementation

P2P networks lack centralised infrastructures, but rather depend on the voluntary participation of peers. One challenge is how to develop a mechanism for organising peers into a cooperative manner so that documents can be retrieved and ranked by any peer in a network. Implementation of P2PIR systems typically involves two issues of decentralised system architectures and data management protocols.

For cooperative P2PIR systems in completely decentralised P2P networks, each peer acts as an equal role, therefore, the decentralised system architecture is designed to fulfil the following functionalities: *(i)* storing document descriptions of neighbouring peers *(in unstructured P2P networks)* or a portion of the global document descriptions *(in structured P2P networks)*; *(ii)* processing and forwarding users' queries; and *(iii)* ranking retrieved documents for a given query [PMB06, SYYW03, SLZ+07, PLPZR08, ZRL+07, LLH+03, LKZ+06, GKN09, TXM03, TXD03, LC04b, KJ03, JYF07, LJT07, KWTA07, XSD+08, CJL+09, RV03, NYF08a, NYF08b, Che05, TDX04, TD04, SLZ+09]. For hierarchical unstructured P2P networks, super peers provide these functionalities instead [KJ04, LLQ+04, JNC06, CAN02, MKA06, MMKA04, CJL+09, RGKK09].

Cooperative P2PIR systems in different P2P networks employ various data

management protocols for document publication, retrieval and update. In structured P2P networks, DHTs and message routing protocols are employed to publish and retrieve documents containing query terms. Most of the approaches [PMB06, SYYW03, SLZ+07, PLPZR08, ZRL+07, LLH+03, LKZ+06, GKN09, TXM03, TXD03, LC04b, KJ03, JYF07, LJT07, KWTA07, XSD+08, CJL+09, RV03, NYF08a, NYF08b, Che05, TDX04, TD04, SLZ+09] modify data management protocols of structured P2P networks by extending the index manager with required information (e.g., term frequency, document length) for document retrieval and ranking. In unstructured P2P networks, the document copies or descriptions are published to either super peers or neighbouring peers. Super peers and neighbouring peers store these information and retrieve relevant documents by random walk or flooding [KJ04, LLQ+04, JNC06, CAN02, MKA06, MMKA04, CJL+09].

2.2.3 Uncooperative Information Retrieval in Peer-to-Peer Networks

A wide variety of research focuses on uncooperative Information Retrieval in P2P networks. Before discussing the relevance of the work on uncooperative IR, the following three assumptions must be made: *(i)* document providers do not provide any descriptions of their documents, instead, they may or may not provide descriptions of themselves; *(ii)* users cannot directly access document copies or statistics due to copyright issues or access limitations; *(iii)* each document provider employs an individual search engine which can run unstructured text queries and provide rankings of retrieved documents with relevance scores. In uncooperative P2PIR, the query process is that when a user submits a query to a network, then a set of peers are selected which may contain relevant information for a given query. The query is then forwarded to the selected peers; in response, each selected peer processes the query and generates a ranked list. Afterwards, the different ranked lists from the selected peers are merged and re-ranked in order to present to the user. According to the search process, uncooperative P2PIR can be viewed as five sub-problems which are described below:

- *Document and Peer Selection Criterion.* Similar to the selection criteria in cooperative P2PIR, selection criteria for peers (i.e., document providers) and documents are related to users' requirements.

- *Peer Description.* Similar to the document descriptions in cooperative

P2PIR but in the peer level, it represents contents or other information (e.g., query history) of peers.

- *Peer Retrieval and Selection.* Given user queries and a set of peer descriptions, this is the problem of how to retrieve and select peers that are most likely to contain documents satisfying the selection criteria.

- *Result Merging.* This is how to merge the results returned from different selected peers and re-rank them as a single list with ordering.

- *Implementation.* Similar to the implementation of cooperative P2PIR, uncooperative P2PIR systems also need a cooperative manner so that peers can be selected and results can be merged by any peer in a network.

Most of the existing work in uncooperative P2PIR systems only considers *retrieval quality* as the selection criterion, which is the same as that of cooperative P2PIR systems. The only exception is that Nottelmann *et al.* [NF07b, NF07a, NFTN05] took cost as the peer selection criterion. Since the selection criterion of the existing uncooperative P2PIR systems are relevance or cost, in what follows, the remainder four sub-problems of uncooperative P2PIR will be explored, including issues on peer description, peer retrieval and selection, result merging and implementation.

2.2.3.1 Peer Description

To decide which peers are likely to contain documents satisfying users' requirements, peer selection approaches require peer descriptions. Peer descriptions (or called peer representations) are variations of document descriptions but at the peer level. For example, in full-text based peer descriptions, term frequency is replaced by document frequency. Same to the document descriptions in cooperative P2PIR, peer descriptions in uncooperative P2PIR also give rise to three problems, which are: *(i)* how to represent a peer; *(ii)* how to obtain peer descriptions from the network; and *(iii)* how to compact the size of a peer description.

Currently, four kinds of peer descriptions have been employed in uncooperative IR, such as *(i) full-text based descriptions; (ii) query-driven based descriptions; (iii) user-interest based descriptions*; and *(iv) cost-based descriptions.* The full-text based peer description is a variant of the full-text based document description, which includes terms with corresponding document frequencies (df) in

a peer, as well as the total number of documents and average of document length. Most of uncooperative P2PIR systems employ full-text based descriptions such as [LC04a, ZCLL04, LC06, LC05, WB05, LC07a, ZTW07, Pap08, ZL06, RP08, RPTW08, ZHTW08, CSB+05, Wit08a]. Query-driven based peer descriptions are similar to the query-driven based document descriptions in cooperative P2PIR but in the peer level as well, which stores information about queries and corresponding satisfied peers in the past. Zeinalipour-Yazti *et al.* [ZYKG05, ZYKG07] and Wu *et al.* [WM09] represented peers by the most recent satisfied queries in the past and the number of documents returned to the corresponding queries. In the Zeinalipour-Yazti *et al.* approach, each peer should continuously monitor and record the past queries it sent and responses it received. Interest-based peer descriptions [DMP+09, RP08] represent peers by the number of interests covered in each peer. By using information filtering methods, the interest sets of each peer can be extracted by analysing documents. Interest-based peer descriptions are based on the assumption that if users were interested in some topic documents in the past, they are likely to be interested in the same topic in the future. The cost-based peer description is proposed by Nottelmann *et al.* [NF07b, NF07a, NFTN05], which contains cost information on computation time, communication time for sending queries and documents through the network, and money cost for accessing or downloading documents.

The most widely used peer description in uncooperative P2PIR so far is the full-text based peer description, which is same with cooperative P2PIR systems. In uncooperative P2PIR, each peer may provide their peer descriptions to the network instead of document descriptions. Transferring full-text based peer descriptions over the network produces a large amount of network traffic, which makes this approach difficult to scale to a large network. The second problem of peer descriptions in uncooperative P2PIR is how to compress peer descriptions. To address this problem, two approaches have been used to compress the size of full-text based peer descriptions in uncooperative P2PIR, which are *Bloom filter* and *top-ranked term selection*. Dazzi *et al.* [DMP+09] and Papapetrou [Pap08] applied the Bloom filter to reduce the size of full-text based peer descriptions, while Witschel *et al.* [Wit08a, WB05] employed top-ranked term selection to cut the low weight terms in a peer description. Witschel *et al.* [Wit08a] studied the different thresholds of top-ranked terms in descriptions to find the trade-offs between the compressed peer descriptions and retrieval performance.

The last problem of peer descriptions in uncooperative P2PIR is how to obtain peer descriptions. In uncooperative P2PIR, peers may or may not provide peer descriptions. When peers do not provide any information to the network, query-based sampling [CC01] is applied to extract the full-text based peer descriptions such as [LC06]. Same with the challenges of obtaining peer descriptions in cooperative P2PIR, one of the most challenging problems in uncooperative P2PIR is how to obtain term statistics (especially, global terms statistics) for peer selection and result merging. To address this problem, two approaches have been used to estimate global term statistics, which are sampling documents [Wit08a] and using a reference corpus [WB05, CSB+05, Wit08a]. Sampling documents is same with the approach in cooperative P2PIR systems, which has been discussed previously. By using a reference corpus to estimate global term statistics, the "global term statistics" of a reference corpus are extracted and used as the global term statistics of the network, while the reference corpus could be independent of the contents in the network. Witschel *et al.* [Wit08b] examined the estimations of global term statistics by comparing two approaches, and they concluded that sampling is more attractive in global term statistics estimation than a reference corpus.

2.2.3.2 Peer Retrieval and Selection

Peer retrieval is to locate peers that can provide documents which are likely to satisfy the selection criteria. To publish and retrieve documents for uncooperative IR, two different approaches have been used in structured and unstructured P2P networks, which are similar to document publication and retrieval in cooperative P2PIRs, but at the peer level. To be specific, DHT is used to publish and retrieve documents in structured P2P networks [CSB+05, ZHTW08, RPTW08, RP08, ZTW07, Pap08], while flooding and a random walk have been applied to publish and retrieve documents in unstructured P2P networks [NFTN05, WM09, ZYKG07, ZYKG05, ZL06, LC07a, WB05, LC06, LC05, LC03, ZCLL04, LC04a].

The same as document retrieval in cooperative P2PIRs, peer retrieval methods also suffer from a number of drawbacks such as high network transmission costs and poor retrieval quality. The key issues to reduce network communication costs and improve retrieval quality are to minimise the number of messages between peers and the number of peers to be queried for each search [ZYKG07]. To

address these problems, several optimal approaches have been developed for efficient searching and improvement of retrieval quality. These include *content-based network overlay, query expansion, query history* and *reducent checking*.

A content-based network overlay has been described in cooperative P2PIR and has also been widely used in uncooperative P2PIR systems [RPTW08, Pap08, RP08, LC07a, WB05, LC06, LC05, LC03, ZCLL04, LC04a]. Query expansion means to reformulate queries to improve retrieval quality. Witschel *et al.* [WB05] and Chernov *et al* [CSB$^+$05] employed topic words to increase the probability of matching the query with relevant documents. Topic words are extracted from the local relevance feedback, in which high-term frequency in top-ranked documents can be added to the query. Witschel *et al.* [Wit08a] also expanded queries by local context analysis [XC96]. Query history retrieval means to explore the locality of past queries, which can save network and process costs. They computed the degree of similarities between current queries and past queries and forwarded the queries to the peers which had provided the relevant documents in the past [ZYKG05, ZYKG07, WM09]. One problem with decreasing the retrieval quality is that many peers in the network may provide the same documents in P2P networks. Therefore, routing a query to multiple peers providing the same documents is a waste of resources. This can reduce the quality of retrieval performance because many documents are the same in the resulting list. Wu *et al.* [WM09] proposed an approach to consider combinations of neighbours which can provide the least reducent results. The main idea is to send queries to the peers which are likely to have the required documents, but not the same to the other selected peers.

One major problem with uncooperative P2PIR systems is how to select a set of peers which are likely to provide the required documents to satisfy the selection criteria. Two approaches have been proposed so far to select a set of peers: *(i)* heuristic-based methods compute the ranking scores of peers, and then retrieve a constant number of peers from the top-ranked peer list; *ii)* a theoretic-based method selects a set of peers within the minimum cost [NFTN05, NF07b]. Most peer selections in uncooperative P2PIR systems employ the first method to rank peers based on a set of peer descriptions and a given query. A set of traditional resource ranking algorithms have been adapted to rank peers. For examples, CORI [Cal00] is applied by [ZHTW08], K-L is [XC99] employed by [CSB$^+$05, ZL06, LC07a, LC06, LC05, LC03, ZCLL04], and VSM [BYRN99] is

used by [ZYKG05, ZYKG07, RPTW08, ZTW07]. The challenge of applying these algorithms is how to obtain the term statistics to compute peer scores. In contrast to the heuristic-based methods, the decision-theoretic framework (DTF) [Fuh99] has been adapted for peer selection in uncooperative P2PIR, which takes cost (e.g., time, money) [NFTN05, NF07b]. This system can compute a clear cut-off value on which peers need to be searched and the number of documents to be retrieved from each of these selected peers. Compared to the heuristic-based approaches, this work has a better theoretical foundation.

2.2.3.3 Result Merging

When a number of documents have been returned from the selected peers, how to integrate them into a single ranked list is a challenging problem in uncooperative P2PIR. This is one of the most significant differences between cooperative IR and uncooperative IR. Merging the results is difficult, because the document scores returned from each peer are not comparable for two reasons, which were discussed in [SC03]: *(i)* each peer may employ different ranking algorithms; and *(ii)* the statistics used to compute peer scores vary with different peers, because peers may use different stemming algorithms and store word lists to generate statistics.

SESS [LC04a, LC05] extended the Kirsch result-merging algorithm [Kir] by using the aggregation of super-peer descriptions instead of global corpus statistics. The limitation of this approach is that SESS requires each peer to provide the statistics of its returned documents, therefore, a strong cooperative relationship is necessary. Although Chernov *et al.* [CSB+05] combined the language modelling score and pseudo-relevance feedback language model score to re-rank documents, this also requires peers to provide the statistics of each returned document. Jie *et al.* [LC06] adapted the semi-supervised learning result merging algorithm [SC03] to re-rank retrieved documents. In order to normalise the document scores, they used the same document with different scores returned from different super peers as training data to learn the score normalising function.

2.2.3.4 Implementation

Similar to the implementation of cooperative P2PIR systems, the implementation of uncooperative P2PIR systems involves two issues, namely, *distributed system architectures* and *data management protocols*.

In completely decentralised P2P networks, each peer should provide the same capabilities, in which the system is designed to store peer descriptions, process and forward queries, select peers and merge results. In hierarchical unstructured P2P networks, super peers should provide the above functions instead.

For data management protocols, several uncooperative P2PIR systems [CSB+05, ZTW07, ZHTW08] have extended the current data management protocols of structured P2P networks to publish, retrieve and update data. A number of uncooperative P2PIR systems employ unstructured P2P network data management protocols [RP08, ZL06, NFTN05, WM09, RPTW08, ZYKG07, ZYKG05, LC06, LC04a, LC05, LC07a, WB05, LC03, ZCLL04] to fulfil the information retrieval tasks. Two uncooperative P2PIR systems are designed to work in hybrid networks. Hence, data management protocols of unstructured and structured P2P networks are employed in one system [Pap08, DMP+09].

2.3 Trust in P2P Networks and Information Retrieval

The function of trust-aware P2PIR systems is to retrieve not only relevant but also trustworthy documents for given queries. Since relevance has been described in the previous section, another key target of trust-aware P2PIR, trustworthiness, will be discussed in this section. Existing work related to trustworthiness in P2P networks and IR will be surveyed in the following paragraphs.

Trust is an important component of computer science, ranging from e-business and agent systems, to distributed systems. It has been studied in a variety of literature. One of the most widely-used trust definitions, from Olmedilla *et al.* [ORMN05], refers to actions which define trust: *"Trust of a party A to a party B for a service X is the measurable belief of A in that B behaves dependably for a specified period within a specified context"*. Thus, trust management means to collect, analyse and present evidence of security or dependability for the purpose of making assessments and decisions regarding trustworthy relationships.

Trust plays an important role in P2P networks because P2P networks lack any centralised infrastructure, but rather depend on the voluntary participation of peers to exchange information. Since vast amounts of untrustworthy information are spread across a network, it is necessary to provide a solution to help users make recommendations and judgements of the reliability and quality of resources.

Most of research on P2PIR focuses on finding a set of relevant documents for a given query; document trust (e.g., document quality and reliability) is usually ignored. Basically, research of trust in Information Retrieval is motivated by retrieving trustworthy documents rather than relevant documents. This section provides an overview of trust in both P2P networks and IR communities. The general categorisation of trust is described in Section 2.3.1, current reputation-based trust management systems in P2P networks are discussed in Section 2.3.2, and a review of work relating to trust in IR is made in Section 2.3.3.

2.3.1 Categorisation of Trust Management Systems

Nowadays, most trust management systems can be grouped into two major categories, which are, *policy-based trust* and *reputation-based trust* [BDOS05, AG07]. Both trust management systems are widely used and designed for different environments. Policy-based trust management systems are designed for structured, organised environments such as centralised networks, while reputation-based trust management systems are developed for unstructured user communities [BDOS05]. These two trust management systems will be discussed in the following paragraphs.

Policy-based trust management is a *"hard security"* mechanism with a set of clear evidence (e.g., signed statements, trust certification or credentials) to verify whether or not the specific resource can be trusted. Atrz and Gil [AG07] state that *"policies describe the conditions necessary to obtain trust, and can also prescribe actions and outcomes if certain conditions are met"*. In policy-based trust, users and services need to exchange their credentials to establish trustworthy relationships with each other. Credentials normally refer to signed statements about an entity, which are issued and verified by a trusted third party which serves as an authority. One simple example of policy-based trust is the login system. When a user provides an application and a number of private facts, the system administrator identifies whether or not the user can be trusted. If the application is accepted, a valid user with the correct username and password, which is given and identified by the system administrator, can log into the system. In policy-based trust management systems, the access decision is typically decided by a well-defined authority. Policy-based trust has two major drawbacks: *(i)* in order to obtain credentials, users have to sacrifice some privacy. For example, when a user wants to register a system, the system administrator always requires

the user to provide some private facts for verification; *(ii)* policy-based trust relies on a trusted third party, therefore, the system risks suffering a single point of failure.

Reputation-based trust can typically be called a *"soft computation"* mechanism to estimate trust value by past experience. Reputation-based trust management systems employ users' local experience or other users' experience, possibly combined, to help users to judge the reliability or quality of items and predict future behaviour. Typically, reputation-based trust is a way to estimate trust by utilising community-based feedback about past experience of items. For example, in ebay, buyers or sellers review the trustworthiness of sellers or buyers based on feedback of past transactions, which includes positive feedback, negative feedback and neutral feedback.

In a survey by Suryanarayana *et al.* [STS+04], social network[2] based trust management systems were mentioned. Social network based trust management systems use past experience between peers to compute the trust relationship between a pair of peers, which is same as the trust mechanism in reputation-based trust systems. Moreover, social relationships between peers and social groups are analysed and integrated to calculate the trust values of items. Since social network based trust management systems also employ past experience between peers to compute trust, social network based trust can be included into reputation-based trust, as in most surveys [BDOS05, AG07].

Numerous research focuses on policy-based trust by trusted third parties [YWS01, YW03, WWJ00, BO05, Ati02, KGM95]. However, this is not applicable to P2P networks where there is no central server serving as a trusted third party. Reputation-based trust management systems can be seen as a way of building trust through social control by utilising community-based feedback about past experience, which may help users make judgements about the reliability of items and peers. Therefore, reputation-based trust management systems in P2P networks will be explored in the next section.

[2]A social network is made of individuals, which are connected by a number of relationships such as friendship, financial exchange and interest.

2.3.2 Reputation-Based Trust Management Systems in P2P Networks

Current P2P networks contain three primary types of adversaries, namely, *malicious peers, front peers* and *selfish peers*. Malicious peers cause harm to the targeted items, peers or the whole network [MGM06]. Front peers can act like good peers, while providing false feedback about malicious peers to increase the reputation values of malicious peers [WN09]. Selfish peers use resources, but only contribute few or no resources [MGM06].

In general, a reputation-based trust management system consists of three basic components [KT06], namely, *information gathering, reputation estimating* and *reputation representation*. To address the different problems produced by adversaries, the various design issues of each component need to be considered. The challenges of building reputation-based trust management systems in P2P networks mainly focus on three aspects [Mai]: *(i)* how to address the problems of different types of peers behaviour, such as misleading feedback of transactions from front peers and malicious peers; *(ii)* how to define trust factors and metrics according to various contexts; *(iii)* how to build a reputable trust management systems in decentralised P2P networks, which are efficient, reliable in calculating trust values, data storage and dissemination.

Two major ways have been proposed to estimate trust values in reputation-based trust management systems, namely, *probabilistic estimation* and *social network estimation* [STS+04, DA06]. Probabilistic estimation means to analyse the characteristics of peers and use aggregated feedback to estimate the reputation values of items and peers. Social network estimation means aggregating the entire feedback in a social network and weighting it against the relationships between peers to calculate the trust value of a specific peer or item. Despotovic [DA06] studied both approaches by combining various classes, and he found that social network estimation cannot be widely used as probabilistic estimation. Moreover, probabilistic estimation performs better for a small fraction of collusive peers, while social network estimation runs better in around half of the peer population. Several reputation-based trust management systems are described in the following section according to the taxonomy of the three primary types of adversaries in P2P networks.

2.3.2.1 Reputation-Based Trust Management Systems for Malicious Peers

UniTEC [KTR05] modified Chaum Mixes [Cha81] to achieve the unlikability of peers in the network, which enables anonymity for the properties of senders and recipients. The approach of Chaum Mixes uses peers as intermediaries to pass messages to other peers, so malicious peers cannot observe the communication between individual peers. The UniTEC reputation-based trust system provides untraceable P2P communications between pseudonyms, which makes trust pseudonyms in P2P networks work strongly. UniTEC is implemented in structured P2P networks.

Stakanova *et al.* [SFWC04] developed a reputation-based trust system to filter malicious peer threats in Gnutella. The reputation value of a peer is calculated by the contributions of a peer to the network in terms of resources upload, resources download and traffic extensiveness. A pre-defined threshold is set to distinguish trust from untrust.

TrustGuard [SXL05] is a highly dependable reputation-based trust system, which is designed to minimise the damages produced by malicious peers in structured P2P networks. The architecture of TrustGuard consists of three components, namely, Trust Evaluation Engine, Transaction Manager and Trust Data Storage Service. The trust value of a peer is estimated by weighting its current reputation, reputation history and reputation fluctuations. To filter fake transactions, TrustGuard defines a policy whereby peers can leave feedback when they truly transact with each other.

Liau *et al.* [LZBT03, OLT03] developed a reputation-based trust system with Public Key Infrastructure (PKI) [Mau96]. In the system, each peer stores its own reputation value by a certificate *RCert*, which consists of rating information about transactions with other peers in the past. To update the *RCert* in the network, two protocols in terms of *RCertP* and *RCertPX* have been proposed.

Repantis and Kalogeraki [RK06] designed a decentralised reputation-based middleware for unstructured P2P networks. The advantage of a middleware approach is that it can facilitate secure peers interoperability without users intervention. This makes it hard to identify peers in the network. The reputation value of a peer is stored in its neighbouring peers and packaged with users' queries on request.

GossipTrust [ZHC08] is a global reputation computation system, which can

be implemented in both structured and unstructured P2P networks. The system employs a Bloom filter to save an amount of space to store reputation data. Gossip-based protocols are designed to reduce the computation and communication overhead for global reputation aggregation. In addition, a score error approach is designed to improve the accuracy of the reputation values of peers. Since the high aggregation of global reputation scores requires more computation time and storage overhead, tradeoffs have been studied to select the appropriate parameters in GossipTrust.

Aringhieri *et al.* [ADDV$^+$06] employed *P2PRep* [DdVPS03] protocols to disseminate and access reputation information in the network. Peer reputation values are calculated by fuzzy aggregation [Yag88] on two levels, including local reputation and network reputation. Local reputation is the past interaction experience between other peers, while network reputation is the synthesis formed by aggregating multiple opinions of a peer from other peers in the network. When a peer has no previous experience of another peer or the local reputation value is not sufficient, the network reputation should be employed.

XRep [DdVP$^+$02] is a protocol consisting of five phases of reputation data management in P2P networks. These are *resource searching, resource selection and vote polling, vote evaluation, best servant check* and *resource downloading*. Since the same resource may be available in different peers (e.g., good peers and malicious peers) with different predictable reputation values, XRep combines resource reputation and peer reputation to improve the accuracy of the reputation values of the resources.

PowerTrust [ZH07] is a reputation-based trust management system, which is inspired by the power-law distribution [FFF99] of ebay users' feedback. By studying the 108 MB feedback data of 10,000 users, it was found that feedback from a reputable peer is more reliable than that from a low reputable peer. Therefore, to calculate a peer reputation value, it is not necessary to collect all of the feedback from all of the peers which have had transaction experiences with that particular peer in the network. PowerTrust selects a small number of the most reputable peers by a distributed ranking algorithm. By leveraging the feedback from these reputable peers, the global reputation value of a peer can be rapidly calculated. PowerTrust can be implemented in both unstructured and structured P2P networks.

Selcuk *et al.* [SUP04] proposed a reputation-based trust management system

for unstructured P2P networks. For a given query, the responding peers are grouped together and the reputation value of the peers in the group are calculated by feedback from each other. Finally, the peer with the maximum trust value and minimum distrust value is selected to fulfil users' requirements.

Credence [WS06] computes files reputation values by using an explicit voting system, which is different with most of the systems which employ an implicit endorsement of a file. Before using a file, a user collects votes for that file from its neighbours in order to calculate the file reputation value. However, if there is no sufficient voting activities from its neighbours, Credence employs a flow-based approach by analysing the trust relationships from its neighbouring peers to more distant peers by weighting the votes from neighbours and distant peers. Credence is designed to be implemented in unstructured P2P networks.

EigenTrust [KSGM03] calculates the global trust value of each peer in the network. Each peer maintains the statistics of all its transactions with other peers locally. The global trust value of a peer is computed by weighting every local trust value of peers. Then, the peer with the highest trust value is selected. EigenTrust relies on some pre-trusted peers, which are assumed to be trusted by all peers. This can be problematic, since once pre-trusted peers receive negative feedback after some transactions, the system may not work reliably.

PeerTrust [XL04] is a transaction-based reputation system which evaluates the trust values of peers in the network. To compute trust, PeerTrust identifies three basic trust parameters and two adaptive factors, namely, feedback of a transaction, the total number of transactions a peer performs, the credibility of the feedback sources, transaction context factor and the community context factor. A general trust model is generated to integrate these five parameters in a coherence scheme by weighting the feedback of transactions from different peers. PeerTrust is designed to be implemented in structured P2P networks.

H-Trust [ZL08] is designed to reduce reputation information communication and computation overhead. To compute the trust values of peers, H-Trust collects the feedback from good peers by h-index aggregation [Hir05] to save network costs. H-Trust consists of five phases, which are *trust recording, local trust evaluation, trust query, spatial-temporal update*, and *group reputation evaluation*. Tradeoffs have been studied between the accuracy of trust values and the network cost.

FileTrust [KLK07] is a reputation-based trust system for both resources and peers. The trust value of a resource is the same as the value in EigenTrust and

PeerTrust. The trust value of a peer is represented by how many trustworthy resources it currently shares. The resource with highest trust value and the lowest contribution value is selected as the final choice. FileTrust also develops the system architecture and data management protocols so that it can be implemented in structured P2P networks.

2.3.2.2 Reputation-Based Trust Management Systems for Front Peers

Swamynathan *et al.* [SZA05] developed a reputation-based trust system to combat fake or misleading feedback from peers which provide honest services. To address this problem, they used two sets of reputation ratings, namely, service rating and feedback rating. By decoupling the service and feedback reputation, this approach can filter the front peer threats. However, in their system, the reputation information of peers is stored locally, which means malicious peers and front peers can easily modify their reputation ratings.

Poisonedwater [WN09] is a social-network based reputation system, in which the reputation value of a peer is calculated by the peers' social position. This approach injects poisoned water (PW) to the network to identify malicious peers and front peers. By analysing the adaptive Spreading Factor(SF) from PW, the percentage of a peer's reputation value, which can be flooded by its neighbours, is decided. The experimental results show that, compared with Eigentrust and Powertrust, Poisonedwater can reduce the error ratio of trust values significantly, since the feedback from the front peers is filtered.

GroupRep [TZWC06] is a social-network based reputation system, which assumes that peers are in different social groups. There are three kinds of trust relationships in GroupRep, namely, trust relationships between peers, trust relationships between groups, and trust relationships between peers and groups. The trust value of a peer is computed by the peer local and group reputation information. A filtering algorithm is designed to filter the feedback, not only from malicious peers, but also from front peers.

TrustRRep [SKT08] is a reputation-based trust system to identify which peers give dishonest feedback. The system computes a credibility ratio of peer feedback to identify whether or not a peer is a liar. A number of pre-trusted peers are employed to manage the reputation information in the network. A threshold is defined to distinguish trusted peers from liar peers (e.g., malicious peers and

front peers). TrustRRep also develops data management protocols in P2P networks. These are: *join, query and queryhit, selection and download*, *evaluation and update*.

2.3.2.3 Reputation-Based Trust Management Systems for Selfish Peers

Andrade *et al.* [AMCB04] developed a reputation-based trust system to discourage selfish peers, which make it unlikely that free-riding can build up a high reputation in the network. In their approach, the reputation information of a peer is computed and stored locally according to the total value of the resources donated by the peer. The reputation information of a peer is updated when other peers interact with it.

Sears *et al.* [SYG05] designed an adaptive reputation-based trust framework to avoid selfish peers to cooperate with malicious peers. They defined three quantifiable metrics, namely, *job satisfactory ratings, reputation* and *trust*. The reputation value of a peer is calculated by its satisfactory job ratings. The trust value of a peer is computed by the reputation value of the peer from its own and other peers' views.

2.3.3 Trust in Information Retrieval

It should be noted that a number of documents which are relevant for a given query in Information Retrieval systems may be returned, and a few of these will be selected by users or agents to review or download. Most of the research into Information Retrieval focuses on finding relevant documents for a given query, where the systems are assumed to be perfect in efficiency, consistency and reliability. Trust of documents in terms of reliability and quality is usually ignored [ZG00, Lyn01]. Therefore, it is necessary to integrate trust and provenance into the next generation of Information Retrieval systems [Lyn01]. Trust in Information Retrieval is motivated by retrieving high-quality and reliable information. Unfortunately, the questions of formalising trust in Information Retrieval systems have been poorly explored so far. Several of these will be studied in the following paragraphs.

TrustRank [GGMP04] is proposed to use the relationships between Web pages to filter spam. In Web environments, a number of spam Web pages are created to mislead search engines in order to achieve a higher ranking on the result pages

than they deserve. In fact, spam Web pages can be evaluated by human beings, but it is expensive. TrustRank is developed based on the assumption that good pages rarely link to bad pages, but bad pages always link to good pages to improve their ranking scores. TrustRank pre-selects a small set of Web pages trusted by experts, and then analyses the links between these trusted Web pages and other Web pages. The limitation of TrustRank is that it is only effective when the link information is available.

Kaza *et al.* [KMF08] proposed a reputation-based trust management system to evaluate items in social information retrieval environments [GF07]. The trust value of an item is a function which combines the approval votes and the reputation of the voters which could be authoritative bodies, reviewers, and the citation network. Their work is highly abstract and there is no experimental evaluation.

Abdul-Rahman and Hailes [ARH99] proposed a reputation-based trust model to determine the trustworthiness of document providers based on statistics of different peers' experiences. The trust value is computed by the local trust experience and recommenders' trust experience. Their approach can help users to reduce the problem of the *reliability of retrieved information, semantic mapping between agents* and *complexity*. This approach is similar to most of the approaches in social network based reputation systems in P2P networks.

Clarke *et al.* [CCL01] developed a question-answering system to select one of the documents in the resulting list as the final answer. Their research problem can be viewed as how to determine which answer can be trusted. The proposed solution is to compute the time of the redundancy of the documents in the resulting list. The most frequent one is the most likely to answer the query and be capable of being trusted.

Quality-based document retrieval was proposed by Zhu *et al.* [ZG00]. They selected 6 metrics as information qualities, namely, *currency, availability,authority, popularity, information-to-noise ratio*, and *cohesiveness*. In their approach, trust is evaluated by Yahoo Internet Life (YIL) reviews [ZDN], which gives a score to a Web page from 2 to 4. If a Web page has not been reviewed by YIL, the trust value (authority) is set to 0. The proposed approach is a policy-based trust mechanism.

2.4 Discussion

Trust-aware Information Retrieval in P2P networks is motivated by the need to find not only relevant, but also reliable and high-quality documents. A review of the literature related to trust-aware P2PIR was made in the previous sections, including issues on P2PIR and trust in P2P and IR. This section will discuss the limitations of the existing P2PIR systems and trust management systems. In addition, the reasons why these systems cannot address the problem of trust-aware P2PIR will be explained.

For full-text based Information Retrieval in P2P networks, a number of traditional document ranking algorithms and resource selection algorithms have been applied or modified to address the document ranking and peer selection problems in cooperative and uncooperative P2PIR scenarios. An important problem in terms of adapting the family of these algorithms, is how to obtain local and global term statistics. Since there is no peer with a global view of the network, the most challenging problem of applying these algorithms is how to obtain the global term statistics of underlying documents or corpus in the network. Most of existing systems assume that global term statistics can either be available in advance, or be generated by *(i)* sampling documents; *(ii)* using a reference corpus; and *(iii)* setting an arbitrary value. The limitations of these methods will be discussed in the following paragraph.

Sampling a set of documents to estimate global term statistics was initially proposed by Viles *et al.* [VF95], and this method has been widely used in P2PIR in both scenarios. However, sampling a set of documents has the following drawbacks: *(i)* when updating global term statistics, peers need to collect sample documents again from the network, re-calculate the global term statistics and disseminate the new statistics to other peers in the network. The whole process is expensive in terms of both computation and timing. This generates high network overheads; *(ii)* according to a study of the dynamics of P2P networks, the network members change completely roughly every 8 hours [TD07], indicating that, in such environments, the process of updating global term statistics needs to be significantly frequent when keeping global term statistics up-to-date; *(iii)* a sample of 30% to 40% of the whole collection is sufficient to estimate accurate global term statistics [CR01], but sampling a small set of documents in the network will not be accurate until 30% to 40% of the documents have been sampled. A peer contacts roughly 30% to 40% of peers in the network, which requires

extensively high costs of computation, storage, time, bandwidth and messaging routing when the underlying network is large-scale; *(iv)* when sampling a set of documents, it is not easy to be consistent in document ranking, peer selection and result merging. For example, a peer can sample the documents from its neighbouring peers. Since different peers may have different neighbouring peers, the sampled documents could be highly different in content. The result is that different peers may have various global term statistics. The global term statistics generated by sampling are dependent upon the location of the peer, and are not comparable for ranking and result merging [Wit08b] unless one peer in the network acts as a central peer to sample the documents and disseminate global term statistics to other peers. However, the fact is that there is no central peer in a P2P network, so sampling is not practical for estimating global term statistics for real P2PIR.

Peter *et al.* [PJS+98] propose to use a reference corpora to estimate global statistics. However, estimating the global term statistics of a P2P network by using a reference corpora has three limitations: *(i)* if the network contents are different from the contents of the reference corpora, the estimated global term statistics extracted from the reference corpora could be useless. For example, the network may be sharing information about medicine. However, if the reference corpora is computer science, then the estimated global term statistics could be significantly different and useless; *(ii)* even if the reference corpora shares the same topic as the network, the reference corpora is static, which cannot properly represent the highly dynamic global term statistics in the network; *(iii)* it should be noted that the size of the reference corpora is an important variable [PJS+98]. For specific networks, the size of the reference corpus should be different and dynamic. So far, there is no experimental evaluations to examine the parameter between the network size and the reference corpora size. Because of this, using a reference corpora is also not a good choice to estimate global term statistics for P2PIR.

Simply setting the value of global term statistics is not practical for the following reasons: *(i)* what value is optimal for a particular network? *(ii)* how can the optimal values be obtained? *(iii)* how can the dynamics of P2P networks be addressed? These questions are hard to answer. Therefore, sampling documents, using a reference corpus and setting an arbitrarily big value are not fit for real

P2P networks which are highly dynamic and distributed. An effective and practical solution is necessary to estimate global term statistics for document ranking, peer selection and result merging.

Currently, trust management systems in both P2P networks and Information Retrieval focus on how to identify entity trust (e.g., policy-based trust and reputation-based trust). However, entity trust is insufficient, because trust is a dynamic phenomenon rather than a static one [JT99, FC04, FC00, BK00]. Falcone *et al.* [FC04] suggest that resources with good trust values may be useless if they are not taken in the context of the resources. Entity trust is not enough in some application areas which are required to select final answers from a large number of resources [GA07]. This is because a number of useful and useless resources could be assigned with the same entity trust values regardless of contents and queries. P2PIR is an environment where the primary goal is to select relevant information from a ranked result list to satisfy users' information needs. Without taking into account queries and document contents, many relevant and irrelevant documents may have the same entity trust values. Since users do always care about the trustworthiness of documents which are relevant to the given query, they may not be interested in the trustworthiness of irrelevant documents in the network. Current entity trust systems cannot distinguish which documents are not only trustworthy but also useful to users.

Having discussed the limitations of existing systems, the reasons why these systems cannot address the problem of trust-aware P2PIR will now be explained:

- Since both relevance and trustworthiness are crucial factors for information retrieval in P2P networks, it would be desirable if there was a system which could retrieve not only relevant but also trustworthy documents for given queries. Unfortunately, existing systems independently focus on either relevance or trustworthiness, not both at the same time. There is no one system to find relevant *and* trustworthy documents simultaneously. The proposed, in this thesis, trust-aware P2PIR system will be designed to address this problem in cooperative and uncooperative P2PIR scenarios.

- Most of the existing relevance-based algorithms in P2PIRs rely on global term statistics to compute document and peer scores. Because of the limitations of methods to estimate global term statistics, these relevance-based algorithms are not easy to employ in real P2P networks, which are highly

dynamic and distributed. The proposed trust-aware P2PIR system provides an effective and practical method to estimate global term statistics by using the characteristics of structured P2P networks (e.g., routing table size). This makes it easier to employ relevance-based document ranking and peer selection algorithms in real P2P networks.

- A wealth of trust management systems have been developed to establish the entity trust values of peers and files in P2P networks. Because of the limitations of entity trust, many relevant and irrelevant documents may have the same entity trust values. Thus, it is hard for users to select "useful resources" (i.e., relevant documents in P2PIR) with the same entity trust values. The proposed trust-aware P2PIR system will not only take entity trust into account, but also contents and queries to assess the content trust value of a document or document provider for a given query.

2.5 Summary

Trust-aware P2PIR are motivated by the need to find not only relevant but also trustworthy documents for given queries. This chapter makes a comprehensive review of the work related to trust-aware Information Retrieval in P2P networks, including issues of P2P network architectures, relevance-based P2PIR and trust management systems in P2P and IR. To be specific, unstructured and structured P2P networks are described by issues of data location schemes and search mechanisms. Cooperative and uncooperative P2PIR are surveyed based on subproblems such as document description, document ranking, peer description, peer selection, result merging and implementation. Moreover, the related work to trust management systems in P2P networks and Information Retrieval are represented individually. The limitations of existing systems have been discussed in this chapter.

Chapter 3

Generic Trust-Aware P2PIR System in P2PIR Environments

3.1 Introduction

The previous chapter presented a comprehensive review of the work related to trust-aware P2PIRs. Currently, there are two major scenarios of Information Retrieval in P2P networks, namely, cooperative P2PIR and uncooperative P2PIR. Peers in the P2P network are dynamic, and self-organise to adjust the network structure. One problem which must be addressed is how to determine a mechanism to organise peers in a cooperative manner so that relevant and trustworthy scores can be computed by any peer in the network. This chapter makes the following two major contributions to address this problem:

- A generic decentralised system architecture of the proposed trust-aware P2PIR system.
- A set of generic data management protocols of the proposed trust-aware P2PIR system in structured P2P networks.

Before designing trust-aware P2PIR systems, it is necessary to understand the existing P2P criteria and discuss the kind of P2P criteria which can be considered and which cannot. In the remainder of this chapter, a set of criteria for the P2P paradigm is described in Section 3.2, and the remit of the proposed trust-aware P2PIR system is also discussed. The generic system architecture and data management protocols of the proposed trust-aware P2PIR system are introduced

in Section 3.3, and an example is also provided in this section. The chapter is summarised in Section 3.4.

3.2 Criteria of P2P Paradigm

Applications (e.g., file sharing, streaming, distributed computing) in P2P networks rely on the cooperation of voluntary peers. So, what criteria should be considered for design when developing a P2P application? To answer this question, Roussopoulos *et al.* [MR04] identified three criteria, which are *self-organising, symmetric communication* and *decentralised control*. They believed that these criteria were important when assessing a P2P application design. Milojicic *et al.* [DSM03] identified eleven P2P criteria, namely, *decentralisation, scalability, anonymity, self-organisation, cost of ownership, ad-hoc connectivity, performance, security, transparency and usability, fault resilience* and *interoperability*. Since Milojicic's P2P criteria include Roussopoulos's criteria, Milojicic's criteria are used in this dissertation. Moreover, some of P2P criteria can be handled in the proposed trust-aware P2PIR system, and these are discussed in each P2P criterion.

- *Decentralization* is one of the most important criteria for P2P application design, since it emphasises users' ownership and control of data and resources [DSM03]. Therefore, decentralisation must be considered when designing a P2P application. Based on the existing system architectures of P2P applications, two kinds of architectures are introduced, namely, completely decentralised system architecture (e.g., completely decentralised unstructured P2P networks in Section 2.1.1 and structured P2P networks in Section 2.2.2), and hierarchical system architecture (e.g., hierarchical P2P networks in Section 2.1.1) [DSM03]. Since structured P2P networks are selected and used as the basic architecture to implement the proposed trust-aware P2PIR system in this dissertation, a completely decentralised system architecture is employed (in Sections 3.3.1, 4.3.1, 5.3.1 and 6.4.1). In the proposed trust-aware P2PIR system, documents are stored locally.

- *Scalability* is one of major criteria of P2P networks which can aggregate the capabilities of all of the participating peers in the network to achieve

scalability. In other words, as peers arrive and the demand on the system increases, the total capacity of the system also increases [LCP+05], which can increase system scalability. Structured P2P networks provide more scalability than unstructured P2P networks [RV03, LLH+03, LCP+05]. The proposed trust-aware P2PIR system is designed based on structured P2P networks. Peers in the proposed system comprise an overlay network, and each peer only maintains information about a small number of other peers, which increases the system scalability. The detailed implementation of the proposed system will be described in Sections 3.3, 4.3, 5.3 and 6.4. Moreover, system scalability is related to communication cost, synchronisation cost and load-balancing [DSM03], and communication cost consists of bandwidth cost, search latency, etc [ZS05, YDRC06]. One of the main problems of P2PIR is that copies of documents or document statistics are significantly large in size. When publishing or retrieving documents, the copies or statistics have to be transferred over the network, which may produce a large amount of network traffic [LLH+03]. To address this problem, different approaches to compress the size of document copies have been explored, and these were introduced in Sections 2.2.2.1 and 2.2.3.1. Although the proposed trust-aware P2PIR system does not develop any new approaches to reduce the size of document copies and statistics, existing approaches (e.g., bloom filter, top-ranked term selection, feature extraction described in Sections 2.2.2.1 and 2.2.3.1) can be employed.

- *Anonymity* allows people to use the system without legal concerns, which is a criterion of P2P networks [DSM03]. There are three different kinds of anonymity between communicating peers in P2P, namely, sender anonymity, receiver anonymity, and mutual anonymity [DSM03]. The proposed trust-aware P2PIR system does not support any anonymity because trust in the proposed system relies on feedback and the credibility of the feedback providers (which is described in Section 4.2.1). If anonymity is employed in the proposed trust-aware P2PIR system, malicious peers can hide in the dark.

- *Self-organisation* means that peers can recognise their neighbours and organise themselves into a network overlay without central or super peer control [MR04, DSM03]. The proposed trust-aware P2PIR system can build

upon structured P2P networks by extending existing data management schemas and routing protocols of structured P2P networks. Peers in the proposed system are dynamic, and self-organise to adjust the network structure for document and peer arrivals and departures, which is described in Sections 3.3.2, 4.3.2, 5.3.2 and 6.4.2 for different problems.

- *Ownership sharing* is one of the premises of P2P computing [DSM03]. Shared ownership may reduce the cost of owning the content and the cost of maintaining it in the network [DSM03]. In the proposed trust-aware P2PIR system, ownership sharing can be categorised as two scenarios, namely, cooperative P2PIR and uncooperative P2PIR. In cooperative P2PIR environments, ownership can be shared without limitation, for example, public web documents (in Chapter 4). On the other hand, in uncooperative P2PIR environments, peers may not provide document ownership due to proprietary or financial cost issues (see Chapter 5).

- *Ad-hoc connectivity* effects the P2P networks in the real world [DSM03], since not all the systems perform the same application at the same time. In fact, some peers are mostly used, some are rarely employed, and the remainder are never used [DSM03]. Ad-hoc is also the nature of a centralised Web search engine and digital library search [KPJD05]. However, theoretically, each peer in structured P2P networks is assumed to provide equal capabilities [LCP+05, SMK+01, RFH+01, RD01, MM02, ACMD+03, MBR03, ZKJ01]. Since the proposed trust-aware P2PIR system is designed based on structured P2P networks, the system proposed in this dissertation initially assumes that each peer can act equally (e.g., each peer stores a similar number of documents in testbeds in Section 4.4.1). In further research, this ad-hoc nature will be taken into account before employing the proposed system in realistic P2P networks.

- *Performance* is a critical criterion of P2P networks [DSM03]. The advantage of P2P networks is to aggregate distributed storage and computing capability to improve system performance. The performance can be influenced by resources, processing, storage and networking [DSM03]. Two questions regarding performance should be considered in P2P, namely, how long it takes to retrieve a file and how much bandwidth a query will consume. Currently,

there are three major approaches to optimise performance, namely, replication, caching, and intelligent routing [DSM03]. Replication means copying files to other peers to improve the efficiency of query routing for popular words in unstructured P2P networks, which mainly rely on flooding and random walk to locate files [LC04b, CJL+09, KWTA07, CAN02]. However, since the proposed trust-aware P2PIR employs DHT and routing protocols of structured P2P networks to retrieve documents, replication may not be used. Moreover, the proposed trust-aware P2PIR system does not provide any kind of mechanism to improve system performance.

- *Security* plays an important role in P2P networks because P2P networks lack any centralised infrastructure, but rather depend on the voluntary participation of peers to exchange information. Since vast amounts of untrustworthy information are spread across a network, it is necessary to provide a solution to help users make recommendations and judgements of the reliability and quality of resources. Current P2P networks contain three primary types of adversaries, namely, *malicious peers, front peers* and *selfish peers*, which are described in Section 2.2. The proposed trust-aware P2PIR system is assumed to filter untrustworthy documents and malicious peers from the network by extending PeerTrust (in Section 4.2), where front peers and selfish peers cannot be handled.

- *Transparency and usability* mean the transparent connection of distributed systems into a seamless local system [DSM03]. In P2P networks, applications can be used in the following three manners: *(i)* as a user of service, typically through Web interfaces; *(ii)* wrapped around non-P2P applications, typically on a P2P platform; *(iii)* as locally installed P2P software [DSM03]. The proposed trust-aware P2PIR system employs the third manner, which requires each peer to install a P2P software client locally to publish documents, rank them, and forward queries (see Sections 3.3, 4.3, 5.3 and 6.4).

- *Fault resilience* is one of the primary design goals of a P2P system in order to avoid a single point of failure. A P2P system may face various failures, such as spanning multiple hosts and network, disconnection, unreachability, partitions and node failures [DSM03]. The proposed trust-aware P2PIR system does not design any particular fault resilience mechanism, but can

employ the fault resilience of the employed structured P2P networks (e.g., Chord).

- *Interoperability* means dealing with the interoperation between different P2P systems [DSM03]. Currently, some efforts have been made toward improving interoperability by some global research groups, such as a P2P working group. The proposed trust-aware P2PIR system does not provide any protocols for interoperability with other P2P applications.

3.3 Generic Trust-Aware P2PIR System Architecture and Data Management Protocols

The objective of this section is to provide an overview of the generic system architecture and data management protocols of the proposed trust-aware P2PIR system. P2P network architectures determine the functionality and responsibility of each peer, as well as data location schemes and message-routing mechanisms. Structured P2P networks are selected to implement the proposed trust-aware P2PIR system. Designing the proposed trust-aware P2PIR system should consider the P2P criteria (discussed in the previous section). Although a Chord-based P2P network is used as the basic network architecture of the proposed trust-aware P2PIR system in this dissertation, the system can also be applied to other structured P2P networks.

3.3.1 Generic Trust-Aware P2PIR System Architecture

A structured P2P network is a completely decentralised P2P network overlay, which has no central server and super peer. Each peer in the network should simultaneously serve as document providers, users and directory services. Therefore, a user, a document provider and a directory service can be peers in the proposed trust-aware P2PIR system. When enacting these three roles, each peer in the proposed system should fulfil the following six requirements: *(i)* sending a query to the network to retrieve relevant and trustworthy documents; *(ii)* providing feedback of used documents for trustworthiness evaluation; *(iii)* generating descriptive information for relevance and trustworthiness score calculations; *(iv)* storing description information as a directory service; *(v)* processing and forwarding queries; and *(vi)* ranking retrieved documents for a given query based on

relevance and trustworthiness. In the proposed trust-aware P2PIR system, description information should provide the ranking algorithm required information to determine which documents or peers are relevant and trustworthy for a given query. Description information should contain two types of information for the computation of relevance and trustworthiness, namely, a content-based description of documents or peers, and reputational values of documents or peers. Figure 3.1 shows the proposed generic system architecture of the trust-aware P2PIR system, which consists of four components, including *Statistics Manager, Reputation Manager, Ranker* and *Data Locator*. These components will be described in detail for different P2PIR problems in Chapters 4, 5 and 6.

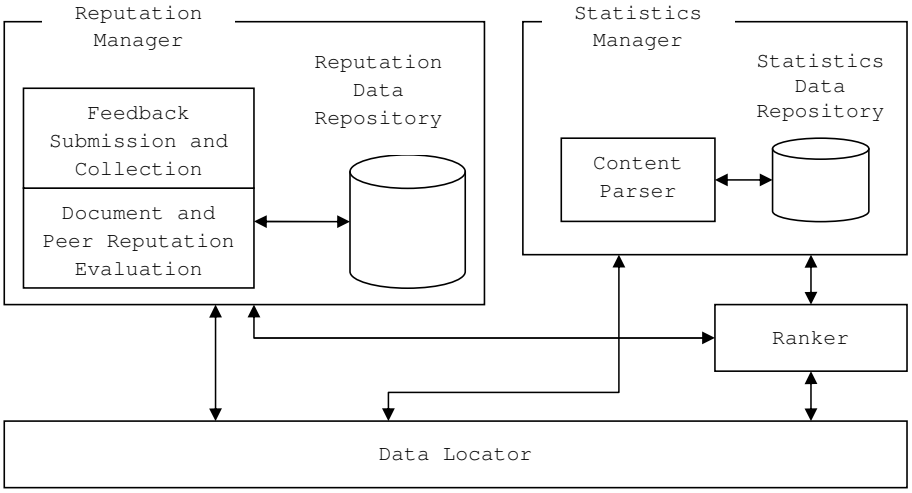


Figure 3.1: Generic system architecture of the proposed trust-aware P2PIR system

The statistics manager consists of two components, namely, *Content Parser* and *Statistics Data Repository*. The content parser is responsible for parsing local documents (in cooperative P2PIR of Chapter 4) and peers (in uncooperative P2PIR of Chapters 5 and 6) to extract content-based descriptions, such as terms and the corresponding term statistical information. A structured P2P network contains a global index and each peer is responsible for storing part of it (e.g., keys in the global key space) [GMH04, ATS04, LCP+05, CC05]. The statistics data repository is a small database which stores a portion of the global vocabulary and its associated statistics. For example, in Figure 3.2, Peer D is responsible for the term *University*, and contains a set of documents (e.g., doc b, doc c and doc e) containing the term *University*.

The reputation manager submits and collects feedback of user reviews of used documents from the network. Moreover, it can compute the reputational value of documents and peers based on users' feedback (this will be described in Section 4.2.1). The reputation manager consists of three components, namely, *Feedback Submission and Collection*, *Document and Peer Reputation Evaluation* and *Reputation Data Repository*. The feedback submission and collection component is responsible for submitting and collecting users' feedback. The document and peer reputation evaluation component is responsible for evaluating the reputational values of documents and peers. The reputation data repository stores a portion of the global reputation information.

The data locator provides a P2P data location scheme for accessing and updating data in the network. Different applications may use different data placement and location schemes, which determine how and where the data can be inserted, updated, and accessed. When implementing the proposed trust-aware P2PIR system, the data locator employs Chord routing protocols [SMK+01]. The ranker in the system collects word statistics and reputational information in order to compute the document ranking scores for a given query.

3.3.2 Data Management Protocols

Since the proposed trust-aware P2PIR system is designed to be implemented in Chord structured P2P networks in this dissertation, the data management protocols of the proposed system should extend the existing data management protocols of Chord in the context of term statistics and reputation data routing. Therefore, the proposed protocols of the trust-aware P2PIR system consist of the following three phases: *(i) join and publish*; *(ii) lookup and rank*; and *(iii) evaluate and update*.

3.3.2.1 Join and Publish

As a peer publishes its documents to the network for sharing, the content parser extracts content-based description information, such as term and term frequency, from the documents, after which the data locator forwards the description information to the peers responsible for these terms by routing protocols. Chord data management strategies define which peers are responsible for which terms (i.e., keys in Chord) [SMK+01]. Moreover, if the document is available in the network

and has already been assigned a reputational value, then the reputational value of the newly-published document is equal to the current reputational value. Alternatively, if the document has never been published before, it is assigned the same reputational value of the peer providing it. If the peer is a new peer in the network, the reputation manager should collect the reputational values of the documents it is sharing and compute its peer reputational value.

3.3.2.2 Lookup and Rank

Once descriptions have been created and stored in the network, they are ready for document and peer ranking (in Chapters 4, and 5). The lookup phase consists of two kinds of protocols: *(i)* to retrieve content-based descriptions, and *(ii)* to retrieve reputation values. The lookup messages are issued by the data locator. Having obtained the messages of content-based descriptions and reputationl values, the ranker is able to compute the document and peer scores.

3.3.2.3 Evaluation and Update

After using documents, users need to submit feedback to the network. The document and peer evaluation component in Figure 3.1 calculates the document reputational value and peer reputational value respectively, and then stores the new values in the reputational data repository.

3.3.3 Case Study: Trust-Aware P2PIR in Cooperative P2PIR Environments

This subsection provides examples of how the proposed trust-aware P2PIR system works. As shown in Figure 3.2, a set of peers (e.g., Peer A, Peer B, ..., Peer F) consists of a Chord-based structured P2P network. Peers share a number of their documents (e.g., doc b, doc m, doc x in Peer F) in the network. Each peer runs a trust-aware P2PIR system (i.e., Figure 3.1). The following examples and data management protocols listed in the previous section are one-to-one mappings.

3.3.3.1 Publishing Documents

In Figure 3.2, Peer F publishes a new document, such as doc b, to the network. Assuming that doc b only contains two terms, namely, "Manchester" and "University", the content parser in Figure 3.1 extracts content-based descriptions for

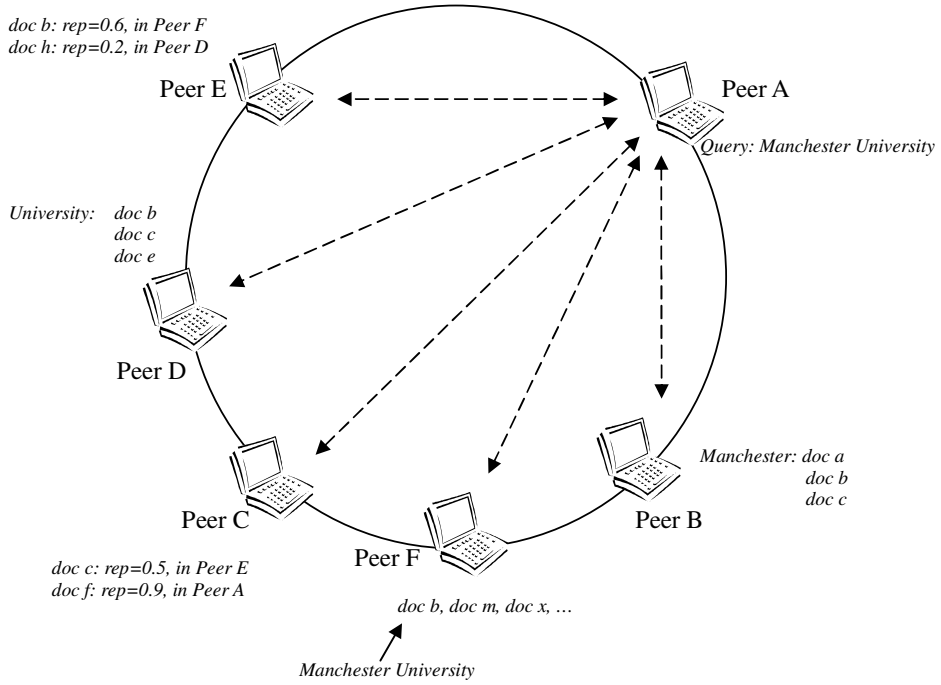


Figure 3.2: Proposed trust-aware P2PIR system in a Chord-based structured P2P network

each term, for example, "Manchester" in doc b, and "University" in doc b. Afterwards, the data locator forwards the content-based description information to the peers responsible for each of the terms, "Manchester" and "University" by routing protocols. In Figure 3.2, Peer B is responsible for "Manchester" and Peer D is responsible for "University". Peers B and D store content-based description information and serve as directory services. At the same time, the data locator forwards doc b to the peer responsible for its reputational value, which is Peer E in Figure 3.2. This assumes that doc b is available in the network and has already been assigned a reputational value of 0.6 (in Figure 3.2, doc b: rep=0.6).

3.3.3.2 Retrieving and Ranking Documents

Once Peer A (i.e., a user) sends a query (e.g., Manchester University in Figure 3.2) to the network, the data locator performs lookup messages for each term in the query to retrieve its content-based description. Peer B returns a list of documents containing the term, "Manchester", namely, doc a, doc b, and doc c. Peer D returns a list of documents containing the term, "University", in doc b, doc c, doc e. The two lists are returned to Peer A, and the ranker in Peer

A intersects the content-based descriptions to obtain the documents containing the query terms, which are doc b and doc c. Then, the data locater of Peer A performs lookup messages for doc b and doc c to retrieve their reputational values. Peer E returns doc b with a reputational value of 0.6 in Peer F (doc b: rep=0.6, in Peer F in Figure 3.2), and Peer C returns doc c with a reputational value of 0.5 in Peer E (doc c: rep=0.5, in Peer E in Figure 3.2). The ranker then computes the integrated scores of doc b and doc c (in Section 4.2.2). Assuming that the doc b score is higher than that of doc c, peer A selects doc b as a final answer and then contacts Peer F to obtain doc b.

3.3.3.3 Evaluating and Updating Reputation Values

After using doc b, Peer A should leave feedback for doc b. The feedback is forwarded by the data locator to the Peer E responsible for the reputational value of doc b. Then, the document and peer reputation evaluation component should calculate the new reputational value and store it in the reputational data repository.

3.4 Summary

When designing and developing a P2P application, it is necessary to understand the existing P2P criteria, and discuss the kind of P2P criteria which can be handled and which cannot. In this chapter, P2P criteria are considered in the design of trust-aware P2PIR. Then, the generic system architecture and data management protocols of the proposed trust-aware P2PIR system are introduced. During this process, a set of examples of the proposed trust-aware P2PIR system is provided.

Chapter 4

Trust-Aware P2PIR in Cooperative P2PIR Environments

4.1 Introduction

The previous chapter gave the generic of the proposed trust-aware P2PIR system. The objective of this chapter is to consider the problem of retrieving and ranking not only *relevant* but also *trustworthy* documents for a given query in cooperative P2PIR environments.

In cooperative P2PIR environments, document providers (i.e., peers) work cooperatively to publish their document copies or statistics to the network for retrieving and ranking. They are able to provide their full documents for users' reviewing, downloading and using without authenticated access. Moreover, users in cooperative P2PIR employ a public search engine to retrieve documents upon request. The query process of cooperative P2PIR is that a user submits a query to the public search engine. Then, the search engine retrieves a set of peers responsible for storing the locations of peers containing query terms. These peers forward the location information to the search engine by their routing tables and the message-routing protocols of the network. Afterwards, the query is routed to the peers containing the documents which satisfy the selection criteria. In response, each of these peers provides the documents for the search engine. Then, the search engine computes document scores for a given query, and ranks them in order to present them to the user [PMB06, SYYW03, SLZ+07, PLPZR08,

ZRL+07, LLH+03, LKZ+06, GKN09, TXM03, TXD03, LC04b, KJ03, JYF07, LJT07, KWTA07, XSD+08, CJL+09, RV03, NYF08a, NYF08b, Che05, TDX04, TD04, SLZ+09].

4.1.1 Assumptions

Before designing the proposed trust-aware P2PIR system in a cooperative P2PIR, the following assumptions must be made:

- Peers should provide any required information for document retrieval and ranking, such as document statistics.
- Any peer in the network needs to employ an integrated public search engine which can run unstructured text queries and return a list of documents.
- Documents are stored locally.
- Anonymity is not supported because the trust computation and feedback should be provided by peers after reviewing documents.
- In the initial system design, assuming replication strategies, storage and communication costs, the ad-hoc nature of document distribution cannot affect the system performance. In fact, all of these properties influence system performance and will be taken into account in further work.
- The proposed trust-aware P2PIR system requires each peer to install a P2P software client locally, which is similar to the file-sharing systems in P2P.

4.1.2 Problems and Contributions

Building a cooperative P2PIR system, which can retrieve relevant *and* trustworthy documents for a given query, involves the following problems:

- Existing relevance-based document ranking algorithms rely on global term statistics. However, current methods to estimate global term statistics may be difficult to employ in real P2P networks (as discussed in Section 2.4). How to design a method which can offer an effective and practical way to estimate global term statistics for relevance-based document ranking algorithms is a challenging problem.

- Since the existing notion of entity trust is insufficient for P2PIR (as discussed in Section 2.4), the problem is how to identify the content trust factors for evaluating the trustworthiness of a document provided by a peer for a given query, and the trustworthiness of a document provider for a given query. Moreover, how to combine the content trust factors into coherent schemes to compute the trust values of documents and document providers.

- Once the above two problems have been addressed, the relevance score and trust value of a document for a given query can be obtained. The next question is how to integrate trust and relevance into an integrated document ranking algorithm.

- P2P networks lack any centralised infrastructure. When employing the proposed trust-aware P2P systems in structured P2P networks, the problem which needs to be addressed is how to determine a mechanism for organising distributed peers into an autonomous and collaborative manner so that relevant and trustworthy information can be collected and computed by any peer in the network.

- There have been no standard metrics and testbeds for evaluating the performance of the proposed trust-aware P2PIR systems in cooperative environments. Therefore, the question of how to develop experimental testbeds, evaluation methodologies and metrics for the evaluation of the proposed trust-aware P2PIR systems needs to be addressed.

This chapter makes four major contributions to address the above problems for trust-aware P2PIR in cooperative environments, which are as follows:

- A method to estimate global term statistics, which is integrated into the traditional K-L algorithm [XC99] to compute the relevance score of a document for a given query (in Section 4.2.2.1).

- A set of content trust factors has been identified to evaluate the trustworthiness value of a document provided by a peer for a given query, and the trustworthiness value of a document provider for a given query. By extending the peer trust model in PeerTrust [XL04], an integrated trust model is designed to combine these factors for calculating the trustworthiness values of a document and a document provider (in Section 4.2.2.2).

- A system architecture is designed to implement the proposed trust-aware P2PIR system in structured P2P networks, which is an extension of the PeerTrust architecture in the context of trust-aware P2PIR (in Section 4.3.1). Moreover, a set of data management protocols is developed by extending the data management protocols of structured P2P networks (in Section 4.3.2).

- A set of testbeds and an evaluation metric are developed to evaluate the performance of the proposed trust-aware P2PIR system in terms of retrieval accuracy, the effectiveness of trust in protecting untrustworthy documents in the top-ranked results list, and scalability of network size (in Section 4.4).

The remainder of this chapter is organised as follows: Section 4.2 describes the proposed trust-aware P2PIR system focusing on document description and ranking. Along the way, the implementation strategies of the proposed trust-aware P2PIR system in structured P2P networks are introduced in Section 4.3, including issues of system architecture and data management protocols. Section 4.4 discusses the testbeds, experimental methodologies and experimental results to demonstrate the performance of the proposed trust-aware P2PIR system on retrieval accuracy, effectiveness of trust and system scalability. This chapter is summarised in Section 4.5.

4.2 Trust-Aware P2PIR in Cooperative P2PIR Environments

As discussed in Chapter 2, there are four problems to be addressed for cooperative P2PIR, which are *document selection criterion, document description, document retrieval and ranking*, and *implementation*. Since the document selection criterion , in this dissertation, is to retrieve *relevant* and *trustworthy* documents for a given query, the remaining three problems will be described in the following three sections: trust-based document description (in Section 4.2.1), document ranking (in Section 4.2.2), the implementation strategies of the proposed trust-aware P2PIR system (in Section 4.3).

4.2.1 Document Description

Document description determines which contents are desirable for presentation in each document. It should provide sufficient information for document ranking algorithms to determine which documents are more likely to satisfy users' requirements. Typically, the problem of document description in distributed networks consists of three sub-problems [Cal00, CC01], which are *discovering and representing what a document contains*, *acquiring the document description*, and *maintaining and updating the document description*. The objective of this subsection is to address the first problem, which is how to represent a document for trust-aware P2PIR. The remaining two problems will be discussed in Section 4.3 because they are related to the implementation strategies of the proposed trust-aware P2PIR system in a cooperative environment.

Document description should provide the document ranking algorithms' required information to determine which documents are relevant *and* trustworthy for a given query in a trust-aware P2PIR system. Document description of trust-aware P2PIR must contain two types of information for the computation of relevance and trustworthiness. In the proposed trust-aware P2PIR system, document description is defined by the tuple $< Con(d_j), Rep(d_j) >$, where $Con(d_j)$ represents the contents of document d_j and $Rep(d_j)$ is the reputation value of document d_j. Both types of information are query-independent, which means that they are only related to the documents themselves. Whatever the queries are, the reputation value and contents of a specific document should be constant. Then, the next question to ask is how to represent the contents $Con(d_j)$ and reputation value $Rep(d_j)$ of document d_j.

Firstly, in order to represent the contents of a document, a number of document description methods have been studied for cooperative P2PIR (in Section 2.2.2). Since full-text based document descriptions can provide much more comprehensive descriptions for text documents than other description forms (e.g., query-driven based or link-based descriptions), and have been extensively applied in existing cooperative P2PIR systems, a full-text based document description is employed to directly represent the document contents $Con(d_j)$ in the proposed trust-aware P2PIR system. This contains statistical information, such as term w_l in document d_j, corresponding term frequency $f(w_l, d_j)$ and document length L_{d_j} in words.

Secondly, a wealth of reputation-based trust management systems in P2P

networks have been developed to evaluate the reputation value of an item. In order to represent the reputation value of a document, the peer trust value in PeerTrust [XL04] is straightforwardly applied to the proposed trust-aware P2PIR system, which is entity trust. The reason PeerTrust is selected for reputation computation is that it is one of the most cited works in the literature of P2P trust management systems [1] . Although PeerTrust is applied to calculate document reputation values in trust-aware P2PIR, other acknowledged reputation-based trust methods (e.g., the reputation-based trust management systems described in Section 2.3.2) can also be used. PeerTrust is reviewed in what follows.

PeerTrust is a reputation-based trust system in structured P2P networks to evaluate the trust value of a peer based on the feedback of past user experiences. To use the PeerTrust approach in document reputation value calculation, $Rep(d_j)$ is defined as a metric which summarises the past evaluations that document d_j has received from users. A high $Rep(d_j)$ value indicates that d_j is regarded as being a reputed document by different users in the network. The reputation value of document d_j is given by

$$Rep(d_j) = \frac{1}{nt(d_j)} \sum_{i=1}^{nt(d_j)} f(p_k)_{(d_j)} * CR(p_k), \tag{4.1}$$

where i is the number of feedback received from peers, $nt(d_j)$ denotes the total amount of feedback for document d_j received from different peers, $f(p_k)_{(d_j)}$ denotes the feedback for document d_j received from peer p_k and $CR(p_k)$ is the credibility value of the participating peer p_k. The reputation value of document d_j is a weighted average of the amount of feedback document d_j receives from different users.

$CR(p_k)$ is the credibility value of peer p_k which summarises the usefulness of the past feedback peer p_k has submitted. The high credibility value of a peer indicates that the feedback provided by the peer is regarded as being reliable feedback. In the proposed trust-aware P2PIR system, let $ncf(p_k)$ and $nf(p_k)$ denote the amount of useful feedback and the total amount of feedback by peer p_k, respectively. The credibility value of peer $CR(p_k)$ is defined by

$$CR(p_k) = \frac{ncf(p_k)}{nf(p_k)}. \tag{4.2}$$

[1] [XL04] is cited by 695 times until June 2010 (source: Google Scholar)

A strategy is proposed to distinguish between useful and useless feedback in the network. Useful feedback can be defined by comparing the document's reputational value in the network with the feedback from a peer. If binary feedback (i.e., 0 is untrusted and 1 is trusted) is used in the proposed trust-aware P2PIR system, the document's reputational value should be between 0 and 1 (i.e., all values from 0 to 1). 0.5 is defined as being the threshold to distinguish trustworthy from untrustworthy documents. If the document's reputational value is equal to or above 0.5 and the peer feedback is 1, then the peer feedback is defined as being useful, otherwise, the feedback is useless. For example, in Figure 3.2, assuming that Peer A totally submitted 4 feedbacks to the network, namely, doc b=1, doc c=0, doc f=0, and doc h=1. In the network, the existing reputational values of the documents are doc b rep=0.6, doc c rep=0.5, doc f rep=0.9 and doc h rep=0.2. When comparing the feedbacks from Peer A with the existing reputational values, only one of the feedback is deemed to be useful. According to Equation 4.2, the credibility value of Peer A is 0.25. Note: since all the peers in structured P2P networks simultaneously act as information providers, users and directory services [LCP+05], the peer's credibility value $CR(p_k)$ can be the information provider's credibility value, as well as the user's credibility value.

4.2.2 Document Ranking

Since both relevance and trustworthiness are the two critical factors of the proposed trust-aware P2PIR, document ranking requires a combination and fusion of these two parallel factors in some way. Fusion for retrieval has been studied for a decade [Lee97]. Early fusion methods combine factors before performing matching, which is not practical [MS05], while late fusion methods perform matching on individual factors and fuse these scores afterwards [MS05]. The proposed trust-aware P2PIR system employs a late fusion method, which computes trustworthiness and relevance scores individually before combining them. An integrated document ranking score should take into account $R(d_j, q_i)$ and $T(d_j, q_i, p_k)$, where $R(d_j, q_i)$ is the relevance score between a document d_j and a query q_i, and $T(d_j, q_i, p_k)$ is the trust value of the document d_j provided by a peer p_k for the given query q_i. A number of fusion methods have been explored, such as normalised-based (i.e., by sum, average, and weight of individual retrieval scores) [Lee97, MS05], evidenced-based combinations [JH97], and probabilistic-based [MS05]. Since the weighted fusion methods (i.e., normalised-based fusion)

is one of the fundamental fusion methods [MS05], the proposed document ranking method initially applies the weighted method. It should be noted that other fusion methods could be investigated in further research. To simplify the problem, the relative weight of relevance and trustworthiness in the ranking algorithm is assumed to be equal in this chapter[2]. Then, the document ranking score in the proposed trust-aware P2PIR system is given by

$$S(d_j, q_i, p_k) = \sqrt{R^2(d_j, q_i) + T^2(d_j, q_i, p_k)}. \tag{4.3}$$

The next question is how to obtain $R(d_j, q_i)$ and $T(d_j, q_i, p_k)$ for a document ranking score computation in the proposed trust-aware P2PIR system. To address this problem, the proposed approaches to compute the relevance-based score $R(d_j, q_i)$ (in Section 4.2.2.1) and the trust-based score $T(d_j, q_i, p_k)$ (in Section 4.2.2.2) of a document for a given query will be described next.

The document ranking consider two, diverse parallel factors, and then requires combining or fusing these two parallel factors in some way. Fusion methods has been research topic for over a dende [Lee97]. Early fusion methods combine feature before performing matching, which is not practical [MS05]. Late fusion methods perform matching on individual features and fuse these scores. In the proposed trust-aware P2PIR system, the late fusion method is used. In order to improve upon the best individual retrieval result.

4.2.2.1 Relevance-Based Document Score Computation

In cooperative P2PIR, to compute the relevance scores between documents and given queries, a number of traditional full-text based document-ranking algorithms have been applied so far (as described in Section 2.2.2). The family of these document ranking algorithms requires global term statistics to compare the importance of terms in the network. As discussed in Section 2.4, existing approaches to estimate global term statistics cannot be effective and practical in real P2P networks which are highly distributed and dynamic. This makes the traditional full-text based document ranking algorithms difficult to employ in real P2P networks. The main contribution of this subsection is to propose an approach to estimate global term statistics in structured P2P networks. Then,

[2]Note, the problem of the relative weight will be studied in Chapter 7.

the estimated global term statistics will be integrated with an existing document ranking algorithm for relevance-based document score computation in the proposed trust-aware P2PIR system.

The proposed estimated global term statistic is called *estimated peer frequency* (EPF), which is a measure of the general importance of a term in the network by utilising the characteristics of structured P2P networks. EPF_{w_l} is defined by

$$EPF_{w_l} = \frac{f(P_{w_l}, E)}{N}, \tag{4.4}$$

where EPF_{w_l} is the estimated peer frequency of term w_l, $f(P_{w_l}, E)$ is the total number of occurrences of P_{w_l} in the network E, P_{w_l} is the peer containing one or more documents with term w_l, and N is the total number of peers in the network E.

In the trust-aware P2PIR system, $f(P_{w_l}, E)$ of Equation 4.4 can be generated during the process of publishing documents to the network. The peers containing w_l publish themselves to the peer responsible for the term w_l. The responsible peers count the number of peers with w_l. When a user sends a query to the network, $f(P_{w_l}, E)$ can be easily collected from the peers responsible for query terms by message routing protocols. The detailed information will be discussed in the implementation section 4.3. To compute N in Equation 4.4, it was observed in [XKY04] that most of the existing DHT schemes in structured P2P networks have a routing table size $O(\log N)$. This means that a network of size N can be estimated by the local routing table size in each peer of a structured P2P network. Then, the estimated size of network N is given by

$$N = x^a, \tag{4.5}$$

Where a denotes the size of the routing table in each peer of the network, x is a pre-defined configuration parameter in a structured P2P network, which indicates how many bits are resolved at each routing step [XKY04].

For example, in Chord [SMK+01], the routing table size in each peer is exactly $\log_2(N)$, while N is the size of a network. The routing table size a can be obtained locally by each peer, and then the network size can be estimated by $N = 2^a$ ($x = 2$). The routing table sizes in Tapestry [ZKJ01] and Pastry [RD01] are similar to that in Chord, except that they use different configuration parameters (Chord

uses 2). This approach can be used in most structured P2P networks except CAN-based ones [RFH+01]. This is because the routing table size of CAN [RFH+01] is independent of the network size.

The idea of EPF_{w_l} is similar to that of global term statistics in traditional document ranking algorithms. The advantages of EPF_{w_l} are: *(i)* the only required information for EPF_{w_l} is $f(P_{w_l}, E)$ and the routing table size. Since there is no extra cost in collecting the routing table size for each peer, computing EPF_{w_l} only requires contacting a few of peers responsible for the terms in a query. This indicates that the cost of obtaining the required information for EPF_{w_l} computation is low; *(ii)* when peers publish new documents to the network, $f(P_{w_l}, E)$ can be updated automatically. The routing table size is kept up-to-date by the structured P2P networks themselves, and is updated frequently. This indicates that EPF_{w_l} can be employed in highly dynamic P2P networks; *(iii)* $f(P_{w_l}, E)$ is shared by the peers responsible for term w_l, and the network size estimated by the routing table size is the same for each peer. This demonstrates that EPF_{w_l} can be constant for the computation of a relevance-based document score by each peer in the network. On the other side, the disadvantages of EPF_{w_l} are: *(i)* the usage range is relatively small, and can only can be available in structured P2P networks except CAN; *(ii)* it is not as accurate as the global term statistics used in traditional IR systems.

Given the above definition for EPF_{w_l}, EPF_{w_l} is used to replace the global term statistics in the K-L retrieval algorithm [XC99] to compute the document relevance score for a given query in the proposed trust-aware P2PIR system, which is given by

$$p(q_i|d_j) = \prod \frac{f(q_i, d_j) + \mu EPF_{q_i}}{|d_j| + \mu}, \tag{4.6}$$

$$R(d_j, q_i) = P(d_j|q_i) \propto p(q_i|d_j), \tag{4.7}$$

where $f(q_i, d_j)$ is the number of occurrences of q_i in d_j, EPF_{q_i} is the estimated peer frequency of q_i appearing in the network, $|d_j|$ is the document length in words and μ is the smoothing parameter in K-L. In Equation 4.7, $R(d_j, q_i)$ is the relevance-base score of a document for a given query used in Equation 4.3.

4.2.2.2 Trust-Based Document Score Computation

Once the relevance score $R(d_j, q_i)$ between document d_j and query q_i in Equation 4.3 has been computed, the next problem is to calculate the trust value $T(d_j, q_i, p_k)$ of a document provided by a peer for a given query. Existing reputation-based trust systems are entity trust and they are insufficient for content trust value computation, as discussed in Section 2.4. This section identifies several content trust factors for evaluating the trustworthiness of a document and a document provider for a given query, and proposes an integrated content trust model to combine these factors.

In general, multiple server peers in a P2P network may have the same document available at the same time, but with a different trust value for each of them. In P2P networks, a client peer can select one or more server peers to review or download a document. In the document trust model of the proposed trust-aware P2PIR system, a peer with a higher trust value is assumed to have a higher likelihood to provide trustworthy documents upon users' requests than a peer with a lower trust value. Therefore, the first content trust factor to take into account is that the document trust value is related to the trust value of the peer providing that document. To quantify and assess the trustworthiness of a document d_j provided by a peer p_k, the document trust value $T(d_j, p_k, q_i)$ in Equation 4.3 can be computed by

$$T(d_j, p_k, q_i) = \frac{Rep(d_j) + \beta T(q_i, p_k)}{1 + \beta}, \tag{4.8}$$

where $Rep(d_j)$ is the reputation value of document d_j, $T(q_i, p_k)$ is the trust value of peer p_k for query q_i, and the parameter β is a positive constant to assign a different weight to $T(q_i, p_k)$. The parameter β will be studied in Section 4.4.3.5.

When users are looking for documents to satisfy their needs, they are always interested in the trustworthiness of relevant documents for a given query, rather than irrelevant documents. It could be argued that a peer containing a few relevant documents with high reputation values could be more useful than a peer containing a number of irrelevant documents with higher reputation values. Therefore, the second content trust factor to take into account is that the peer trust value (i.e., $T(q_i, p_k)$ in Equation 4.8) is related to the reputation values of relevant documents for a given query. A reputation value $Rep(d_j)^{R^+}$ is attributed to each relevant document to the given query q_i in the peer p_k, and $Rep(d_j)^{R^-}$ is

the reputation value of irrelevant document. For a given query q_i, if we know the number of relevant documents r in the peer p_k, then the trust value of the peer p_k to the given query q_i is defined by

$$T(q_i, p_k) = \frac{\frac{1}{r}\sum_{i=1}^{r} Rep(d_j)^{R^+}}{\frac{1}{r}\sum_{i=1}^{r} Rep(d_j)^{R^+} + \frac{1}{s-r}\sum_{i=1}^{s-r} Rep(d_j)^{R^-}} * Rep(p_k), \tag{4.9}$$

where $Rep(p_k)$ is the reputation value of the peer p_k and s is the total number of documents shared by peer p_k.

It should be noted that the trust value of each peer depends on a given query and its relevant documents. In order to eliminate the effect on the peer trust values by the number of relevant and irrelevant documents for any given query, the average reputation values of relevant and irrelevant documents for a given query are calculated with the same weight.

The next task is to compute the reputation value $Rep(p_k)$ of peer p_k in Equation 4.9. In P2P networks, peers can provide a number of documents with different individual reputation values. The higher the percentage of reputed documents a peer can provide, the better reputation the peer should have. Therefore, the third content trust factor to take into account is that the peer reputation value is related to the reputation values of documents it is sharing. Then, the reputation value $Rep(p_k)$ of peer p_k can be defined as being the average of the reputation values of documents that p_k is currently sharing and is given by

$$Rep(p_k) = \frac{1}{s}\sum_{i=1}^{s} Rep(d_j). \tag{4.10}$$

4.2.2.3 Differences Between PeerTrust and Content Trust Model

The proposed content trust model in trust-aware P2PIRs is described in the previous section. This section discusses the differences between PeerTrust [XL04] and the proposed content trust model. PeerTrust is an entity trust model, which can evaluate the reputational value of a peer based on the feedback of past user experience, regardless of users' queries. The proposed content trust model denotes the degree of assessment of the reliability and quality of the relevant document to meet the user's request (as described in Section 1.2.1). In fact, the content trust model is an extension of PeerTrust in the application of P2PIR. The proposed content trust model (i.e., Equations 4.8, 4.9, 4.10) in trust-aware P2PIRs is to compute the trust values of documents and peers for a given query, for

which the reputational values of documents are needed. Therefore, the original PeerTrust (i.e., Equations 4.1 and 4.2) is used to compute the reputational value of documents for the content trust model. The reason PeerTrust is selected to compute the reputational values is that it is one of the most cited works in the literature of P2P trust management systems. Although PeerTrust is applied to calculate documents' reputational values in the content trust model, other acknowledged reputation-based trust approaches can also be employed if the trust model matches two conditions: *(i)* reputation-based trust model; *(ii)* in structured p2p network.

4.3 Implementation Strategies

Since peers in the P2P network are dynamic and self-organise to adjust the network structure, one problem which must be addressed is how to determine a mechanism to organise peers in a cooperative manner. This makes the document contents and reputation information can be collected, and the relevant and trustworthy scores can be computed by any peer in the network. Typical issues on implementing applications in a P2P network include decentralised system architectures and data management protocols. P2P network architectures determine the functionality and responsibility of each peer, as well as data location schemes and message-routing mechanisms. Structured P2P networks are selected to implement the proposed trust-aware P2PIR system in cooperative P2PIR environments. Therefore, design and implementation strategies of the proposed trust-aware P2PIR system should consider the unique characteristics of structured P2P networks (e.g., completely decentralised, routing protocols) and cooperative P2PIR (e.g., information can be accessed without limiting authentication). Although a Chord-based P2P network is used as the base architecture to implement the proposed trust-aware P2PIR system in this dissertation, the system can also be applied to other structured P2P networks.

4.3.1 System Architecture

A structured P2P network is a completely decentralised P2P network overlay, and there is no central server and super peer. Each peer in the network should serve as document providers, users and directory services at the same time. Therefore, the system architecture of each peer in the proposed trust-aware P2PIR system

should be designed to fulfil the following functions: *(i)* storing some document descriptions (e.g., word statistics and reputation data) for directory services; *(ii)* processing and forwarding queries; and *(iii)* ranking retrieved documents for a given query. Figure 4.1 shows the proposed system architecture of the trust-aware P2PIR system which consists of four components, including *Statistics Manager, Reputation Manager, Ranker* and *Data Locator*.

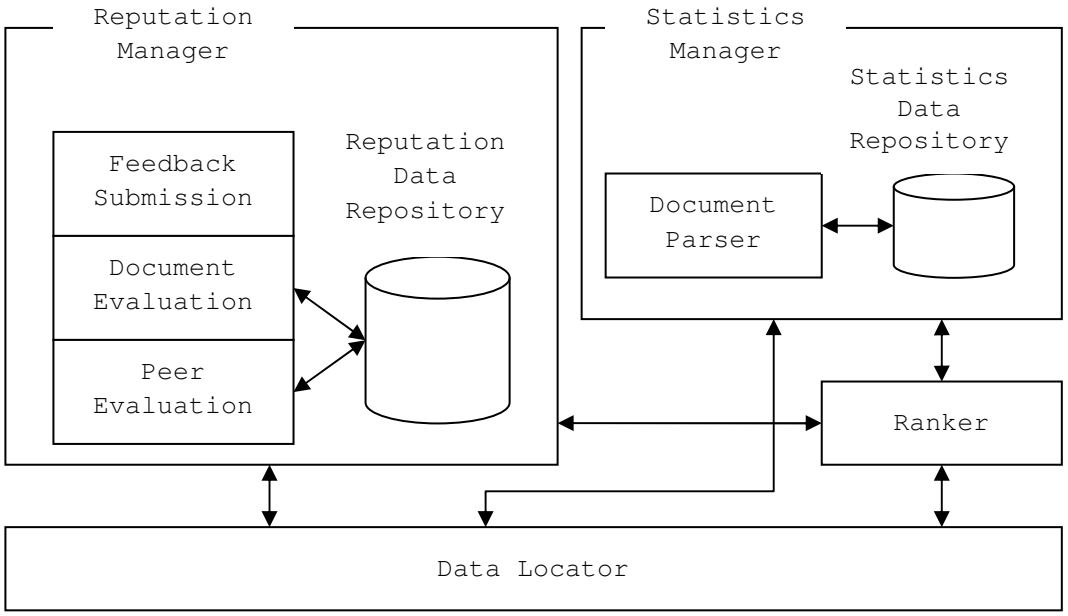


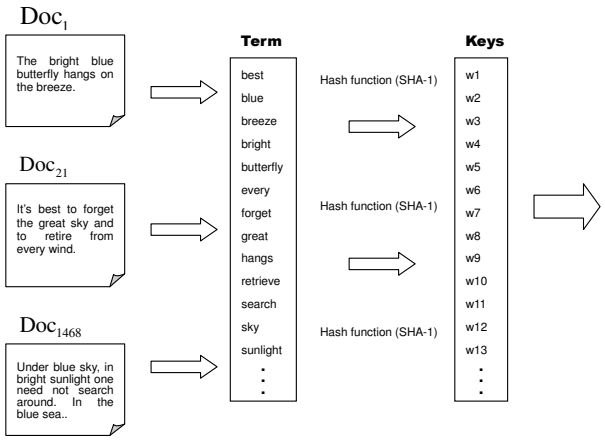
Figure 4.1: System architecture of the proposed trust-aware P2PIR system

The statistics manager consists of two components, namely, *Document Parser* and *Statistics Data Repository*. The document parser is responsible for parsing local documents to extract term statistics such as terms and the corresponding term statistical information. In a structured P2P network, there is a global index and each peer is responsible for storing part of the global index (e.g., keys in the global key space). The statistics data repository is a small database which stores a portion of the global vocabulary and its associated statistics, such as term-id, ID_{w_l}; document-id, *Document List*; term frequency, $f(w_l, d_j)$ and document length, L_{d_j}. In Table 4.1, the ID_{w_l} is a numeric key in the global hash space, which is the hash value of converting the term w_l to a numeric key using an ordinary hash function such as SHA-1 [EJ]. The *Document List* stores a list of document-ids which represent the documents containing the term w_l. The document-id is a unique identifier of document d_j in the network, which is a hash

value of the document name. $f(w_l, d_j)$ is the number of occurrences of term w_l in the document d_j, and L_{d_j} is the document length in words.

Table 4.1 is an inverted index, which is a data structure storing a mapping from words or number to statistics or locations. The proposed trust-aware P2PIR system employs an inverted index as a data structure because the proposed system is designed based on structured P2P networks. An inverted index has been widely used for Information Retrieval in structured P2P networks [PMB06, SYYW03, SLZ+07, PLPZR08, ZRL+07, LLH+03, LKZ+06, GKN09, TXM03, TXD03, LC04b, KJ03, JYF07, LJT07, KWTA07, XSD+08, CJL+09, RV03, NYF08a, NYF08b, Che05, TDX04, TD04, SLZ+09]. DHTs can store an inverted index in structured P2P networks by providing an interface to put and get pairs of *key/-value*. When publishing documents to a network, the *keys* inserted into DHTs are the terms of documents, and the *values* in DHTs are the corresponding posting lists, including term frequency, document name and document location.

Table 4.1: Statistics Data Repository



ID_{w_l}	*Document List*	$f(w_l, d_j)$	L_{d_j}
w_1	Doc_{21}	1	13
w_2	Doc_1	1	8
	Doc_{1468}	2	15
w_3	Doc_1	1	8
w_4	Doc_1	1	8
	Doc_{1468}	1	15
w_5	Doc_1	1	8
w_6	Doc_{21}	1	13
w_7	Doc_{21}	1	13
w_8	Doc_{21}	1	13
w_9	Doc_1	1	8
w_{10}	Doc_{21}	1	13
w_{11}	Doc_{1468}	1	15
w_{12}	Doc_{21}	1	13
	Doc_{1468}	1	15
w_{13}	Doc_{1468}	1	15
⋮	⋮	⋮	⋮

The reputation manager is the adaptation of PeerTrust architecture [XL04] with the extension of a Document Evaluation part. The reputation manager consists of four components, namely, *Feedback Submission, Document Evaluation, Peer Evaluation* and *Reputation Data Repository*. The feedback submission is responsible for submitting users' feedback after using the document. Document evaluation is responsible for evaluating the reputation value of document d_j. The peer evaluation component is to evaluate the reputation value of peer p_k. The reputation data repository stores a portion of the global reputation information

which contains document-id, ID_{d_j}; document reputation value, $Rep(d_j)$; peer-id, *Owners list*; peer reputation value, $Rep(p_k)$; peer location, *Location* and communication port, *Port*, which is shown in Table 4.2. The location of peers represents the current peer IP address.

Table 4.2: Reputation Data Repository

ID_{d_j}	$Rep(d_j)$	*Owners List*	$Rep(p_k)$	*Location*	*Port*
Doc_{21}	0.5	N_{12}	0.6	124.12.35.25	4000
Doc_{10}	0.8	N_{75}	0.6	196.126.1.3	4001
		N_{59}	0.8	139.36.25.1	4000
Doc_{79}	0.1	N_2	0.3	96.12.178.3	4000
Doc_{47}	0.5	N_{35}	0.6	84.13.256.6	4000

The data locator provides a P2P data location scheme for accessing and updating data in the network. Different applications may use different data placement and location schemes, which determine how and where the data can be inserted, updated, and accessed. In the implementation of the proposed trust-aware P2PIR system, the data locator employs Chord routing protocols [SMK+01]. The ranker in the system collects word statistics and reputation information in order to compute the document ranking scores for a given query.

4.3.2 Data Management Protocols

Since the proposed trust-aware P2PIR system is designed to be implemented in Chord structured P2P networks in this dissertation, the data management protocols of the proposed system should extend existing data management protocols of Chord in the context of term statistics and reputation data routing. Therefore, the proposed protocols of the trust-aware P2PIR system consist of the following three phases: *(i) join and publish*; *(ii) lookup and rank*; and *(iii) evaluate and update*.

4.3.2.1 Join and Publish

As a peer publishes its documents to the network for sharing, the statistics manager should parse the sharing documents to extract the statistical information such as terms, term frequencies and document length. Afterwards, the data locator converts the terms and the document names to numeric keys by using the SHA-1 hash function. Two types of messages are issued simultaneously by the

statistics manager and reputation manager through the data locator. The data locator sends the $PUBLISH(ID_{w_l}, (ID_{d_j}, f(w_l, d_j), L_{d_j}))$ messages for each distinct term w_l in document d_j to the peers responsible for the term key ID_{w_l}. The peer responsible for the term w_l adds the ID_{d_j}, $f(w_l, d_j)$, L_{d_j} to the statistics data repository. If the statistics data repository does not contain information of the terms and documents, it adds a new row. In the meantime, the reputation manager issues a $PUBLISH(ID_{d_j}, (Rep(d_j), ID_{p_k}, location))$ message for the document d_j. The data locator forwards the message to the peer responsible for the document reputation value. If the document is available in the network and has already been assigned to a reputation value, then the reputation value of the new published document is equal to the current reputation value. Alternatively, if the document has never been published before, it is assigned the same reputation value of the peer providing the document. If the peer is a new peer in the network, the reputation manager should collect the reputation values of the documents it is sharing and compute its peer reputation value. To collect the reputation values of the documents the peer is sharing, the data locator takes the hash values of document names ID_{d_j} as arguments to retrieve the reputation values of the documents from the network. The lookup protocol is described in the following section.

4.3.2.2 Lookup and Rank

The lookup phase consists of two protocols in terms of $FIND_{doc}^{sta}$ and $FIND_{doc}^{rep}$. The $FIND_{doc}^{sta}$ takes the hash value of term ID_{w_l} as an argument to obtain the list of documents containing the query terms and the corresponding word statistics. The peer responsible for that term returns $< ID_{d_j}, f(w_l, d_j), L_{d_j} >$ triples. Afterwards, the $FIND_{doc}^{rep}$ takes the hash values of document names ID_{d_j} obtained from $FIND_{doc}^{sta}$ as arguments to retrieve the reputation values of documents and corresponding peers. The peer responsible for the reputation values of document d_j and peer p_k returns $< Rep(d_j), ID_{p_k}, Rep(p_k), IP_{p_k}, Port_{p_k} >$. When the ranker receives the response messages of statistics and reputation data for the document d_j, it is able to compute the document scores for ranking.

4.3.2.3 Evaluation and Update

After using documents, users need to evaluate them and leave feedback. If a user thinks the document is trustworthy, the feedback value is set to be 1, otherwise,

0. The evaluation of the document is very subjective, since it depends on users themselves. In order to update the reputation value of document d_j, the data locator sends an updated message $UPDATE(ID_{p_k}, ID_{d_j}, f(p_k)_{(d_j)})$ to the peer responsible for that document reputation value. Then, the document evaluation and peer evaluation components in Figure 4.1 should compute the document reputation value and peer reputation value, respectively. The new reputation values are stored in the reputation data repository.

4.3.3 Limitations of the Proposed Trust-Aware P2PIR

The proposed trust-aware P2PIR system is developed to be implemented in structured P2P networks, which employ DHT to publish and retrieve information. Flooding and random walk are not used in the proposed system because these retrieval mechanisms are developed for unstructured P2P networks (which are randomly-established network overlays) [LCP+05]. The proposed trust-aware P2PIR system employs full-text based document descriptions which provide much more comprehensive descriptions for text documents. However, the size of a full-text based description is significantly large, which requires more storage and involves more network transmission costs. These are both major problems for existing Information Retrieval in structured P2P networks, and have been studied by Li *et al.* [LLH+03] and Yong *et al.* [YDRC06]. To address the problem of storage, a number of approaches have been proposed to compress the size of document descriptions, such as *bloom filter, top-ranked term selection, feature extraction, query history* and *stemming*. These methods are described in Section 2.2.2.1. Moreover, several optimal approaches have been designed to resolve the problem of network transmission costs. These include *content-based network overlay, query expansion, query history* and *document replication policy*. These approaches are discussed in Section 2.2.2.2. Some of the above approaches can be used in the proposed trust-aware P2PIR system to reduce storage costs, such as top-ranked term selection.

4.4 Evaluation

The objective of this section focuses on evaluating the performance of the proposed trust-aware P2PIR system in cooperative environments in terms of *retrieval accuracy*, *effectiveness of trust* and *scalability*. In addition, the parameter β in

the document trust model (i.e., Equation 4.8) will also be studied in this section. The experimental testbeds and evaluation methodologies are described in Sections 4.4.1 and 4.4.2, and the experimental results are discussed in Section 4.4.3.

4.4.1 Trust-Aware P2PIR Testbeds in Cooperative Environments

Since there is no standard testbed available today to evaluate the performance of the proposed trust-aware P2PIR system, a set of testbeds are developed based on one of the public Information Retrieval evaluate corpus TREC W10g [trea]. This is inspired by the P2PIR testbeds in hierarchical P2P networks [LC06] which are designed to evaluate the retrieval accuracy of different search mechanisms for uncooperative P2PIR in hierarchical P2P networks. Compared to the P2PIR testbeds [LC06], the proposed trust-aware P2PIR testbeds in this dissertation have three main differences compared to the P2PIR testbeds [LC06] and these are as follows:

- Trust-aware P2PIR testbeds focus not only on evaluation of the retrieval accuracy of different document ranking algorithms, but also on evaluation of different trust methods to protect untrustworthy documents in the top-ranked results list.
- Each peer in the P2PIR testbeds employs an individual search engine for document indexing and ranking, while peers in the proposed trust-aware P2PIR testbeds apply one public search engine.
- Trust-aware P2PIR testbeds are developed to work on structured P2P networks instead of hierarchical P2P networks which employ super peers to index documents and direct query routing processes.

The proposed trust-aware P2PIR testbeds use the same TREC WT10g collection as the testbeds of Lu *et al.* [LC06] and Klampanos *et al.* [KPJD05], but with different setups, for the following reasons:

- Each peer in a structured P2P network is assumed to provide equal capabilities [LCP+05, SMK+01, RFH+01, RD01, MM02, ACMD+03, MBR03, ZKJ01], and the proposed trust-aware P2PIR system is developed based

on structured P2P networks. The proposed testbeds initially assume that peers act as equals and provide a similar number of documents, while the number of documents stored in the peers of testbeds of Lu and Klampanos are significantly different because their approaches require super peers and leaf peers [LC06, KPJD05].

- Both of their approaches are content-based network overlays. They consider the document content distribution in a network, which is not necessary in the proposed trust-aware P2PIR system.

- The proposed trust-aware P2PIR testbeds are developed for working in two scenarios, namely, cooperative P2PIR (e.g., P2P Web search) and uncooperative P2PIR (e.g., P2P digital library search). In uncooperative P2PIR, each simulated digital library is expected to have a number of documents for sharing. However, both works define each peer as a Web domain of a TREC WT10g collection, in which some of the Web domains may contain thousands of documents and some may contain less than 10 documents. This is not realistic for a digital library simulation.

Having considered the above three reasons, their setups are not used in the proposed trust-aware P2PIR testbeds. However, Klampanos *et al.* [KPJD05] identify three properties of realistic P2PIR testbeds, two of which can be employed in further work of the proposed testbeds. These three properties are: *(i)* a peer shares a limited number of topics; *(ii)* documents are distributed in a power-law pattern; *(iii)* and content replication [KPJD05]. Since the proposed trust-aware testbeds do not consider document topic distribution, the remaining two properties can be integrated with the proposed testbeds by Klampanos's methodologies [KPJD05] in the further research.

The methodology of how to generate the testbeds to evaluate the performance of the proposed trust-aware P2PIR system in cooperative environments is described in the following sections. This includes the contents of testbeds, query set, relevant and trustworthy judgement files, and experimental settings.

4.4.1.1 Contents of Testbeds

For the purposes of the experiments, three small-sized testbeds with 100, 200 and 400 peers, and one medium-sized testbed with 1000 peers are generated. When

simulating a large-sized testbed (e.g., 10,000-level peers), a large document corpus is needed, such as TREC .GOV2 [treb]. For example, Lu [LC07b] simulated a medium-sized P2PIR testbed of 2,500 peers by TREC W10g [trea], and a large-sized P2PIR testbed of 25,000 peers by TREC .GOV2. Since the existing dataset we have is TREC WT10g, small and medium-sized testbeds are preferred rather than large ones. TREC WT10g is a 10 gigabyte corpus which contains 1 692 096 English Web documents used for the evaluation of Information Retrieval systems. TREC WT10g was originally divided into 5,157 collections, and; 100, 200, 400, 1000 collections were randomly selected from them. Each of the collections is defined as a document provider (i.e., peer) in the network. The statistics of four different testbeds are shown in Table 4.3.

Table 4.3: Statistics of Four Different Sized Testbeds

Testbed Name	Num Docs	AVE DocLen	Num Terms	Num Unique Terms
100 Peers:	49881	417	20842114	334435
200 Peers:	108270	385	41730312	568393
400 Peers:	216307	393	85221594	901881
1000 Peers:	399916	438	175328732	1913108

In Table 4.3, **Num Docs** is the total number of documents, **AVE DocLen** is the average length of the documents, **Num Terms** is the overall number of terms, and **Num Unique Terms** is the number of terms without overlap.

4.4.1.2 Query Set

For the TREC WT10g corpus, the standard query set is TREC topics 451-550 [trec], which are provided by the US National Institute for Standards and Technology (NIST). Basically, TREC topics consist of the three fields of $< title >$, $< description >$ and $< narrative >$. According to the study of users' query behaviour in P2P networks, it has been observed that the average query length for text retrieval is 2.2 words [NJYF07]. Therefore, the $< title >$ field in each TREC topics 451- 550 is selected as the query set in this experiments because the average length of $< title >$ field is 2, which is close to the average query length in real P2P networks. The query set is called the TREC 451-550 short queries (or the TREC 451-550 short query set) in this dissertation. The query set has been filtered by the stop word list[3] and stemmed by the Porter algorithm [Por97]. Examples of the query set are shown in Table 4.4.

[3]http://www.lextek.com/manuals/onix/stopwords1.html

Table 4.4: Examples of the TREC 451-550 short queries

TREC 451-550 Short Queries	
Query Number	**Contents**
TREC 451	bengal cat
TREC 455	when did Jacki Robinson appear game
TREC 456	world go end 2000
TREC 524	eras scar

4.4.1.3 Relevant and Trustworthy Judgements

Typically, when measuring the retrieval accuracy of an IR system, relevant judgement assessments are needed to verify whether or not a retrieved document is relevant. NIST provides standard judgement assessments for TREC WT10g and TREC topic 451-550 [4]. The relevant assessments provided by NIST are generated by a group of experts from different areas. The experts define a document as being relevant if any piece of the document is relevant to a given query. The relevant judgement assessments employ binary judgement, in which the document is either relevant (marked 1) or irrelevant (marked 0). Since there are no trustworthy judgement assessments available for any of the current public corpus (e.g., TREC collections), pseudo-trustworthy document judgement assessments are generated. The trustworthiness of a document is pre-defined as being that, if the documents are provided by a malicious peer, they are set to be untrustworthy (marked 1). On the contrary, documents provided by a good peer are set to be trustworthy (marked 0). Different trustworthy judgement assessments are individually generated for the four trust-aware P2PIR testbeds, depending on the experimental settings.

4.4.1.4 Experimental Settings

The experimental settings for the evaluation of the performance of the proposed trust-aware P2PIR system are inspired by PeerTrust [XL04]. Currently, four types of peers in real P2P networks can be distinguished, namely, good peers, malicious peers, front peers and selfish peers, which have been described in Chapter 2. Since the proposed content trust model in trust-aware P2PIR system is developed based on PeerTrust, which is a reputation-based trust management system to filter feedback from malicious peers, two types of peers are simulated

[4] http://trec.nist.gov/data/qrels-eng/index.html

in the experiments, namely, good peers and malicious peers. The percentage of malicious peers is initially set to 20%. Good peers provide positive feedback to trustworthy documents and negative feedback to untrustworthy documents. On the other hand, malicious peers submit positive feedback for malicious peers, and negative feedback for good peers. The credibility values (as described in Section 4.2.1) of good peers and malicious peers are randomly set to 90% and 20%, respectively. The number of feedback entries for each document is initially set to 10. A number of experimental settings could change for the evaluations, for example, the percentage of malicious peers in the network, the credibility of good peers and malicious peers, the percentage of untrustworthy documents in a good peer, and the percentage of trustworthy documents in a malicious peer. For the initial evaluation of retrieval accuracy, effectiveness of trust, and scalability of the proposed trust-aware P2PIR system in cooperative environments, the current settings are considered to be sufficient.

4.4.2 Evaluation Methodologies

The performance of the proposed trust-aware P2PIR system in cooperative environments is measured by retrieval accuracy, effectiveness of trust, and scalability of network size.

When measuring the retrieval accuracy of the proposed trust-aware P2PIR system, it is first necessary to introduce two fundamental evaluation metrics in Information Retrieval, which are *recall* and *precision* [BYRN99]. Recall is the fraction of the relevant documents which has been retrieved. Precision is the relevant fraction of the retrieved documents. Recall and precision are set-based measures, which are normally used to evaluate the unordered sets of retrieved documents for retrieval algorithms. To evaluate the ranked-based results set which are usually given by the top-k retrieved documents, standard rank-based measurements for traditional centralised IR, distributed IR and P2PIR are applied, which are *average precision at given document cut-off values* and *11-point interpolated average precision versus recall* [BYRN99, NF04, LC07b]. 11-point interpolated average precision versus recall focuses on evaluating the overall retrieval accuracy of retrieval algorithms. Compared with 11-point interpolated average precision versus recall, average precision at given document cut-off values is more closely correlated with user satisfaction [BV04]. In this dissertation, both metrics are applied to evaluate the retrieval accuracy of the proposed trust-aware P2PIR

system in cooperative environments.

In order to evaluate the effectiveness of the proposed document trust model in trust-aware P2PIR system, a new ranked-based evaluation metric is developed. Since in IR systems, users are always concerned with the top-ranked documents rather than the whole results set [BV04], the traditional IR ranked-based metric *average precision at the given document cut-off values* is modified to an *average percentage of untrustworthy documents at the given document cut-off values* for the experimental purposes. The metric computes the average percentage of untrustworthy documents over a set of queries where the top-ranked documents have been seen for each query. Since cut-off values such as 5, 10, 15, 20, and 30 have been used extensively in the Information Retrieval literature [BYRN99], these values are chosen in this experiments. The formalised definition of the average percentage of untrustworthy documents at the given document cut-off values is given by

$$\overline{Per}(k) = \sum_{i=1}^{N_q} \frac{Per_i(k)}{N_q}. \tag{4.11}$$

where $\overline{Per}(k)$ is the average percentage of untrustworthy documents when the top-k documents in the results set have been seen. N_q is the number of queries used and $Per_i(k)$ is the percentage of untrustworthy documents at the top-k documents for the $i-$th query.

The scalability of the proposed trust-aware P2PIR system is evaluated by the effect of network size on both retrieval accuracy and the effectiveness of trust. The retrieval accuracy and effectiveness of trust are assessed by the evaluation metrics described above.

4.4.3 Experimental Results

This section focuses on evaluating the performance of the proposed trust-aware P2PIR system in cooperative environments to demonstrate that: *(i)* estimated global term statistics integrated with the K-L retrieval algorithm can achieve acceptable retrieval accuracy compared to the existing document retrieval algorithms with accurate global term statistics; *(ii)* the proposed document trust model can provide a better combination of retrieval accuracy and the effectiveness of trust than several existing entity trust models; and *(iii)* the system can

be scaled to large-sized networks. Five sections are devoted to the experimental results with regard to evaluating: *(i)* the retrieval accuracy of the proposed estimated global term statistics; *(ii)* the effectiveness of different trust models in protecting untrustworthy documents on the top-ranked results list; *(iii)* the effect of different trust models on retrieval accuracy; *(iv)* system scalability; and *(v)* the study of the parameter β in the document trust model. To be specific, this subsection begins with the individual experimental descriptions and continues to present the different experimental results.

- ***Evaluation of Retrieval Accuracy***
  - *[Experiment 1 in Section 4.4.3.1]*: this experiment evaluates the retrieval accuracy of the proposed estimated global term statistics, which is integrated with K-L for relevance-based document score computation in the proposed trust-aware P2PIR system. Accurate global term statistics with K-L and VSM document ranking algorithms are compared. This experiment does not take into account trust metrics to shield the evaluation of the retrieval algorithm from the factors which may affect retrieval accuracy.
- ***Evaluation of the Effectiveness of Trust***
  - *[Experiment 2 in Section 4.4.3.2]*: the objectives of this experiment are: *(i)* to explore the effectiveness of the proposed trust-aware P2PIR system (as described in Section 4.2.2); and *(ii)* to study the effectiveness of the proposed document trust model alone (as described in Section 4.2.2.2) in protecting untrustworthy documents in the top-ranked results list. Firstly, to study the effectiveness of the proposed trust-aware P2PIR system, an existing P2PIR approach (e.g., the vector space model (VSM) in P2PIR [LKZ+06, LJT07, KJ03, CAN02, LLQ+04]) is compared with the proposed trust-aware P2PIR system by the percentage of untrustworthy documents retrieved in the top-ranked results list. Secondly, for the purpose of studying the effectiveness of the proposed document trust models, several of the existing most-cited reputation-based peer and file trust models have been selected for comparison. In order to ignore the effect of relevance-based retrieval algorithms on the evaluation of the effectiveness of different

trust models, relevance-based retrieval algorithms are not taken into account.

  - *[Experiment 3 in Section 4.4.3.3]*: this experiment explores the effect of different trust models on retrieval accuracy. When combining trust metrics to rank documents, retrieval accuracy should be sacrificed because the relevant but untrustworthy documents are removed from the top-ranked results list. Moreover, existing trust models only produce entity trust, and do not take relationships between queries and contents into account, so the irrelevant but trustworthy documents gain higher ranks. This can decrease retrieval accuracy. This experiment studies the percentage of the degrading of retrieval accuracy yielded by different trust models. In order to focus on the comparison of retrieval accuracy reduced by different trust models, the relevance-based retrieval algorithm (as described Section in 4.2.2.1) is integrated with different trust models.

- ***Evaluation of Scalability***

  - *[Experiment 4 in Section 4.4.3.4]*: this experiment evaluates the scalability of the proposed trust-aware P2PIR system in terms of retrieval accuracy and the effectiveness of trust. This experiment employs four testbeds, namely, 100, 200, 400, and 1000-peer ones. The experimental results demonstrate the performance on both retrieval accuracy and the effectiveness of protecting untrustworthy documents in different testbeds.

- ***Parameter β Study***

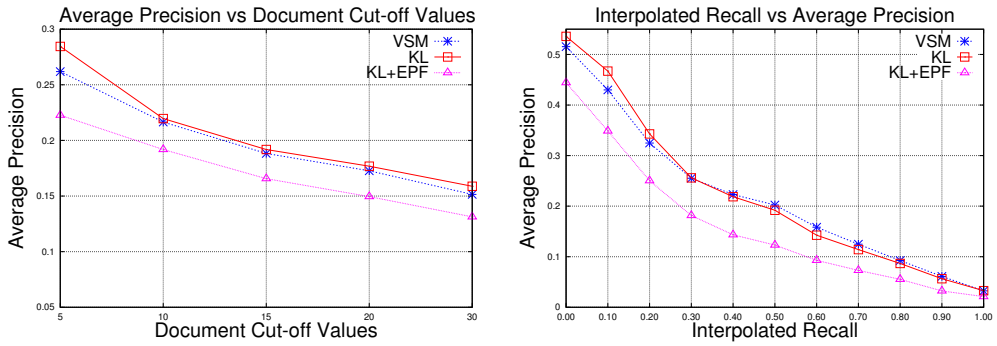
  - *[Experiment 5 in Section 4.4.3.5]*: the parameter β in Equation 4.8 is the weight between peer trust and document reputation. This experiment studies the effect of retrieval accuracy and the effectiveness of trust for the document trust model when changing the parameter β.

According to the literature of cooperative P2PIR in Chapter 2, the most widely-used relevance-based document retrieval algorithm is the vector space model (as described in Section 2.2.2.2). Therefore, it is better to set the vector space model as the baseline to study the retrieval accuracy of different approaches,

with estimated global term statistics and accurate global term statistics. To study the effectiveness of the proposed trust-aware P2PIR system, several existing peer and file trust models need to be compared with the proposed document trust model. Since the document trust model in the proposed trust-aware P2PIR system is reputation-based and implemented in structured P2P networks, the policy-based trust models and the trust management systems which require global trust values and pre-trusted peers are not applicable and comparable. Having considered these conditions, three reputation-based trust models are selected and implemented in the experiments, namely, PeerTrust [XL04], PowerTrust [ZH07] and eBay [eBa]. PeerTrust and PowerTrust are two of the most cited trust management systems in P2P networks, and eBay is one of the most widely used reputation-based trust systems on the current Internet. Moreover, the eBay trust mechanism is the basis of most of reputation-based trust management systems in P2P networks and Information Retrieval.

4.4.3.1 Retrieval Accuracy

The experimental results in this section demonstrate the retrieval accuracy of different relevance-based document retrieval methods, which are the vector space model (VSM), the K-L retrieval algorithm with accurate global term statistics (KL), and the K-L retrieval algorithm with estimated global term statistics (KL+EPF).



(a) average precision vs document cut-off (b) average precision vs interpolated recall values

Figure 4.2: Retrieval accuracy of different relevance-based document retrieval algorithms for the TREC 451-550 short query set in the 1000 peer-sized network.

Figure 4.2 displays the experimental results for the TREC short query set using different relevance-based ranking algorithms in the 1000 peer-sized network.

Figures 4.2 (a)-(b) depicts the results of two evaluation metrics, which are (a) the average precision vs. document cut-off values, and (b) 11-point interpolated recall vs. average precision. The higher the average precision in both figures, the better the retrieval accuracy the retrieval algorithm can achieve. In Figure 4.2 (a), the K-L with accurate global term statistics provides the better retrieval accuracy than the others. Figure 4.2 (b) shows that the overall retrieval accuracy of K-L and the VSM is close, and that both of them achieve the better retrieval accuracy than KL+EPF. The two plots in Figure 4.2 demonstrate that KL+EPF cannot provide a competitive retrieval performance to the algorithms with accurate global statistics on both the overall retrieval accuracy and user satisfaction. The reasons for this are: *(i)* EPF is an estimated term statistic; *(ii)* the original global term statistic in K-L is a document level statistic and EPF is a peer level statistic. The original can provide more accurate descriptions of terms' importance in the network than EPF. For example, terms A and B have the same EPF, 105, which means that they both appear in 105 peers of the 1000-sized network, but their global document frequency could be very different; A may be 1450 and B could be 250. The performance of KL+EPF is worse than those of the algorithms with accurate global term statistics, but is still comparable. Moreover, by considering the limitations of the existing methods to estimate global term statistics, EPF may provide an acceptable retrieval performance and a practical solution to real P2P networks, which are highly dynamic and distributed.

4.4.3.2 Effectiveness of Trust

The objective of the experiment explores the effectiveness of the proposed trust-aware P2PIR system. Figure 4.3 shows the percentage of untrustworthy documents appearing on the top-ranked results list by different methods, such as VSM [LKZ+06], the proposed trust-aware P2PIR system (as described in Section 4.2), document trust model alone of the proposed trust-aware P2PIR system (as described in Section 4.2.2.2), PeerTrust [XL04], PowerTrust [ZH07] and eBay [eBa]. Four of them are always zero, which are document trust model alone of the proposed trust-aware P2PIR system, PeerTrust, PowerTrust and eBay.

To study the effectiveness of the proposed trust-aware P2PIR system, an existing P2PIR approach is needed for comparison. Since VSM is widely used in cooperative P2PIR, it is selected for comparison with the proposed trust-aware P2PIR system. In the experiment, documents are ranked by scores computed

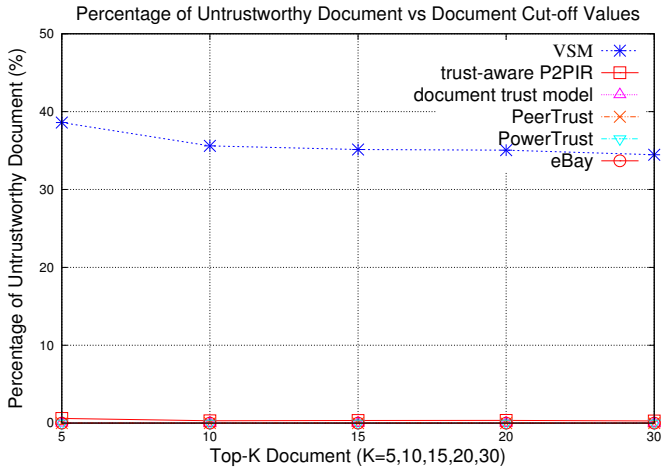


Figure 4.3: Effectiveness of different methods in protecting untrustworthy documents appearing on the top-ranked result list for the TREC 451-550 short query set in the 1000 peer-sized network.

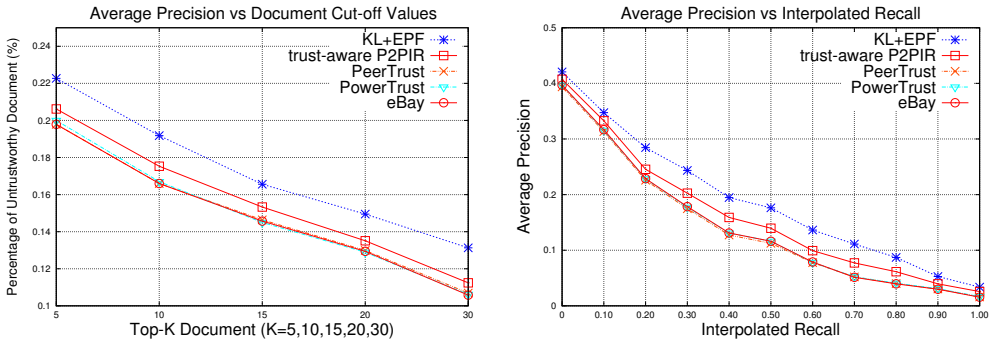
by VSM and the trust-based document ranking algorithm (i.e., Equation 4.3). Some observations can be made from Figure 4.3, the first of which is that the percentages of untrustworthy documents range from 30% to 40% yielded by VSM, indicating that trustworthiness is a critical factor for cooperative P2PIR. Without trust metrics, users run a higher risk of reviewing or downloading untrustworthy documents, even if they are relevant for a given query. Secondly, the percentage of untrustworthy documents stays from 0.3% to 0.6% produced by the proposed trust-aware P2PIR system, which demonstrates that the proposed trust-aware P2PIR system can significantly reduce the possibility of untrustworthy documents appearing in the top-ranked results lists. In other words, cooperative P2PIR can achieve great benefits from the proposed trust-aware P2PIR system, which can effectively filter untrustworthy documents in the top-ranked results list.

In order to focus on the comparison of the effectiveness of different trust models to filter untrustworthy documents, documents are ranked by scores calculated by different trust models only, such as the document trust model alone of the proposed trust-aware P2PIR system (i.e., Equation 4.8), PeerTrust, PowerTrust and eBay. The relevance-based retrieval algorithms are ignored in this experiment. The experimental results are 0 for all of them, demonstrating that the proposed document trust model can be as effective as the existing entity trust models in

completely filtering untrustworthy documents in the top-30 documents. It is believed that the proposed document trust model should sacrifice the effectiveness of trust in the whole ranked results list. This is because trustworthy but irrelevant documents obtain lower scores than those yielded by existing entity trust models. There should be some differences in the lower ranks (e.g., top-10000 documents). However, since top-30 documents are widely studied in the IR literature, the proposed document trust model is believed to be effective for the top-30 documents to filter untrustworthy documents.

In summary, the experimental results suggest that cooperative P2PIR can significantly benefit from the proposed trust-aware P2PIR system. Moreover, the proposed document trust model is as effective as the existing entity trust models to filter untrustworthy documents in the top-30 results list.

4.4.3.3 Effect of Different Trust Models on Retrieval Accuracy



(a) average precision vs document cut-off (b) average precision vs interpolated recall values

Figure 4.4: Retrieval accuracy of different trust models for the TREC 451-550 short query set in the 1000 peer-sized network.

In the previous experiment, the effectiveness of different trust models in protecting untrustworthy documents in the top-ranked results list was explored. In this experiment, the effect of different trust models on retrieval accuracy is studied in a 1000-peer network. KL+EPF (as described in Section 4.2.2.1) is selected as the baseline for comparison. The two retrieval accuracy metrics are applied in this experiment.

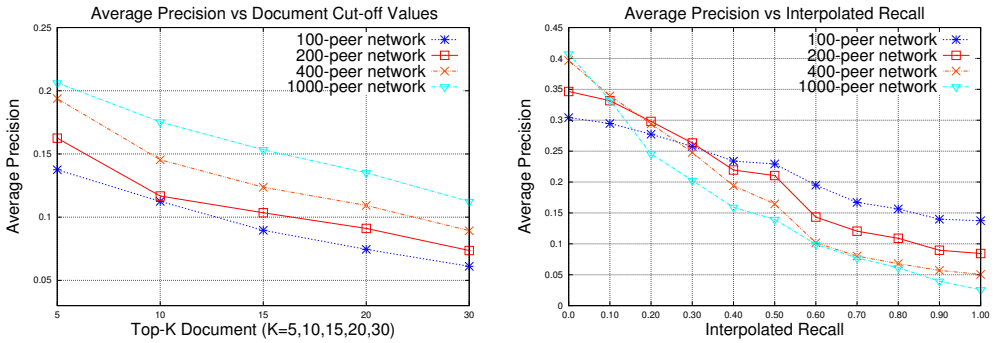
Figure 4.4 shows the experimental results for the TREC short query set using different trust models in a 1000-peer network. Figures 4.4 (a)-(b) plots the results

of the two evaluation metrics, namely, (a) the average precision vs document cut-off values, and (b) 11-point interpolated recall vs average precision. As expected, KL+EPF can achieve the best retrieval accuracy because KL+EPF ranks both trustworthy and untrustworthy documents in the top-ranked results list for a given query. Compared with PowerTrust, PeerTrust and eBay, the proposed trust-aware P2PIR system is able to outperform other entity trust models on both evaluation metrics. The reason for this is that existing trust models only consider entity trust, and do not take contents and given queries into account. This means that trustworthy documents get high scores and ranks whether they are relevant or not. Therefore, some trustworthy but irrelevant documents can be ranked in the top-ranked results list, which can reduce the retrieval accuracy in terms of both user satisfaction (in Figure 4.4 (a)) and the overall retrieval performance (in Figure 4.4 (b)). This phenomenon shows that: *(i)* relationships between document contents and queries are essential factors for calculating trust scores for P2PIR; *(ii)* the proposed trust-aware P2PIR system can achieve a better retrieval performance than existing entity trust models. The retrieval accuracy of PeerTrust, PowerTrust and eBay is too close, with little differences, as shown in Figure 4.4, which indicates that the different existing trust models are not sensitive in P2PIR even if they have more differences in their evaluations of peer and file trust management systems [ZH07]. The reason for this is believed to be the experimental settings, the weights of different trust models in the final scores of documents, the percentage of malicious peers in the network, and the credibility of good peers and malicious peers. Since the experimental results have shown that the proposed trust-aware P2PIR system has less sacrifice on retrieval accuracy than current trust models, the subtle differences between the retrieval accuracy of each existing trust model are of no interest.

In summary, combining the conclusion of the previous experiment, the proposed trust-aware P2PIR system could provide the better retrieval accuracy without sacrificing the effectiveness of trust in protecting untrustworthy documents in the top-ranked results list.

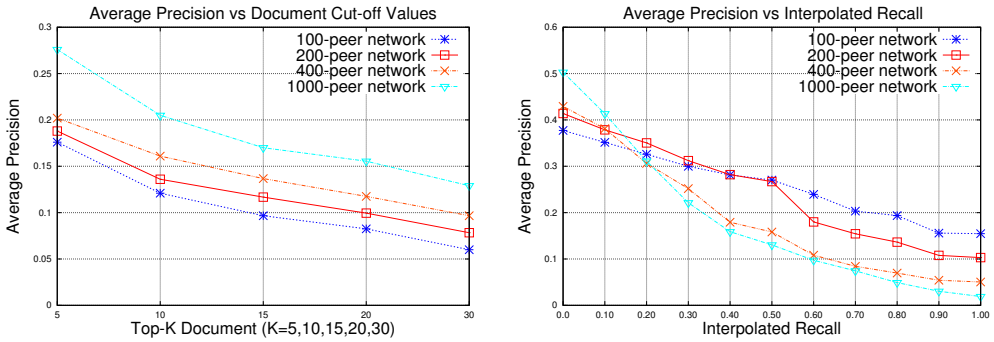
4.4.3.4 Scalability of Network Size

This set of experiments studies the scalability of the proposed trust-aware P2PIR system on retrieval accuracy and the effectiveness of trust in different sized networks. Four testbeds are employed in the experiments, including three small-sized



(a) average precision vs document cut-off (b) average precision vs interpolated recall values

Figure 4.5: Retrieval accuracy of the proposed trust-aware P2PIR system for the TREC 451-550 short query set in different sized networks.



(a) average precision vs document cut-off (b) average precision vs interpolated recall values

Figure 4.6: Retrieval accuracy of K-L with accurate global term statistics for the TREC 451-550 short query set in different sized networks.

testbeds and a medium-sized one. Figure 4.5 shows the experimental results of the retrieval accuracy for the TREC short query set in different sized networks. This displays the results of the two evaluation metrics, which are: (a) the average precision vs document cut-off values, and (b) 11-point interpolated recall vs average precision. It can be seen from Figure 4.5 (a) that the larger the network is, the better retrieval accuracy the system can provide, which indicates that the proposed trust-aware P2PIR system can better scale large-scale networks for retrieval accuracy on user satisfaction. However, the overall retrieval accuracy decreased when the network was scaled to larger networks, as shown in Figure 4.5 (b), indicating that the proposed trust-aware P2PIR system favours user satisfaction more than the overall retrieval performance. The reason for this is that the K-L algorithm with accurate global term statistics favours user satisfaction

more, which is shown in Figure 4.6. Although the proposed estimated global term statistics and document trust model in the proposed trust-aware P2PIR system do make some shape changes between Figures 4.5 and 4.6 (a)-(b), in general, the curves in Figure 4.5 (a)-(b) and Figure 4.6 (a)-(b) are very similar. This suggests that the scalability of the proposed trust-aware P2PIR system on retrieval accuracy mainly depends on the scalability of K-L. While K-L has been shown to be scalable to large-scale networks in traditional Information Retrieval, the proposed trust-aware P2PIR system in cooperative environments can also provide scalability on retrieval accuracy in large-scale networks.

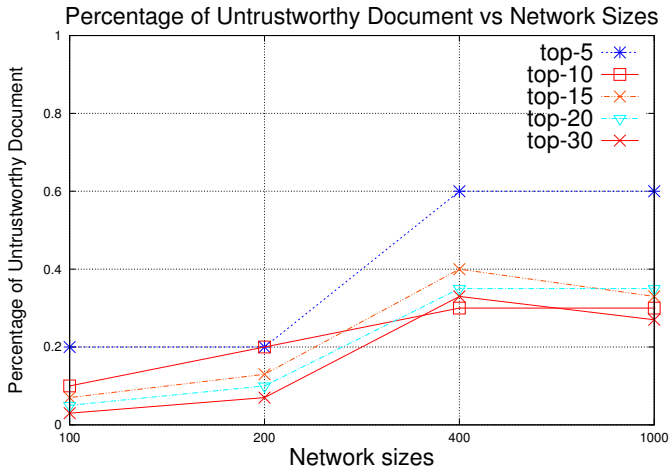
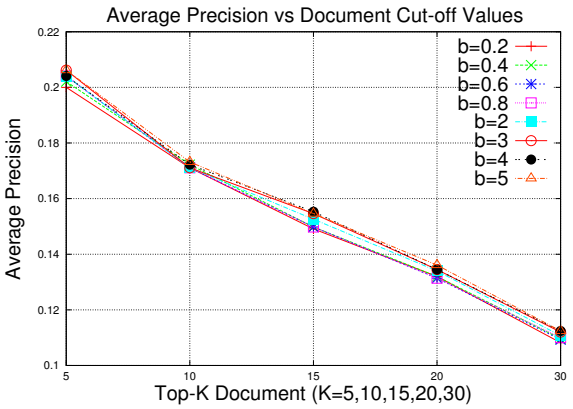
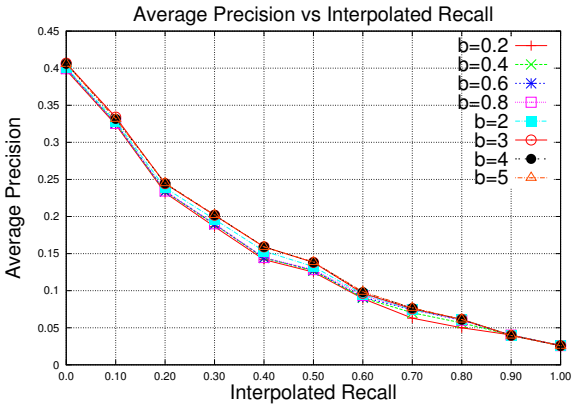


Figure 4.7: Percentage of untrustworthy documents in the top-ranked results list for the TREC 451-550 short query set in different sized networks.

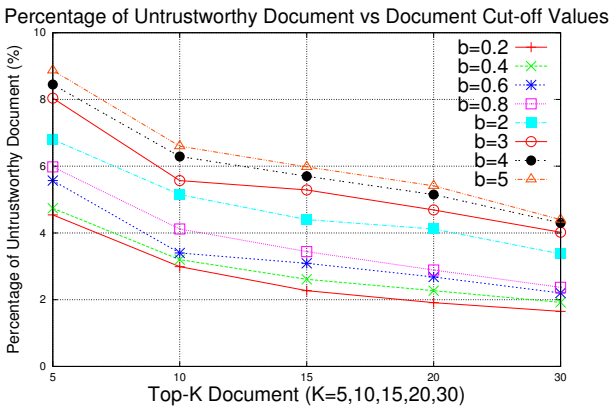
In Figure 4.7, the percentage of untrustworthy documents in the top-ranked results lists increased between the 100-peer testbed and the 400-peer testbed. When the network size was scaled to 1000 peers, the percentages of untrustworthy documents stayed at the same level as in the 400-peer testbed. This is a good sign, showing that, when the network is scaled from small to medium size, it does not increase the risk of reviewing and downloading untrustworthy documents. Moreover, the percentages in all the situations of Figure 4.7 are at a relatively low level (from 0.03% to 0.6%), compared to existing P2PIR approaches (e.g., VSM in Figure 4.3). This indicates that the proposed trust-aware P2PIR system is effective in protecting untrustworthy documents in the top-ranked results list when applied to large situations.



(a) average precision vs document cut-off values



(b) interpolated recall vs average precision



(c) percentage of untrustworthy documents on the top-ranked result list

Figure 4.8: Study of the parameter β in the proposed document trust metrics for the TREC 451-550 short query set in 1000 peer-sized network.

4.4.3.5 Parameter β Study

In the proposed trust-aware P2PIR system, there is one weight β between peer trust and document reputation in Equation 4.8. To study the effect of the document trust model on both retrieval accuracy and the effectiveness of trust with respect to β, a set of experiments is conducted. The range of β value is set to from 0.2 to 5 in the experiments. The selected weight range is that either document reputation is more important ($\beta < 1$) for Equation 4.8, or peer trust takes more weight ($\beta > 1$). It can be seen from Figures 4.8 (a)-(b), that the retrieval accuracy is close when changing β, which demonstrates that the retrieval accuracy is not sensitive on β values. For the study of the effectiveness of trust, the better the effectiveness, the lower the β is, indicating that, when β becomes lower, the document trust value is increased. This suggests that the document reputation value is more important than the peer trust value for document trust value computation. In the end, it is better to select a lower β value which can greatly improve the effectiveness of trust without degrading the retrieval accuracy very much.

4.5 Summary

This chapter proposed a trust-aware P2PIR system for cooperative P2PIR environments. A method is proposed to estimate global term statistics integrated with K-L to compute relevance-based document scores for the proposed trust-aware P2PIR system. This approach can facilitate effective and practical Information Retrieval in real P2P environments which are highly dynamic and distributed. Moreover, a set of content trust factors is identified in the context of P2PIR, and trust models are proposed to calculate the trustworthiness value of a document or document provider for a given query. A system architecture and data management protocols are designed to implement the proposed trust-aware P2PIR system in structured P2P networks, which is an extension of the PeerTrust architecture and data management protocols of structured P2P networks in the context of trust-aware P2PIR in cooperative P2PIR environments. A set of testbeds is developed to evaluate the performance of the proposed trust-aware P2PIR system on retrieval accuracy, effectiveness of trust, and scalability of network size.

Chapter 5

Trust-Aware P2PIR in Uncooperative P2PIR Environments

5.1 Introduction

The previous chapter addressed the problem of trust-aware P2PIR in cooperative environments. The objective of this chapter is to design a trust-aware P2PIR system which can find *relevant* and *trustworthy* documents for a given query in uncooperative P2PIR environments. Unlike cooperative P2PIR environments, document providers (i.e., peers) in uncooperative P2PIR environments may not provide document descriptions due to proprietary or financial cost issues, where access is limited. Instead, they may provide descriptions of themselves (i.e., document provider descriptions), or may not provide any descriptive information at all. In addition, a document provider may employ an individual search engine which can run unstructured text queries. In uncooperative P2PIR, detailed techniques of search engines may not be public, such as ranking algorithms and stemming algorithms.

5.1.1 Assumptions

Before designing the proposed trust-aware P2PIR system in uncooperative P2PIRs, the following assumptions must be made (most of these are the same as the contents of Section 4.1.1):

- Document providers (i.e., peers) do not provide any description of their documents. Instead, they may or may not provide descriptions of themselves.
- Users cannot directly access document copies or statistics due to copyright issues or access limitations.
- Each document provider employs an individual search engine which can run unstructured text queries and provide rankings of the retrieved documents with relevance scores.
- Documents are stored locally and each document provider (i.e., peer) has the ownership of its documents.
- Anonymity is not supported and feedback should be provided by peers after reviewing documents.
- In the initial system design, assuming replication strategies, storage and communication costs, ad-hoc nature of document distribution cannot affect system performance.
- The proposed trust-aware P2PIR system requires each peer to install a P2P software client.

5.1.2 Problems and Contributions

Three problems have been identified in terms of traditional Information Retrieval in uncooperative environments [Cal00]: *resource description, resource selection* and *result merging*. Resource description is responsible for representing the contents of a text database. Resource selection is responsible for selecting a set of text databases to search based on resource descriptions which are relevant to a given query. Result merging is responsible for integrating and re-ranking the results retrieved from the selected text databases into a coherent list [Cal00]. For trust-aware P2PIR in uncooperative environments, these problems are extended because: *(i)* the selection criteria of documents and document providers are not only *relevance* but also *trustworthiness*; *(ii)* the strategy to implement trust-aware P2PIR systems in structured P2P networks. Therefore, trust-aware P2PIR in uncooperative P2PIR involves the following problems:

- How to design an algorithm which can offer an effective and practical way to select relevant and trustworthy peers to search for a given query in uncooperative P2PIR. This is because: *(i)* relevance-based peer selection algorithms rely on global term statistics. However, the existing methods to estimate global term statistics may not be useful in real P2P networks (as discussed in Section 2.4); *(ii)* the existing peer selection algorithms cannot distinguish between trustworthy and untrustworthy peers.

- How to merge and re-rank documents returned from the selected peers because the document scores computed by each peer are not directly comparable.

- How to design a mechanism to organise distributed peers into an autonomous and collaborative manner so that relevant and trustworthy information can be collected, and document and peer scores can be computed by any peer in a P2P network.

- How to generate testbeds and develop methodologies for the evaluation of the proposed trust-aware P2PIR system in uncooperative P2PIR environments. This is because there are no standard testbeds for evaluating the performance of the proposed trust-aware P2PIR system in uncooperative P2PIR environments.

To address the above problems, a trust-aware P2PIR system is proposed in this chapter to find *relevant* and *trustworthy* documents for a given query in uncooperative P2PIR environments. The major contributions of this chapter are:

- Trust-based peer descriptions for the peer selection algorithm.

- A traditional resource selection algorithm is modified with the estimated global term statistics EPF (as described in Section 4.2.2.1) to compute relevance-based peer scores.

- A heuristic-based estimation function to merge results is proposed by extending the INQUERY result merging function with trust.

- A system architecture and data management protocols have been designed to implement the proposed trust-aware P2PIR system in uncooperative P2PIR environments, as well as structured P2P networks.

- A set of testbeds is developed to evaluate the performance of the proposed trust-aware P2PIR system in uncooperative environments. Preliminary experimental results show retrieval accuracy, effectiveness of trust and scalability of network size.

The remainder of this chapter is organised as follows: Section 5.2 proposes a trust-aware P2PIR system in uncooperative P2PIR environments, including issues of peer description, peer selection and result merging. The implementation strategy of the proposed trust-aware P2PIR system in uncooperative environments, as well as structured P2P networks, has been described in Section 5.3, including a system architecture and data management protocols. Section 5.4 presents the proposed testbeds, experimental methodologies and the initial experimental results of the performance of the proposed trust-aware P2PIR system. This chapter is concluded in Section 5.5.

5.2 Trust-Aware Information Retrieval in Uncooperative P2PIR Environments

This section mainly focuses on the techniques of trust-aware P2PIR system, including issues of trust-based peer description, peer selection and result merging. Peer description (in Section 5.2.1) represents the contents and reputation information of a peer for peer selection algorithms to choose the potential peers which can provide relevant and trustworthy documents. The proposed peer selection approach (in Section 5.2.2) ranks peers by the integrated scores of relevance and trustworthiness, and then a small number of top-ranked peers are selected to search. Result merging (in Section 5.2.3) can merge and re-rank the retrieved documents from each of the selected peers obtained from the peer selection step.

5.2.1 Peer Description

Peer description determines what content can be presented or is desirable to be presented for each peer. In this dissertation, it is similar to document description, and should provide sufficient information for peer selection algorithms to determine which peers are relevant and trustworthy. Prior research of distributed information retrieval proposes that the problem of resource description consists of three sub-problems: *discovering and representing what a resource contains*,

acquiring resource descriptions and *maintaining and updating resource descriptions* [Cal00, CC01]. Therefore, peer descriptions in uncooperative P2PIR environments should also address these three sub-problems. In this section, only the first problem will be described, which is how to represent a peer (i.e., document provider) for trust-aware P2PIR. The remaining two sub-problems will be addressed in Sections 5.3.2.2 and 5.3.2.3 because they are related to the implementation strategy of the proposed trust-aware P2PIR systems in uncooperative P2PIR environments, as well as structured P2P networks.

Since peer descriptions should provide sufficient information for peer selection algorithms to determine which peers are more likely to contain relevant and trustworthy documents for trust-aware P2PIR, peer descriptions should contain two aspects of information for relevance and trustworthiness computation. This is defined as *trust-based peer description* in this dissertation. The document description in the previous chapter is adapted to the peer description in uncooperative P2PIR. Therefore, in the proposed trust-aware P2PIR system, a peer can be represented by the tuple $< Con(p_k),\ Rep(p_k) >$, where $Con(p_k)$ describes the contents of peer p_k and $Rep(p_k)$ is the reputation value of peer p_k. Like document description, peer description is query-independent, which means peer description is only related to the peers themselves and this correlation is independent of different queries. Whatever the queries are, the reputation value and contents of a peer should be constant. Then, the next question to ask is how to represent the contents $Con(p_k)$ and reputation value $Rep(p_k)$ of peer p_k.

Full-text based peer descriptions are employed to describe the contents of a peer, because full-text based peer descriptions can provide much more comprehensive descriptions and are widely applied in uncooperative P2PIR environments. A full-text based peer description contains peer statistical information, such as terms, corresponding peer frequencies, and the number of document in the peer p_k. Moreover, to represent the reputation value $Rep(p_k)$ of the peer p_k, Equation 4.10 in Chapter 4 is applied in this chapter.

5.2.2 Peer Selection

Given a number of peer descriptions and a specific query, users should make a decision which peers to search because it is costly to forward the query to all of the peers in the network. The process of choosing a small set of peers to search is defined as *peer selection*. This usually depends on a peer selection algorithm which

can rank peers by the relevance-based scores for a given query in most existing uncooperative P2PIR systems, and then a pre-defined number of top-ranked peers are selected to search. Because both relevance and trustworthiness are the critical factors for trust-aware P2PIR, a trust-based peer selection algorithm is proposed by integrating the relevance and trustworthiness of a peer for a given query. Similar to the document ranking algorithm (as described in Section 4.2.2), the peer selection algorithm in this chapter can be calculated as a function of $R(p_k, q_i)$ and $T(q_i, p_k)$, where $R(p_k, q_i)$ is the relevance score between a peer p_k and a query q_i, and $T(q_i, p_k)$ is the trust value of the peer p_k for the given query q_i. To simplify the problem in this chapter, the relative weight of relevance and trustworthiness in the peer selection algorithm is assumed to be equal. Then, the peer ranking score in the proposed trust-aware P2PIR system is given by

$$S(q_i, p_k) = \sqrt{R^2(p_k, q_i) + T^2(q_i, p_k)}. \tag{5.1}$$

The next question is how to obtain $R(p_k, q_i)$ and $T(q_i, p_k)$ for peer ranking score computation. In Equation 4.1, the trust value $T(q_i, p_k)$ of the peer p_k for the given query q_i can be computed by Equation 3.9, which is described in Section 3.2.2.2. The remaining problem is how to compute the relevance-based peer scores $R(p_k, q_i)$ for Equation 4.1. To address this problem, an approach is proposed in the following paragraph.

To compute relevance-based peer scores $R(p_k, q_i)$ in uncooperative P2PIR, a number of traditional full-text based resource selection algorithms have been applied (as described in Section 2.3.2). The family of these resource selection algorithms needs global term statistics to compare the importance of the terms in the network. Existing peer selection approaches in uncooperative P2PIR either assume the global term statistics available in advance, or use document sampling and a reference corpus to estimate global term statistics. As discussed in Section 2.4, none of them are likely to be useful in practice. To address this problem, the estimated global term statistics EPF (as describe in Section 4.2.2.1) are employed, which is an estimated measure of the general importance of the term in the network, and is used in K-L for document ranking in cooperative P2PIR (as describe in Section 4.2.2.1). EPF will be integrated into a traditional resource selection algorithm to compute the relevance-based peer scores for the proposed

trust-aware P2PIR system. Since prior research has shown that the CORI resource selection algorithm [Cal00] is effective and stable in a wide variety of distributed information retrieval environments [CBH00, FPC+99, PFC+00, XC98] and uncooperative P2PIR environments [ZHTW08], the attention is initially restricted further than just the CORI resource selection algorithm. The resource score in CORI increases in proportion to the number of times a word appears in the different documents of the database but is offset by the frequency of the word in the databases of the network [Cal00].

EPF is used to replace the global term statistics in CORI to compute the relevance-based peer score for a given query in the proposed trust-aware P2PIR system, which is given by

$$T = \frac{nd_{(q_i,p_k)}}{s_{p_k}}, \tag{5.2}$$

$$EPF_{q_i} = log\frac{f(P_{q_i}, N)}{|x^a|}, \tag{5.3}$$

$$P(q_i|p_k) = b + (1-b) * T * EPF, \tag{5.4}$$

$$R(p_k, q_i) = P(q_i|p_k) = \frac{1}{|m|}\sum_{n=1}^{m} P(q_i|p_k). \tag{5.5}$$

where m is the number of query terms in a query Q, $nd_{(q_i,p_k)}$ is the number of occurrences of the query term q_i in the peer p_k, s_{p_k} is the number of documents in the peer p_k and b is the minimum belief component in CORI (e.g., 0.4).

Equations 5.2, 5.3 and 5.4 are components of Equation 5.5. When Equation 5.5 computed the relevance-based score $R(p_k, q_i)$ of peer p_k for Equation 5.1, then several top-ranked peers are selected to search and an equal number of documents is retrieved from each of these selected peers, while the numbers are pre-determined and fixed, depending on the empiricism [Cal00].

5.2.3 Result Merging

After a number of documents have been returned from each of the selected peers, they should be merged and re-ranked into a single list. Result merging is a challenging problem because the document ranking scores returned from

each selected peer cannot be directly compared. This is because different peers may employ different retrieval models and term statistics to compute document scores [SC03]. Therefore, the document scores usually need to be normalised. One solution to normalise document scores is that peers cooperatively exchange the corpus and document statistics [XC98]. Since this chapter assumes that peers do not provide document statistical information, this approach cannot work in this scenario. In such an environment, a solution is required which does not need specific cooperation from peers in the network. Where document scores can be estimated from the information obtained by observation, for example, document scores provided by each peer. A few heuristic-based estimation functions have been proposed for result merging in traditional distributed information retrieval [Cal00, SJCO02, CCB95], which combine the document score and text database score into an integrated scheme to produce a normalised document score. Among these methods, the INQUERY search engine [CCB95] provides an effective approach which can produce stable results in most IR testbeds. Therefore, the INQUERY merging function [CCB95] is selected to merge results in the proposed trust-aware P2PIR system. It is extended by combining relevance and trustworthiness. To simplify the problem, it is assumed that the weight between relevance and trustworthiness in the normalised document score is equal, then the estimated document score is given by

$$D' = (S(d_j, p_k, q_i) * Rep(d_j)) * \frac{1 + 0.4 * Score(p_k, q_i)'}{1.4}, \tag{5.6}$$

where D' is the estimated document score, $S(d_j, p_k, q_i)$ is the document d_j score returned from the peer p_k for the given query q_i, $Rep(d_j)$ is the reputation value of the document d_j, $Score(p_k, q_i)'$ is the normalised peer score, 0.4 and 1.4 are the constants in the INQUERY result merging function [CCB95].

Since without document statistics, the document trust model (as described in Section 4.2.2.2) cannot be used in the proposed trust-aware P2PIR system in uncooperative P2PIR environments. The reputation value of the document is used instead in this chapter. In Equation 5.6, $S(d_j, p_k, q_i)$ is assumed to be provided by peers, and $Score(p_k, q_i)'$ can be obtained by normalising the peer score returned by the peer selection algorithm in the proposed trust-aware P2PIR system (i.e., Equation 5.1). To compute $Score(p_k, q_i)'$, the methodology of normalising the peer score in INQUERY is employed in the proposed trust-aware P2PIR system, which is given by:

$$Score(p_k, q_i)' = \frac{Score(p_k, q_i) - Score(p_k, q_i)_{min}}{Score(p_k, q_i)_{max} - Score(p_k, q_i)_{min}}. \quad (5.7)$$

where $Score(p_k, q_i)$ is a peer score computed by the peer selection algorithm in Equation 5.1, $Score(p_k, q_i)_{max}$ for the peer p_k is calculated by setting the T component (i.e., Equation 5.2) in the peer selection algorithm to its maximum value 1, $Score(p_k, q_i)_{min}$ for the peer p_k is calculated by setting the T component to its minimum value 0.

5.3 Implementation Strategies

P2P networks lack any centralised infrastructure. One challenge which must be addressed is how to design a mechanism for organising peers in a cooperative manner so that peer contents and reputation information can be collected, and peer and document scores can be computed by any peer in the P2P network. Typical issues of implementing applications in a P2P network include decentralised system architectures and data management protocols. P2P network architecture determines each peer's functionality and responsibility, as well as data location schemes and message routing mechanisms. Similar to the implementation strategies of the proposed trust-aware P2PIR system in cooperative P2PIR and structured P2P networks (as described in Section 4.3), the proposed trust-aware P2PIR system in this chapter should consider the unique characteristics of structured P2P networks (e.g., completely decentralised, routing protocols) and uncooperative P2PIR (e.g., results merging). A Chord-based P2P network is used as the basic architecture to implement the proposed trust-aware P2PIR system in this chapter, and the system can also be applied to other structured P2P networks.

5.3.1 System Architecture

Since Chord is a completely decentralised P2P network, there is no central server and super peer. Each peer in the system should serve as a document provider, user and directory service at the same time. Therefore, the system architecture of each peer in the proposed trust-aware P2PIR system in uncooperative P2PIR environments should be designed to fulfil the following functions: *(i)* generating and storing peer descriptions (e.g., word statistics and reputation data) for directory

services; *(ii)* processing and forwarding queries; *(iii)* selecting a small number of peers to search for each query; and *(iv)* merging retrieved documents from each of the selected peers. Figure 5.1 shows the system architecture of the proposed trust-aware P2PIR system in uncooperative P2PIR environments, which consists of four components including *Statistics Manager, Reputation Manager, Ranker* and *Data Locator*.

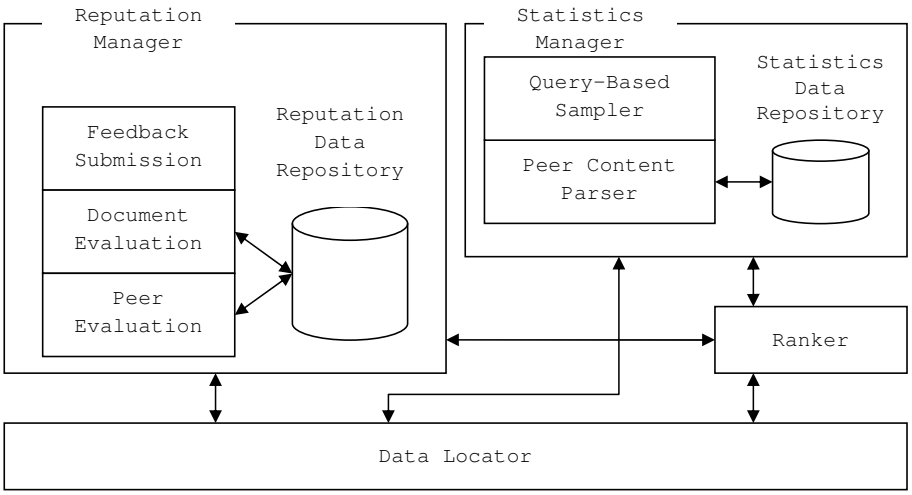


Figure 5.1: System architecture of the proposed trust-aware P2PIR system

The statistics manager consists of three components, namely, *Peer Content Parser*, *Query-based Sampler* and *Statistics Data Repository*. The peer content parser is responsible for extracting term statistics from peers such as terms, corresponding document frequencies, and the number of documents in a peer. The query-based sampler is responsible for sending queries to peers to obtain a number of sample documents for generating peer statistical descriptions $Con(p_k)$, when if peers do not provide content descriptions to the network. In structured P2P networks, there is a global index and each peer is responsible for storing a part of it (e.g., keys in the global key space). The statistics data repository is a small database which stores a portion of the global term statistical information. It contains: term-id, ID_{w_l}; peer-id, *Owners List*; document occurrence, $df_{(w_l,p_k)}$ and the number of document in the peer, s_{p_k}. An example is shown in Table 5.1. The term-id and peer-id are the hash values of converting the term and peer IP address to the respective numeric keys using the SHA-1 hash function [EJ].

The reputation manager extends the PeerTrust system architecture [XL04] with the document evaluation component. The reputation manager consists of

Table 5.1: Statistics Data Repository

ID_{w_l}	$Owners\ List$	$df_{(w_l,p_k)}$	s_{p_k}
w_1	p_{21}	13	583
w_2	p_1	5	1240
	p_{5021}	18	423
	p_{1468}	5	641
w_3	p_{79}	23	2315

Table 5.2: Reputation Data Repository (1)

ID_{d_j}	$Rep(d_j)$	$Owners\ List$
Doc_{21}	0.5	p_{12}
Doc_{10}	0.8	p_{124}
Doc_{79}	0.1	p_2
Doc_{47}	0.5	p_{35}

four components, which include *Feedback Submission, Document Evaluation, Peer Evaluation* and *Reputation Data Repository*. The reputation manager in the proposed trust-aware P2PIR system in uncooperative P2PIR environments is similar to the one in cooperative P2P environments. The feedback submission is responsible for submitting users' feedback. The document evaluation is responsible for evaluating the reputation value of document d_j. The peer evaluation computes the reputation value and trust value of peer p_k. The reputation data repository stores two hash tables which contain the reputation information for documents and peers, respectively. The information stored in the reputation data repository contains: document-id, ID_{d_j}; document reputation value, $Rep(d_j)$; peer-id, *Owners list*; peer reputation value, $Rep(p_k)$; and peer IP address, *Location*. Two examples are shown in Tables 5.2 and 5.3. The document-id is the hash value of converting the document name to the respective numeric key using the SHA-1 hash function.

The data locator is responsible for publishing, accessing and updating data in P2P networks. Different data placement and location schemes can be employed in different P2P networks. In terms of implementation, the data locator employs the Chord routing protocols [SMK+01] in this chapter. The ranker in the system performs the peer selection algorithm (as described in Section 5.2.2) and the result merging algorithm (as described in Section 5.2.3) for a given query and sorts the results in a descending order.

Table 5.3: Reputation Data Repository (2)

ID_{w_l}	*Owners List*	$Rep(p_k)$	*Location*
w_{34}	p_{12}	0.6	124.12.35.25
w_{15}	p_{75}	0.6	196.126.1.3
	p_{59}	0.8	139.36.25.1
w_{79}	p_2	0.3	96.12.178.3
w_{47}	p_{35}	0.6	84.13.256.6

5.3.2 Data Management Protocols

Since the proposed trust-aware P2PIR system is designed to be implemented in Chord structured P2P networks in this chapter, the data management protocols of the proposed system should extend the existing data management protocols of Chord in the context of term statistics and reputation data routing. By extending the resource index manager of Chord with peer content statistics, and document and peer reputation information, the proposed protocol in the trust-aware P2PIR system consists of the following phases: *(i)* create and publish peer descriptions; *(ii)* acquire peer descriptions for peer selection and result merging; and *(iii)* update peer descriptions and document reputation information.

5.3.2.1 Create and Publish Peer Descriptions

In uncooperative P2P environments, peers do not provide document copies and term statistics. Instead, they may provide statistical information of themselves, or not provide any kind of peer descriptions at all. When peers do not provide content descriptions, a method is employed to generate them, which is query-based sampling [CC01]. It learns the peer statistical description $Con(p_k)$ by submitting queries to the peer p_k and obtaining a set of sampled documents from that peer p_k. Query-based sampling is performed by the query-based sampler in the proposed trust-aware P2PIR system (as shown in Figure 5.1). To create the reputation description $Rep(p_k)$ of the peer p_k, the peer evaluation component in the reputation manager needs to collect and average the document reputation values shared by the peer p_k. If new documents join the network, or the document has never been used before (hence, there is no reputation information), the document reputation value is equal to the peer reputation value which provides that document.

As a peer joins the network for sharing documents, two types of messages are

simultaneously issued by the statistics manager and reputation manager. When the peer provides content description information, the peer content parser should extract statistical information. If it does not, then the query-based sampler is employed to generate content description information. Afterwards, the data locator converts the terms and peer IP address to numeric keys by using the SHA-1 hash function, and then sends $PUBLISH(ID_{w_l}, p_k, df_{(w_l,p_k)}, s_{p_k})$ messages for each distinct term w_l in the peer p_k to the peers responsible for the term keys ID_{w_l}. The peer responsible for the key ID_{w_l} adds the peer-id, $df_{(w_l,p_k)}$ and s_{p_k} to the statistics data repository. In the meantime, the reputation manager should collect the reputation values of documents, which the peer p_k is sharing, from the network, and compute its reputation value. To collect the reputation values of documents, the data locator takes the hash value of the document name ID_{d_j} as the argument to retrieve the reputation value of the document d_j. The process of retrieving the reputation values will be described in the following section.

5.3.2.2 Acquire Peer Descriptions for Peer Selection and Result Merging

Once peer descriptions have been created and stored in the network, they are ready for peer selection. To acquire peer descriptions, three protocols in terms of $FIND^{sta}_{peer}$, $FIND^{rep}_{peer}$ and $FIND^{rep}_{doc}$ are proposed. $FIND^{sta}_{peer}$ takes the hash value of term ID_{w_l} as an argument to obtain the list of peers containing the query term w_l and corresponding term statistics. Simultaneously, $FIND^{rep}_{peer}$ takes the hash value of term ID_{w_l} as an argument to obtain the corresponding peer reputation value and IP address. Afterwards, the ranker performs the peer selection algorithm with statistical and reputation data to determine which peers to search. Then, queries are forwarded to each of the selected peers. Once the selected peers have been searched, a number of documents should be returned from each of them. The $FIND^{rep}_{doc}$ lookup messages are issued by the data locator to retrieve the reputation values of documents returned from the selected peers for result merging. The data locator takes the hash values of document names as arguments to find document reputation values. Finally, the ranker merges the results and produces a single ranked list.

5.3.2.3 Update Peer Descriptions and Document Reputation Information

When peer descriptions changed, peers need to update their descriptions in the network. The peer description can be updated periodically, or when the differences between the previous description and the new description are significant. Updating a peer description consists of two parts, which are statistical information update and reputation information update. The process of updating statistical descriptions for each peer is identical to the process for creating and publishing peer statistical descriptions. To update the reputation information, users need to evaluate the document and leave feedback for it. If a user thinks the document is trustworthy, the feedback value is set to be 1, otherwise, it is set to be 0. In order to update the reputation values of document d_j and peer p_k, the data locator sends a message $UPDATE(ID_{d_j},\ ID_{p_k}, f(p_k)_{(d_j)})$ to the peer responsible for that document reputation value. The document evaluation of the reputation manager in the proposed trust-aware P2PIR system should compute the new document reputation value. Then, the new document reputation value is used by the peer evaluation of the reputation manager to calculate the new peer reputation value.

5.4 Evaluation

The proposed trust-aware P2PIR system in uncooperative P2PIR environments finally should generate an ordering of the retrieved documents. Documents appearing at the top of this ordering are considered to be more likely to satisfy users' requirements, which are trustworthiness and relevance, in this dissertation. The objective of this section is to evaluate the three aspects of the proposed trust-aware P2PIR system in uncooperative P2PIR environments, which are *retrieval accuracy*, *effectiveness of trust*, as well as *scalability*. In Section 5.4.1, the experimental settings and methodologies are described. Section 5.4.2 discusses the initial experimental results.

5.4.1 Experimental Settings and Methodologies

The same as for trust-aware P2PIR in cooperative P2PIR environments, there is no standard testbed available to evaluate the performance of the proposed trust-aware P2PIR system in uncooperative P2PIR environments. To fulfil this task, a set of testbeds for the evaluation of the proposed trust-aware P2PIR system is generated, which is modifications of the proposed trust-aware P2PIR testbeds in cooperative P2PIR environments (as described in Section 4.4.1). The major difference between the cooperative P2PIR testbeds and uncooperative P2PIR testbeds is that, in uncooperative P2PIR environments, each peer provides an individual search engine to index and retrieve documents on request, which does not happen in the cooperative P2PIR testbeds. Although, the proposed trust-aware P2PIR system do not make the assumption that each peer uses the same document indexing and retrieval approaches. For the consideration of convenience of experiments, each peer in this experiment is a text collection running the INQUERY search engine and provides relevance-based document scores for given queries. INQUERY is implemented by Lemur toolkit [lem]. The experiment initially restrict that the number of selected peers to search to 10% of the network size. The TREC topics 451-550 short query set, and relevant and trustworthy judgement assessments in cooperative trust-aware P2PIR testbeds (as described in Sections 4.4.1.2 and 4.4.1.3) are employed in this experiment.

The peer and document reputation values are developed based on PeerTrust, which is a reputation-based trust management system to filter feedback from malicious peers. Two types of peers are simulated in this experiment, which are good peers and malicious peers. All of the documents provided by good peers are set to be trustworthy. On the contrary, all of the documents provided by malicious peers are untrustworthy. The experimental settings in this section are the same as those described in Section 4.4.1.4. The percentage of malicious peers is initially set to 20%, randomly selected from the network. Good peers provide positive feedback to trustworthy documents and negative feedback to untrustworthy documents. On the other hand, malicious peers submit positive feedback for malicious peers, and negative feedback for the documents provided by the good peers. The credibility values (as defined in Section 4.2.1) of good peers and malicious peers are randomly set to 90% and 20%, respectively. The number of feedback entries for each document is initially set to 10.

The performance of the proposed trust-aware P2PIR system in uncooperative

P2PIR environments is measured by retrieval accuracy, effectiveness of trust and scalability of network size. The experimental methodologies and metrics used in this chapter are same as those used in the previous chapter (as described in Section 4.4.2). The experimental results will be discussed in the next section.

5.4.2 Experimental Results

This section focuses on evaluating the performance of the proposed trust-aware P2PIR system in uncooperative P2PIR environments, to demonstrate that: *(i)* the proposed estimated global term statistics integrated with the CORI resource selection algorithm can achieve competitive retrieval accuracy, compared to the existing peer selection algorithms with accurate global term statistics; *(ii)* the proposed trust-aware P2PIR system can provide a better combination of the retrieval accuracy and the effectiveness of trust in protecting untrustworthy documents in the top-ranked results list than several current peer and file trust models; and *(iii)* the system can be scaled to large-sized networks. Four sections are devoted to the experimental results with regard to evaluating: *(i)* the retrieval accuracy of the estimated global term statistics (EPF) with CORI; *(ii)* the effectiveness of different trust models in protecting untrustworthy documents in the top ranked results list; *(iii)* the effect of different trust models on retrieval accuracy; *(iv)* system scalability of network size.

- ***Evaluation of Retrieval Accuracy***

  - *[Experiment 1 in Section 5.4.2.1]*: this experiment evaluates the retrieval accuracy of the proposed estimated global term statistics EPF (as described in Section 4.2.2.1) in uncooperative P2PIR environments, which is integrated with CORI for relevance-based peer score computation in the proposed trust-aware P2PIR system. Accurate global term statistics with CORI [ZHTW08], K-L [CSB+05, ZL06, LC07a, LC06, LC05, LC03, ZCLL04] and VSM [ZYKG05, ZYKG07, RPTW08, ZTW07] peer selection algorithms used in uncooperative P2PIR (as described in Section 2.2.3.2) are implemented and compared in this experiment. To shield the evaluation of the retrieval algorithm from the factors which may affect retrieval accuracy, the experiment does not take trust metrics into account.

- ***Evaluation of Effectiveness of Trust***
  - *[Experiment 2 in Section 5.4.2.2]*: the objectives of this experiment are: *(i)* to explore the effectiveness of the proposed trust-aware P2PIR system, and *(ii)* to study the effectiveness of different trust models to protect untrustworthy documents in the top-ranked results list. Firstly, to study the effectiveness of the trust models in the proposed trust-aware P2PIR system, an approach without trust will be compared with the proposed trust-aware P2PIR system by the percentage of untrustworthy documents retrieved in the top-ranked results list. Moreover, several existing most cited reputation-based peer and file trust models have been selected for comparison. To ignore the effect of relevance-based retrieval algorithms on the evaluation of the effectiveness of different trust models, the relevance-based peer selection algorithm and result merging algorithm are not taken into account.
  - *[Experiment 3 in Section 5.4.2.3]*: this experiment explores the effect of different trust models on retrieval accuracy. When combining trust metrics to peer selection and result merging, retrieval accuracy should be sacrificed because the relevant but untrustworthy peers and documents are removed from the top-ranked results list. Moreover, the existing trust models only produce entity trust, which do not take into account relationships between queries and contents, so the irrelevant but trustworthy peers and documents get higher ranks. This can decrease retrieval accuracy. This experiment studies the percentage of the degrading of retrieval accuracy yielded by different trust models. In order to focus on the comparison of retrieval accuracy, as reduced by different trust models, the same relevance-based peer selection algorithm (i.e., Equation 5.5) and result merging algorithm (as described in Section 5.2.3) are integrated with different trust models.
- ***Evaluation of Scalability***
  - *[Experiment 4 in Section 5.4.2.4]*: this experiment evaluates the scalability of the proposed trust-aware P2PIR system in uncooperative P2PIR environments on retrieval accuracy and the effectiveness of trust. The experiment employs four testbeds which are 100, 200, 400, and 1000-peer ones. The experimental results show the performance

of both the retrieval accuracy and effectiveness of trust in protecting untrustworthy documents in different testbeds.

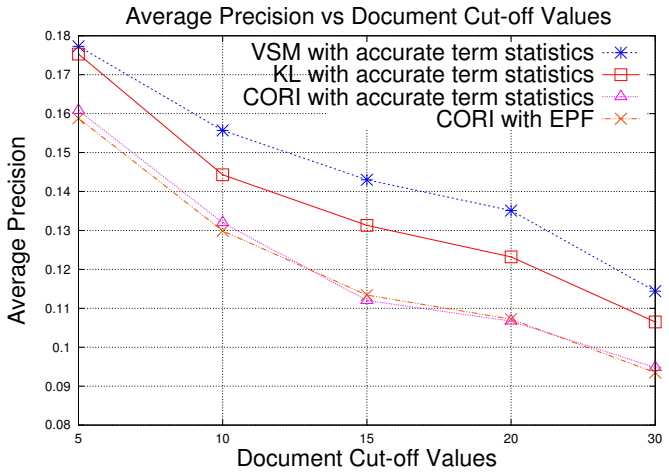
According to the literature related to uncooperative P2PIR (as described in Section 2.2.3.2), the most widely used relevance-based peer selection algorithms are the vector space model [ZYKG05, ZYKG07, RPTW08, ZTW07], K-L [CSB+05, ZL06, LC07a, LC06, LC05, LC03, ZCLL04] and CORI [ZHTW08]. In this experiment, these algorithms are compared with the peer selection approach in the proposed trust-aware P2PIR system for retrieval accuracy study. To explore the effectiveness of the proposed trust-aware P2PIR system, three existing reputation-based trust approaches are implemented for comparison. These are PeerTrust [XL04], PowerTrust [ZH07] and eBay [eBa].

5.4.2.1 Retrieval Accuracy

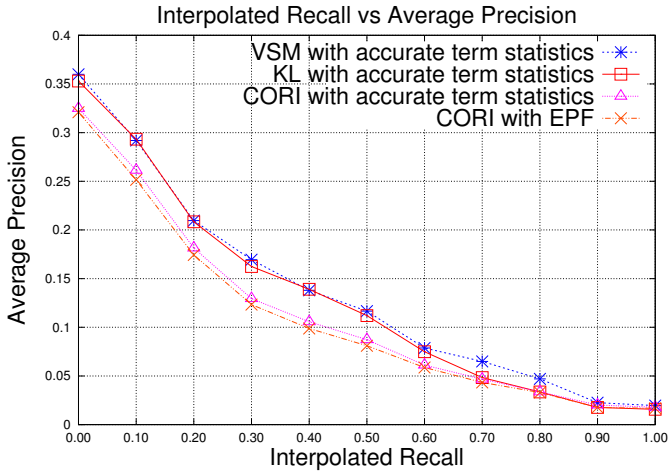
This experiment focuses on exploring the retrieval accuracy of the estimated global term statistics (EPF) within the peer selection algorithm in the proposed trust-aware P2PIR system. To study the effectiveness of the estimated global term statistics in uncooperative P2PIR (as described in Section 4.2.2.1), the retrieval accuracy of CORI with accurate global term statistics and with estimated global term statistics are compared. Moreover, in order to study the retrieval accuracy of the peer selection algorithm in the proposed trust-ware P2PIR system, several existing most used relevance-based peer selection algorithms in uncooperative P2PIR are also compared.

In Figure 5.2, CORI with EPF is the approach used to compute relevance-based peer scores (i.e., Equation 5.5) in the proposed trust-aware P2PIR system. VSM, KL, and CORI with accurate term statistics are several most used peer selection algorithms in uncooperative P2PIR (as described in Section 2.2.3.2). Figure 5.2 shows the experimental results of the retrieval accuracy of different approaches for the TREC 451-550 short query set in the 1000 peer-sized network. Figure 5.2 (a)-(b) depicts the results of two evaluation metrics, which are (a) the average precision vs. document cut-off values, and (b) 11-point interpolated recall vs. average precision. The higher the average precision in both figures, the better the retrieval accuracy achieved by the relevance-based peer selection algorithm.

In Figure 5.2 (a) K-L with accurate term statistics can achieve better retrieval accuracy than CORI with accurate term statistics and CORI with EPF,



(a) average precision vs document cutoff values



(b) average precision vs interpolated recall

Figure 5.2: Retrieval accuracy of different relevance-based peer selection algorithms for the TREC 451-550 short query set in the 1000 peer-sized network.

but worse than VSM with accurate term statistics, indicating that VSM is the best relevance-based peer selection algorithm by the TREC short query set in existing uncooperative P2PIR. The retrieval accuracy of CORI with accurate term statistics is close to CORI with EPF, which means that EPF can provide competitive accuracy compared with the accurate global term statistics on user satisfaction. In Figure 5.2 (b), the gaps between curves of VSM with accurate term statistics and K-L with accurate term statistics are very small, and they both perform better than CORI with accurate term statistics and CORI with EPF. This indicates that K-L and VSM can provide the better overall retrieval performance than the CORI peer selection approach. Moreover, the curves of

CORI with accurate term statistics and EPF are close, demonstrating that the estimated global term statistics can achieve competitive retrieval accuracy on the overall retrieval performance. The reason for this is that the original global term statistics in CORI is a peer level statistic and EPF is also a peer level statistic. Therefore, EPF can provide better accuracy in the relevance-based peer selection algorithm than it does in the document ranking algorithm (as shown in Section 4.4.3).

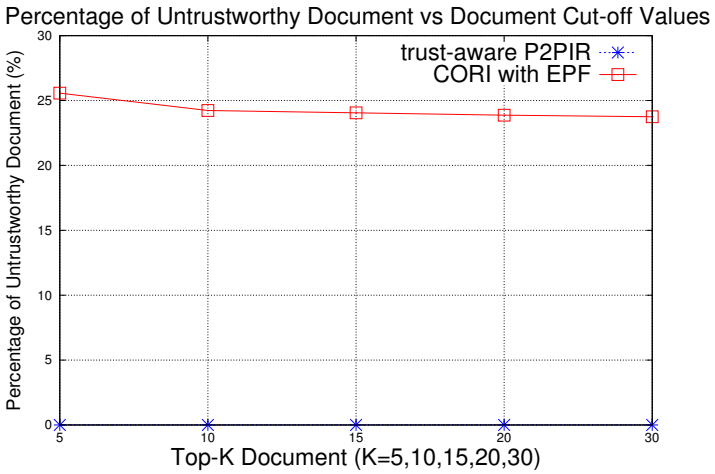
In summary, the following conclusions can be made: *(i)* for both user satisfaction and the overall retrieval performance, VSM and K-L yield better results than CORI for the TREC topics 451-550 short query set; *(ii)* EPF can provide competitive retrieval accuracy in relevance-based peer score computation, compared to the accurate global term statistics.

5.4.2.2 Effectiveness of Trust

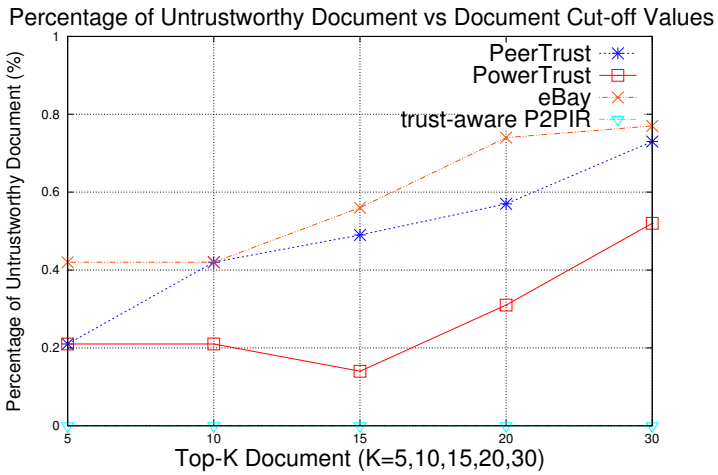
This experiment explores the effectiveness of the trust models in the proposed trust-aware P2PIR system in uncooperative P2PIR environments. Figure 5.3 shows the experimental results of the percentage of untrustworthy documents appearing in the top-ranked results list, for the TREC topics 451-550 short query set by different approaches, which are trust-aware P2PIR (i.e., Equation 5.1), CORI with EPF, PeerTrust [XL04], PowerTrust [ZH07] and eBay [eBa].

Firstly, to study the effectiveness of the trust models (i.e., peer trust model Equation 4.9 and document reputation model Equation 4.1) in the proposed trust-aware P2PIR system, an approach without trust is compared with the proposed trust-aware P2PIR system by the percentage of untrustworthy documents retrieved in the top-ranked results list. The relevance-based peer selection CORI+EPF (i.e., Equation 5.5) combined with INQUERY result merging [CCB95] are selected as the baseline, which is called *CORI with EPF* in Figure 5.3. Figure 5.3 (a) displays the results of the two approaches with and without the proposed trust models. It can be observed from the figure that: *(i)* the percentages of untrustworthy documents generated by the approach without trust remain high in the top-ranked results list, demonstrating that the trustworthiness of peers and documents is a critical factor for uncooperative P2PIR environments. It is risky for users to run a higher risk to review and download untrustworthy documents, even if they are relevant to a given query; *(ii)* the percentage of untrustworthy documents appearing in the top-30 documents is 0

when the trust models in the proposed trust-aware P2PIR system are employed. This indicates that the trust models in the peer selection and result merging of the proposed trust-aware P2PIR system can effectively protect untrustworthy documents appearing in the top-30 of the results list.



(a) comparison of retrieval methods with and without trust models



(b) comparison of effectiveness of different trust models

Figure 5.3: Effectiveness of different trust methods to protect untrustworthy documents appearing in the top-ranked results list for the TREC 451-550 short query set in the 1000 peer-sized network.

Secondly, to study the effectiveness of different trust models in the peer selection and result merging algorithm to filter untrustworthy documents, several existing most cited reputation-based peer and file trust models are selected for comparison. To ignore the effect of relevance-based retrieval algorithms on the evaluation of the effectiveness of different trust models, the relevance-based peer

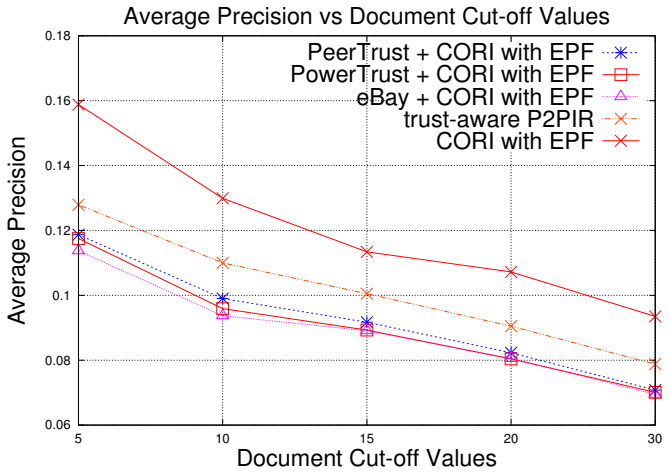
selection algorithm and result merging algorithm are not taken into account in this experiment. Figure 5.3 (b) displays the comparison of the effectiveness of different trust models to protect untrustworthy documents appearing in the top-ranked results list. As displayed in the figure, only the proposed trust-aware P2PIR system can completely filter untrustworthy documents from the top-ranked results list. This is because the proposed trust-aware P2PIR system employs two trust models, namely, the peer trust model and the document reputation model. The peer trust model can filter untrustworthy peers from the network and the document reputation model can filter untrustworthy documents from the selected peers, which does not happen in the existing trust models. This indicates that the document reputation and peer trust models in peer selection and result merging can yield a better performance than current most cited trust management systems in terms of PeerTrust, PowerTrust and eBay.

In summary, the experimental results suggest that uncooperative P2PIR can benefit significantly from the proposed trust-aware P2PIR system and protect users from reviewing or downloading untrustworthy documents. Moreover, the trust models in the proposed trust-aware P2PIR system provide better effectiveness to filter untrustworthy documents than the existing trust models.

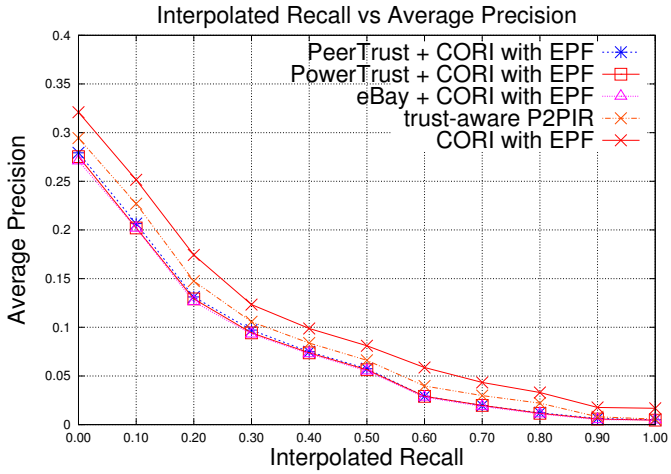
5.4.2.3 Effect of Different Trust Models on Retrieval Accuracy

In the previous experiment, the effectiveness of different trust models to protect untrustworthy documents in the top-ranked results list was explored. In this experiment, the effect of different trust models on retrieval accuracy is studied for the TREC topic 451-550 short query set in the 1000-peer network. The relevance-based peer selection CORI+EPF (i.e., Equation 5.5) combined with INQUERY result merging [CCB95] are selected as the baseline, which is called CORI with EPF in Figure 5.4. The two retrieval accuracy metrics are applied in this experiment.

Figure 5.4 shows the experimental results for the TREC 451-550 short query set using different trust models in the 1000-peer network. Figure 5.4 (a)-(b) plots the results of the two evaluation metrics, which are (a) the average precision vs document cut-off values, and (b) 11-point interpolated recall vs average precision. As expected, CORI+EPF can achieve the best retrieval performance in Figure 5.4 (a) and (b), because CORI+EPF and INQUERY result merging can rank both trustworthy and untrustworthy documents in the top-ranked results list.



(a) average precision vs document cutoff values



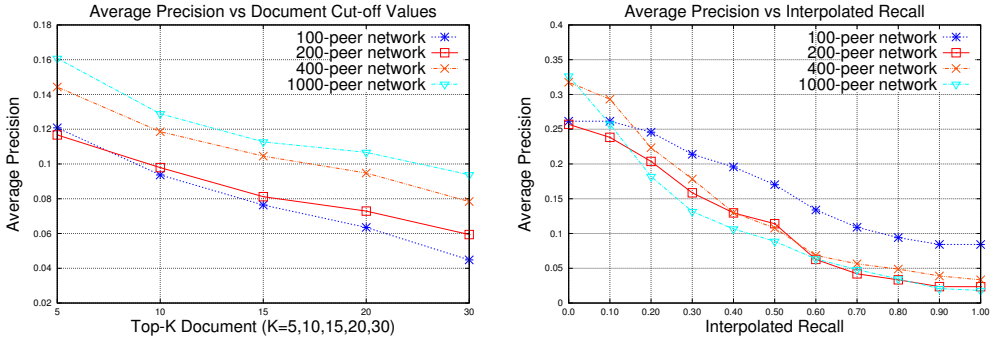
(b) average precision vs interpolated recall

Figure 5.4: Retrieval accuracy of different trust models for the TREC 451-550 short query set in the 1000 peer-sized network.

Figure 5.4 demonstrates that the proposed trust-aware P2PIR system is able to provide better retrieval accuracy than PowerTrust, PeerTrust and eBay in terms of both user satisfaction and the overall retrieval performance. The reason for this is that these existing trust models only consider entity trust, but do not take into account relationships between peers or document contents and given queries. This means that trustworthy peers and documents get high scores and ranks whether they are relevant or not. Some trustworthy but irrelevant peers and documents can be ranked in the top-ranked results list, which may reduce retrieval accuracy. This phenomenon shows that: *(i)* relationships between peer and document contents, and queries are essential factors for calculating trust

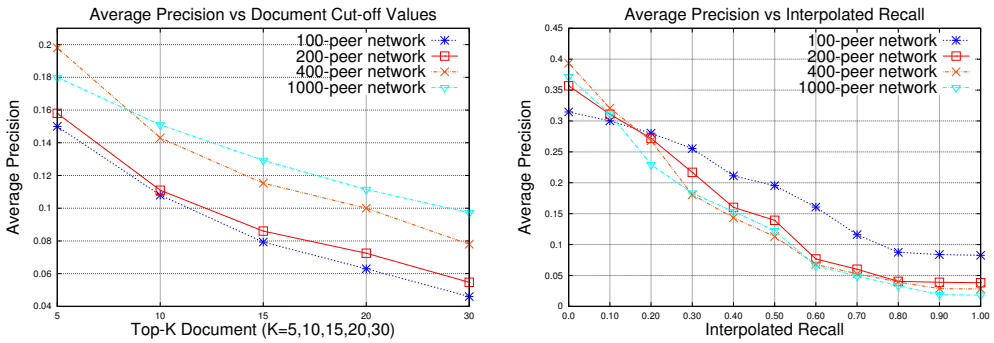
scores for uncooperative P2PIR; *(ii)* the proposed trust-aware P2PIR system can achieve better retrieval accuracy than the existing entity trust models.

In summary, combining the conclusion of the previous experiment, the proposed trust-aware P2PIR system can provide the better retrieval accuracy and the effectiveness of protecting untrustworthy documents in the top-ranked results list than the existing entity trust models.



(a) average precision vs document cut-off (b) average precision vs interpolated recall values

Figure 5.5: Retrieval accuracy of the proposed trust-aware P2PIR system for the TREC 451-550 short query set in the different sized networks.



(a) average precision vs document cut-off (b) average precision vs interpolated recall values

Figure 5.6: Retrieval accuracy of CORI for the TREC 451-550 short query set in the different sized networks.

5.4.2.4 Scalability of Network Size

This set of experiments study the scalability of the proposed trust-aware P2PIR system on retrieval accuracy and the effectiveness of trust to protect untrustworthy documents in the top-ranked results list in different sized networks. Four

testbeds are employed in the experiments, which are three small-sized testbeds and a medium-size one. Figure 4.5 shows the experimental results of the retrieval accuracy of the proposed trust-aware P2PIR system for the TREC short query set in different sized networks. Figure 5.5 (a)-(b) shows the results of two evaluation metrics, which are (a) the average precision vs document cut-off values, and (b) 11-point interpolated recall vs average precision. It can be seen from Figure 5.5 (a), that the larger the network is, the better retrieval accuracy the system can provide, which indicates that the proposed trust-aware P2PIR system can scale better in large-scale networks on retrieval accuracy for user satisfaction. However, the overall retrieval accuracy decreased when the network scaled to larger ones (as shown in Figure 5.5 (b)), indicating that the proposed trust-aware P2PIR system favours user satisfaction more than the overall retrieval performance. The reason for this is that the CORI peer selection algorithm favours user satisfaction more than the overall retrieval performance, which has been proved by the retrieval accuracy of the CORI peer selection in Figure 5.6 (a)-(b). However, the estimated global term statistics EPF and trust models in the proposed trust-aware P2PIR system make the some changes on points between Figure 5.5 (a)-(b) and Figure 5.6 (a)-(b). For example, the precision of the top-5 documents in a 1000-peer network is worse than that of any other networks in Figure 5.5 (a), the precision of the top-5 documents in a 1000-peer network is just worse than that of a 400-peer network in Figure 5.6 (a). In general, the shapes of curves in Figure 5.5 (a)-(b) and Figure 5.6 (a)-(b) are very similar, which means that the scalability of the proposed trust-aware P2PIR system on retrieval accuracy mainly depends on the scalability of the CORI peer selection algorithm. Since CORI has been shown to be scalable to large-scale networks by different testbeds in traditional Information Retrieval [Cal00, CC01], the proposed trust-aware P2PIR system can also provide scalability of retrieval accuracy in large-scale networks.

The experimental results of the percentage of untrustworthy documents in the top-ranked results list are all zero in 100, 200, 400 and 1000 sized networks. This indicates that the proposed trust-aware P2PIR system can effectively filter untrustworthy peers and documents for given queries when the network scales to a larger sized network.

5.5 Summary

This chapter proposed a trust-aware P2PIR system in uncooperative P2PIR environments. The estimated global term statistics are integrated with CORI to compute relevance-based peer scores in the proposed trust-aware P2PIR system. Trust-based peer description, peer selection and result merging approaches are proposed to find not only relevant but also trustworthy peers and documents in the network. To implement the proposed trust-aware P2PIR system in structured P2P networks and uncooperative P2PIR environments, a system architecture and data management protocols are designed, which are the extension of the PeerTrust architecture and data management protocols of structured P2P networks in the context of trust-aware P2PIR. A set of testbeds is developed to evaluate the performance of the proposed trust-aware P2PIR system on retrieval accuracy, effectiveness of trust, and scalability of network size.

Chapter 6

A Theoretical-Based Peer Selection Approach in Uncooperative P2PIR Environments

6.1 Introduction

In the previous chapter, the problem of trust-aware P2PIR in uncooperative P2PIR environments was addressed, including issues of *trust-based peer description, peer selection, result merging* and *implementation*. Among these problems, peer selection means to select a set of peers to search, which are relevant and trustworthy for a given query. The same as in most existing peer selection algorithms, a peer ranking is computed in a heuristic way (as described in Section 2.2.3.2). A heuristic-based peer selection algorithm (as described in Section 5.2.2) is designed to rank peers based on their relevance and trustworthiness in the proposed trust-aware P2PIR system in uncooperative P2PIR environments. Then, the number of the top-ranked peers in the results list of peers is selected to search, and an equal number of documents is retrieved from each of the selected peers. These numbers are pre-determined and fixed empirically. For example, the top-10 peers are selected to search, and the top-100 documents are retrieved from each selected peer. In fact, these numbers are query-specific[1]. It is not appropriate to

[1]There are 66 peers containing relevant documents for the TREC query 452, in the 1000 peers testbed, but only 12 peers with relevant documents for the TREC query 521.

use a fixed and pre-determined threshold to decide the number of peers to search and the number of documents to retrieve from each selected peer for any queries. For example, it is costly to select 30 peers to search when there are only 10 peers relevant to the given query, and selecting 10 peers to search is not enough, when there are 30 peers relevant to the given query. The challenge is how to design an approach which has the ability to compute a clear cut-off for the number of peers to search, and the number of documents to be retrieved from each of these selected peers.

To address this problem, a theoretical-based peer selection model is proposed, which is inspired by the decision-theoretic framework (DTF) approach [Fuh99]. The contributions of this chapter are:

- A precision-risk $PrRi$ peer selection model which computes a clear cut-off value for which peers to search, and the particular number of documents to retrieve from each of the selected peers.
- A system architecture and data management protocols are proposed to implement the $PrRi$ peer selection model in structured P2P networks.
- A set of experiments has been conducted and the results show the advantages and disadvantage of the $PrRi$ peer selection model, compared to the heuristic-based peer selection approach (as described in Section 5.2.2).

In the remainder of this chapter, the precision-risk peer selection model is described in Section 6.2, after which the methodology of DTF to estimate the number of relevant documents in the results set is reviewed in Section 6.3. Section 6.4 explores the implementation strategies of the proposed precision-risk peer selection model in structured P2P networks. Section 6.5 discusses the different between DTF and PrRi. Section 6.6 presents the experimental methodologies and initial experimental results of the performance of the precision-risk peer selection model. In Section 6.7, a variant of the precision-risk peer selection model is considered, and this chapter is concluded in Section 6.8.

6.2 The Precision-Risk Peer Selection Model

In uncooperative P2PIR, a set of peers are selected to search. Each peer can run a text-based search engine, and produce a ranked list for any queries. Then,

an equal number of documents is retrieved from the ranked list of the selected peers. This is defined as *peer selection*. The proposed trust-aware P2PIR system (as described in Chapter 5), in uncooperative P2PIR environments, employs a heuristic-based approach to rank peers, and then selects fixed and pre-determined numbers of peers and documents. However, these numbers make peer selection either lose relevant and trustworthy peers and documents, or obtain redundant irrelevant and untrustworthy peers and documents.

To address this problem, this section proposes a precision-risk peer selection model, which can explicitly compute which peers to search, and the number of documents to retrieve from each of the searched peers. Compared with the heuristic-based peer selection approach (i.e., Equation 5.1), the precision-risk peer selection model has a better theoretical foundation. The numbers can be computed by maximising the number of relevant documents, and minimising the risk of reviewing or downloading untrustworthy documents in the results set. The precision-risk peer selection model will be described in what follows.

The precision-risk ($PrRi$) model is proposed to compute the number of relevant documents and the risk values of documents in the results set. In the $PrRi$ peer selection approach, *precision* is a fraction of the relevant documents for a given query in the retrieved documents [BYRN99], and *risk* is the aggregation of risk assessments of documents in the results set. To be consistent with the peer selection algorithm in Section 5.2.2, the relative weight of relevance and trustworthiness in the proposed precision-risk peer selection model is assumed to be equal. Then, by retrieving n documents, the precision-risk peer selection model is given by:

$$PrRi(n) = Precision(n) * Risk(n) = \frac{r}{n} * Risk(n), \tag{6.1}$$

where $Precision(n)$ and $Risk(n)$ are the precision and risk value of the results set; n is the number of documents in the results set; r is the number of relevant documents in the results set for a given query.

Since the results set consists of relevant and irrelevant documents for a given query, $Risk(n)$ can be represented by the sum of the risk values of relevant documents $Risk^{+}(n)$ and irrelevant documents $Risk^{-}(n)$. It is risky for users to review untrustworthy documents (e.g., documents with a virus or retrieved documents which are different from their descriptions in the network), even if they are relevant. In another case, users may not review trustworthy documents which are

irrelevant in Information Retrieval systems. Therefore, it is assumed that users are only interested in the risk values of relevant documents for a given query rather than those of irrelevant documents. Then, the $PrRi$ model (i.e., Equation 6.1) is represented and simplified by:

$$PrRi(n) = \frac{r}{n} * (\sum Risk^{+}(n) + \sum Risk^{-}(n)) \approx \frac{r}{n} * \sum Risk^{+}(n), \qquad (6.2)$$

The relevance of a document for a given query is query-specific, and this is dependent on the user's judgement. Since users have very different backgrounds and knowledge, even for the same query, they may have different results of relevant documents retrieved from the same database. It is hard to accurately determine which particular document is relevant, and the number of relevant documents in the results set, before users provide the relevance feedback for the retrieved documents. Therefore, the challenge of the $PrRi(n)$ model is that the precision value of the results set $\frac{r}{n}$ and the risk values of relevant documents $Risk^{+}(n)$ in the results set cannot be obtained before query processing. The only way to address this problem is to estimate the precision and risk value of the results set. Then, the estimated values of the precision-risk model $EPrRi(n)$ can be arrived at by retrieving n documents from the network. $EPrRi(n)$ is given by:

$$EPrRi(n) = EPrecision(n) * ERisk(n) \approx \frac{er}{n} * \sum ERisk^{+}(n), \qquad (6.3)$$

where $EPrecision(n)$ and $ERisk(n)$ are the estimated precision and risk value of the results set; er is the estimated number of relevant documents, and $ERisk^{+}(n)$ is the estimated risk value of a relevant document for a given query in the results set.

Assuming that there are k different peers in the network, the results set is the aggregation of documents retrieved from those peers. A corresponding vector $n = (n_{p_1}, n_{p_2}, ..., n_{p_k})$ gives the number of documents to be retrieved from each peer, and the union of them is the number of documents in the results set n. Then, Equation 6.3 is represented by:

$$EPrRi(n) \approx \frac{\sum_1^k er_{p_k}}{\sum_1^k n_{p_k}} * \sum_1^k ERisk^+_{p_k}(n_{p_k}) = \frac{\sum_1^k er_{p_k}}{n} * \sum_1^k ERisk^+_{p_k}(n_{p_k}), \quad (6.4)$$

where $ERisk^+_{p_k}(n_{p_k})$ is the estimated risk value of a relevant document for a given query in the results set n_{p_k} of peer p_k, er_{p_k} is the estimated number of relevant documents in the results set n_{p_k} of peer p_k.

Since it is hard to know which document is relevant for a given query before query processing, the risk values of the relevant documents retrieved from peer p_k can only be estimated. The document risk value $Risk(d_j)$ in this chapter is set as the reciprocal value of the document reputation value $Rep(d_j)$ (i.e., Equation 4.1). The peer risk value $Risk_{p_k}$ is defined as the aggregation of the risk values of documents $Risk(d_j)$ in that peer. It is assumed that the risk values of the documents retrieved from the peer p_k are proportional to the risk value of peer $Risk_{p_k}$. Moreover, it is assumed that the risk values of the relevant documents in the results set are proportional to the risk value of the results set from peer p_k. Then, the estimated risk value of relevant documents $ERisk^+_{p_k}(n_{p_k})$ in the results set n_{p_k} retrieved from peer p_k are given by:

$$ERisk^+_{p_k}(n_{p_k}) \approx Risk_{p_k} * \frac{n_{p_k}}{nd_{p_k}} * \frac{er_{pk}}{n_{p_k}} = Risk_{p_k} * \frac{er_{p_k}}{nd_{p_k}}, \quad (6.5)$$

where nd_{p_k} is the total number of documents in peer p_k.

By combining Equations 6.4 and 6.5, the estimated value of the precision-risk model $EPrRi$ is given by:

$$EPrRi(n) \approx \frac{\sum_1^k er_{p_k}}{n} * \sum_1^k (Risk_{p_k} * \frac{er_{p_k}}{nd_{p_k}}). \quad (6.6)$$

The methodology of estimating the number of relevant documents er_{p_k} in the results set of peer p_k will be described in the next section. When users specify the number of documents n to be retrieved in the results set, the $EPrRi$ peer selection algorithm computes a selection (i.e., which peers to search and the number of documents to received from each searched peer) by simultaneously maximising the number of relevant documents, and minimising the risk values of documents in the results set. In other words, a selection is obtained by minimising the number of irrelevant documents and the risk values of documents in the results

set. Then, we can arrive at:

$$minimise\ EPrRi(n), \tag{6.7}$$

subject to

$$M = \{(n_{p_1}, n_{p_2}, ..., n_{p_k}) : n_{p_1} + n_{p_2} + ... + n_{p_k} = n\}. \tag{6.8}$$

When minimising/maximising a function subject to fixed outside conditions or constraints, Lagrange multipliers can be used as an approach to resolve such kinds of problems. Here, Lagrange multipliers are employed to compute the values which peers need to be selected to search, and the number of documents to be retrieved from each selected peer. Let

$$f(n_{p_1}, n_{p_2}, ..., n_{p_k}) = \sum_{1}^{k} EPrRi(n_{p_k}), \tag{6.9}$$

denote the objective function, and let

$$g(n_{p_1}, n_{p_2}, ..., n_{p_k}) = n_{p_1} + n_{p_2} + ... + n_{p_k}. \tag{6.10}$$

denote the constraint function. At the minimum, $\bigtriangledown_{n_{p_1}, n_{p_2}, ..., n_{p_k}} f = \lambda \bigtriangledown_{n_{p_1}, n_{p_2}, ..., n_{p_k}} g$, λ are the Lagrange multipliers. Then, in order to locate the minimum point on the set M, the $k+1$ Lagrange multipliers equations need to be resolved, which are

$$\begin{gathered}
\frac{\partial f(n_{p_1}, n_{p_2}, ..., n_{p_k}, \lambda)}{\partial n_{p_1}} = 0, \\
\vdots \\
\frac{\partial f(n_{p_1}, n_{p_2}, ..., n_{p_k}, \lambda)}{\partial n_{p_k}} = 0, \\
\frac{\partial f(n_{p_1}, n_{p_2}, ..., n_{p_k}, \lambda)}{\partial \lambda} = 0.
\end{gathered} \tag{6.11}$$

The values of $n_{p_1}, n_{p_2}, ..., n_{p_k}$ can be computed, indicating which peer to search (when $n_{p_k} \neq 0$), and the number of documents to be retrieved from each peer.

The next section will describe the method of estimating the number of relevant documents in the results set before query processing.

6.3 Estimating the Number of Relevant Documents in the Results Set

The number of relevant documents in the results set is needed in the precision-risk model. To address this problem, the methodology of the decision-theoretic framework (DTF) [Fuh99] is adapted to estimate the number of relevant documents in the results set. In DTF, different methods have been proposed to estimate the er_{p_k} value retrieved from peer p_k [Fuh99, NF03]. However, this chapter only focuses on the DTF-rp approach (rp is the initial of recall-precision) for two reasons, which are as follows: *i)* each peer has its own performance curve in terms of recall and precision; *ii)* the evaluation of IR systems typically use recall-precision curves, such as 11-point interpolated recall vs. average precision. The remainder of this section reviews the methodology used in [NF03].

According to the methodology of DTF-rp, four steps are needed to estimate the relevant number of documents er_{p_k} in the results set n_{p_k} of peer p_k for a given query, which are as follows:

1. A relevance-based peer score is computed for a given query.
2. A mapping function is employed to transform the relevance-based peer score into the possibility of relevance for peer p_k.
3. By using the possibility of relevance for peer p_k to estimate the total number of relevant documents in peer p_k for a given query.
4. Then, the estimated total number of relevant documents in peer p_k is used with the peer's own performance curve (i.e., a recall-precision function) to estimate the number of relevant documents er_{p_k} in the results set n_{p_k} of peer p_k.

In the first step, to compute the relevance-based peer score for a given query $R(p_k, q_i)$, the proposed approach in Section 5.2.2 (i.e., Equation 5.5) is applied in DTF-rp. Once the relevance-based peer score is obtained, DTF-rp employs a

mapping function to transform $R(p_k, q_i)$ into the possibility of relevance $Pr(rel|q_i, p_k)$ in the second step. This is given by:

$$f^{'} : [0,1] \mapsto [0.1], f^{'}(R(p_k, q_i)) \approx Pr(rel|q_i, p_k), \tag{6.12}$$

In DTF-rp [NF03], two mapping functions have been proposed, which are linear mapping and logistic mapping. By using these mapping functions, the peer score can be transformed into the probability of relevance for peer p_k. A linear mapping function is used to begin with. If it is assumed that the number of relevant documents in a peer is proportional to the relevance-based peer score, then an affine linear function is obtained, which is as follows:

$$f^{'} : [0,1] \mapsto [0.1], f^{'}(R(p_k, q_i)) = c_0 + c_1 * R(p_k, q_i), \tag{6.13}$$

where c_0 and c_1 are query-specific. Since no relevance feedback is available before query processing, query-independent constants c_0 and c_1 are used instead [NF03].

As an alternative to the linear mapping function, a logistic mapping function can be employed, which is as follows:

$$f^{'} : [0,1] \mapsto [0.1], f^{'}(R(p_k, q_i)) = \frac{exp(b_0 + b_1 * R(p_k, q_i))}{1 + exp(b_0 + b_1 * R(p_k, q_i))}, \tag{6.14}$$

where b_0 and b_1 in the logistic mapping function are also query-specific, which is similar to the parameters c_0 and c_1 in the linear mapping function.

In the third step, with probability $Pr(rel|q_i, p_k)$, the estimated total number of relevant documents $E(rel|q_i, p_k)$ in peer p_k for a given query q_i is computed by:

$$E(rel|q_i, p_k) = |nd_{p_k}| * Pr(rel|q_i, p_k), \tag{6.15}$$

The last step is to estimate the number of relevant documents er_{p_k} in the results set n_{p_k} returned from peer p_k. The recall-precision function of peer p_k is needed to estimated er_{p_k}. This assumes that the recall-precision function of a peer is equal for all of the queries, and can be provided by the peer p_k or learnt from the previous query results. The recall-precision curves are linearly decreasing functions with different shapes, shown in Figure 6.1 [NF03, Fuh99]. DTF-rp models different recall-precision shapes by means of two functions:

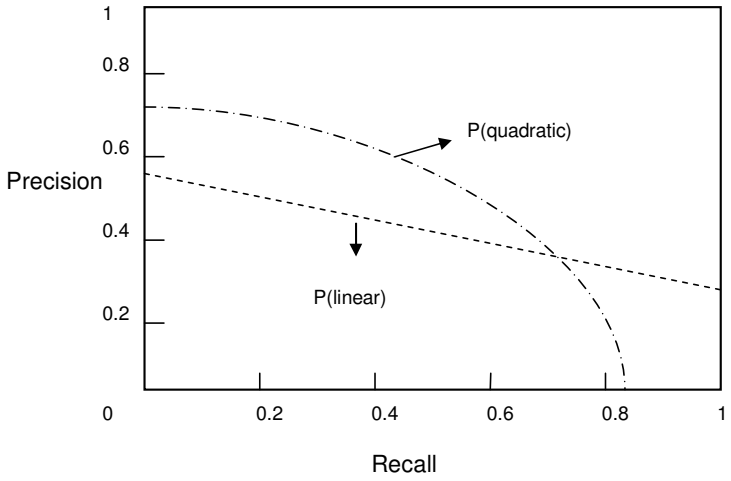


Figure 6.1: Different recall-precision functions [NF03, Fuh99]

$$P_{linear} : [0,1] \mapsto [0.1],\ P_{linear}(R) = l_0 - l_1 * R, \tag{6.16}$$

where $P_{linear}(R)$ is a linearly decreasing recall-precision function with two degrees of freedom, l_0 and l_1 are constants.

$$P_{quadratic} : [0,1] \mapsto [0.1],\ P_{quadratic}(R) = q_0 + q_1 * R - q_2 * R^2, \tag{6.17}$$

where $P_{quadratic}$ is a quadratic recall-precision function with three degrees of freedom, q_0, q_1 and q_2 are constants.

The estimated precision of peer p_k can be defined as $EPrecision_{p_k} = er_{P_k}/n_{p_k}$ and the estimated recall of peer p_k can be defined as $ERecall_{p_k} = er_{p_k}/E(rel|q_i, p_k)$. Then, the following equations can be arrived at for both the linear recall-precision function and quadratic recall-precision function. For the linear one, the equation is:

$$EPrecision_{p_k} = \frac{er_{p_k}}{n_{p_k}} = Pre(ERecall_{p_k}) = l_0 - \frac{l_1 * er_{p_k}}{E(rel|q_i, p_k)}, \tag{6.18}$$

For quadratic ones, the equation is:

$$EPrecision_{p_k} = \frac{er_{p_k}}{n_{p_k}} = Pre(ERecall_{p_k}) = q_0 + \frac{q_1 * er_{p_k}}{E(rel|q_i, p_k)} - \frac{q_2 * er_{p_k}^2}{E(rel|q_i, p_k)^2}. \quad (6.19)$$

The estimated number of relevant documents er_{p_k} in the results set n_{p_k} returned from peer p_k can be obtained by solving these equations.

6.4 Implementation Strategies

To implement the *PrRi* peer selection model in uncooperative P2PIR environments and a structured P2P network, decentralised system architectures and data management protocols are proposed in this section. The implementation strategies of the *PrRi* peer selection model should consider the unique characteristics of structured P2P networks (e.g., routing protocols). A Chord-based P2P network is used as the basic architecture to implement the *PrRi* peer selection model, which is the same network as in the previous chapters.

6.4.1 System Architecture

Since Chord is a completely decentralised P2P network, and there is no central server and super peer available, the data for the *PrRi* peer selection model should be maintained by each peer in a distributed manner. The system architecture of each peer for the *PrRi* peer selection model in structured P2P networks will be designed to fulfil the following functions: *(i)* storing and collecting the peer risk values and statistical information; *(ii)* minimising *EPrRi* to compute a clear cut-off for which peers to search and the number of documents to be retrieved from each of the selected peers. Figure 6.2 shows the system architecture of the precision-risk peer selection model, which consists of five components, namely, *Risk Manager, Ranker, Retrieval Quality Estimator, Optimal Peer Selector* and *Data Locator*.

The risk manager is responsible for storing peer risk values, and computing the risk values of relevant documents in that peer for a given query. It has two parts, namely, *Peer Risk Evaluation* and *Risk Data Repository*. The peer risk evaluation component evaluates the risk values of relevant documents in the peer p_k for the given query q_i. The risk data repository stores a portion of the global

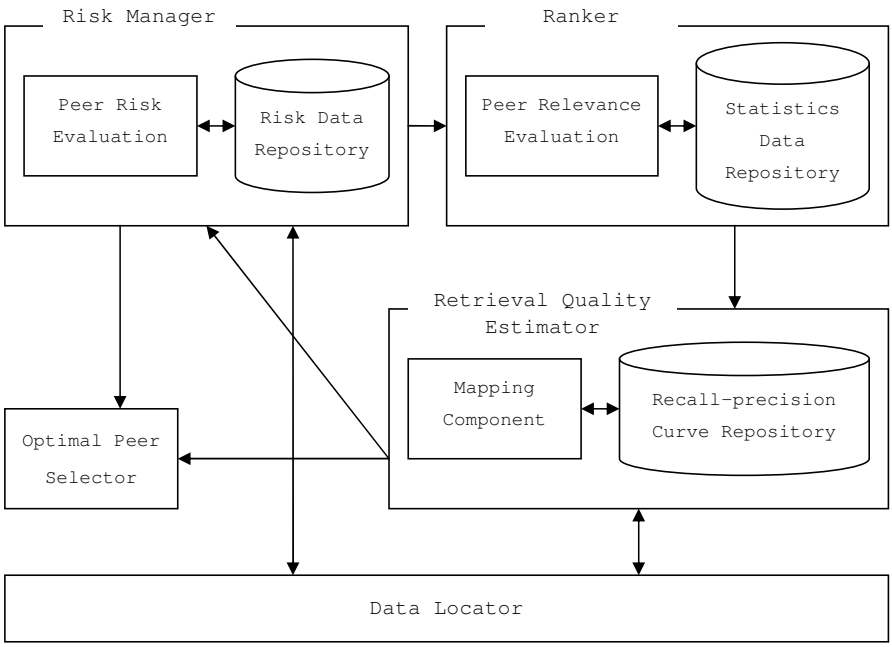


Figure 6.2: System architecture of the $PrRi$ peer selection model

risk values of peers in the network. The information stored in the risk data repository contains peer-id, ID_{p_k}; the risk value of peer p_k, $Risk_{p_k}$; and the peer IP address, *Location*. One example is shown in Table 6.1. The peer-id is the hash value of converting the peer IP address to the numeric key using the SHA-1 hash function.

Table 6.1: Risk Data Repository

ID_{p_k}	$Risk_{p_k}$	*Location*
p_{34}	0.6	124.12.35.25
p_{15}	0.6	196.126.1.3
p_{79}	0.3	96.12.178.3
p_{47}	0.6	84.13.256.6

The ranker employs Equation 5.5 to compute the relevance-based peer score for a given query. This consists of two components: *Peer Relevance Evaluation* and *Statistics Data Repository*. The peer relevance evaluation component is responsible for computing the relevance-based peer scores for a given query. The statistics data repository is a small database storing a portion of the global terms and corresponding statistics, such as: term-id, ID_{w_l}; peer-id, $Owners\ List$; document frequency, $df_{(q_i,p_k)}$ and the number of documents in peer p_k, nd_{p_k}. An

example is shown in Table 6.2. Similar to the peer-id, the term-id is the hash value of converting the term to the respective numeric key by using the SHA-1 hash function.

Table 6.2: Statistics Data Repository

ID_{w_l}	$Owners\ List$	$df_{(q_i,p_k)}$	nd_{p_k}
w_1	p_{21}	13	583
w_2	p_1	5	1240
	p_{5021}	18	423
	p_{1468}	5	641
w_3	p_{79}	23	2315

The retrieval quality estimator mainly focuses on estimating the number of relevant documents er_{p_k} in the results set n_{p_k} of peer p_k for a given query. It consists of a *mapping component* and a *recall-precision curve repository*. The mapping component is responsible for mapping the relevance-based peer score into the number of relevant documents in peer p_k, which can either employ the linear mapping function or the logistic one. The recall-precision curve repository stores a portion of the global recall-precision curves of peers in the network. The information stored in the recall-precision curve repository contains peer-id, ID_{p_k} and peer recall-precision curve, $P_{p_k}(R)$. An example is shown in Table 6.3.

The optimal peer selector stores the $PrRi$ peer selection model, which determines which peers to search and the particular number of documents to be retrieved from each of the selected peers. The data locator is responsible for publishing, accessing and updating data in P2P networks. In this section, the data locator employs the Chord routing protocols [SMK+01].

6.4.2 Data Management Protocols

Since the $PrRi$ peer selection model is designed for implementation in Chord structured P2P networks, the proposed data management protocols should extend the existing data management protocols of Chord in the context of term statistics and risk data routing. By extending the resource index manager of Chord with peer content statistics and risk information, the proposed protocol consists of the following phases: *(i) join and publish*, *(ii) lookup and select*, and *(iii) update*.

Table 6.3: Recall-Precision Curve Repository

ID_{p_k}	$P_{p_k}(R)$
p_{34}	$P_{p_{34}}(R) = 0.5 - 0.7 * R$
p_{15}	$P_{p_{15}}(R) = 0.4 + 0.8 * R - 0.6 * R^2$
p_{79}	$P_{p_{79}}(R) = 0.8 - 0.4 * R$
p_{47}	$P_{p_{47}}(R) = 0.6 - 0.8 * R$

6.4.2.1 Join and Publish

As a new peer joins the network for sharing documents, three different types of messages are simultaneously issued by the risk manager, ranker and retrieval quality estimator. The ranker should extract the peer statistical information and send messages $PUBLISH(ID_{w_l}, p_k, nd_{p_k}, df_{(q_i,p_k)})$ for each distinct term in peer p_k to the peers responsible for the term key ID_{w_l}. Before issuing the messages, the data locater converts the terms and peer IP addresses to the numeric keys by using the SHA-1 hash function. The peer responsible for the key ID_{w_l} adds the peer-id, $df_{(w_l,p_k)}$ and nd_{p_k} to the statistics data repository. In the meantime, the risk manager sends a $PUBLISH(ID_{p_k}, Risk_{p_k}, location)$ message from the peer p_k to the peer responsible for storing its risk value. The peer risk value is the reciprocal aggregation of the document reputation values in that peer. The process of collecting the reputation values of documents is described in the implementation strategies of the proposed trust-aware P2PIR system in Section 5.3.2. Moreover, the retrieval quality estimator sends a message $PUBLISH(ID_{p_k}, P_{p_k}(R))$ to the peer responsible for the recall-precision curve of peer p_k.

6.4.2.2 Lookup and Select

When the statistical information, risk values and recall-precision curves of peers have been stored in the network, they are ready for the $PrRi$ peer selection model. When a user submits a query to the $PrRi$ peer selection model, the model obtains the required data from the network. The lookup phase of retrieving the required data consists of the following three steps: *(i)* in order to compute relevance-based peer scores for a given query, the query terms are converted to the numeric keys by using the SHA-1 hash function. Then, the data locator forwards the lookup messages to the peers responsible for those keys by the Chord routing protocols. In response, the peers responsible for the keys return a list of peers containing the query terms and corresponding statistics. When

the ranker receives the statistical information, the relevance-based peer scores are computed; *(ii)* the relevance-based peer scores are forwarded to the retrieval quality estimator, and then a linear or logistical mapping function is employed to estimate the number of relevant documents in the peer p_k; *(iii)* after obtaining the peers list from the first step, the data locator uses ID_{p_k} in $Owners\ List$ as arguments to locate the peers which store the recall-precision curves of peers. Once the retrieval quality estimator obtains the recall-precision functions $P_{p_k}(R)$, the estimated number of relevant documents er_{p_k} in peer p_k can be calculated and forwarded to the optimal peer selector and risk manager, respectively. The risk manager then uses the peers list obtained from the first step to locate the risk values of peers. Having received the estimated number of relevant documents in the result of peer p_k for the given query, the risk manager calculates the estimated risk value of the results set retrieved from peer p_k, and then forwards the value to the optimal peer selector. Finally, the optimal peer selector employs the $PrRi$ peer selection model to compute a clear cut-off for which peers to search and the particular number of documents should be retrieved from each of the selected peers.

6.4.2.3 Update

When statistical data, recall-precision curves and risk values of peers changed, they need to be updated and stored in the network, and this information can be updated periodically. To update the peer risk values, users need to evaluate the document used and leave feedback for it. The processes of updating risk information and peer statistical information are the same as described in Section 5.3.2. After every search, a peer should obtain a new recall-precision curve, the new curve will be sent to the peer which is storing that peer's recall-precision curve, and the new recall-precision curve should be merged with the previous curve to generate an integrated one.

6.5 Differences Between DTF and $PrRi$

The previous sections describe the proposed precision-risk ($PrRi$) peer selection model including theory and implementation strategies. As stated in Section 6.1, $PrRi$ is inspired by the decision-theoretic framework (DTF) approach [Fuh99].

The differences between DTF and $PrRi$ are discussed in this section. The proposed $PrRi$ in this dissertation has three main differences compared to DTF and these are described as follows:

- DTF is a cost-based peer selection model, which calculates financial cost, computation time cost, and communication cost for sending documents [Fuh99], whereas $PrRi$ is a trust-based peer selection model, which computes the documents and peers trust values for given queries.
- The goal of DTF is to minimise cost in the result set [Fuh99], whereas the proposed $PrRi$ model does not only maximise the number of relevant documents, but also minimises the risk values of the retrieved documents in the result set.
- DTF is designed to be implemented in hierarchical networks which require super peers to collect information and control Information Retrieval [Fuh99], whereas the proposed $PrRi$ develops work on structured P2P networks where no super peers exist. A mechanism is needed in the proposed $PrRi$ peer selection model to organise peers in a cooperative manner so that the peer contents and reputational information can be collected, and the $PrRi$ values can be computed by each peer in the network.

6.6 Evaluation

The objective of this section is to evaluate the performance of the $PrRi$ peer selection model on retrieval accuracy and effectiveness of trust in protecting untrustworthy documents. The theoretical-based $PrRi$ peer selection model will be compared with the heuristic-based peer selection approach (as described in Section 5.2.2). In the remainder of this section, the experimental settings and methodologies are described in Section 6.5.1 and the initial experimental results are represented in Section 6.5.2.

6.6.1 Experimental Settings and Methodologies

In this section, a 1000-peer testbed (as described in Section 5.4.1) is employed to evaluate the performance of the $PrRi$ peer selection model in uncooperative P2P environments. The experimental settings in Section 5.4.1 are employed for

the evaluation of the heuristic-based peer selection method for comparison in this section. The experiments initially restrict the number of selected peers to search to be 10% of the network size and the top-10 ranked documents are retrieved from each selected peer. The TREC topics 451-550 short query set, and relevant and trustworthy judgement files (in Sections 4.4.1.2 and 4.4.1.3) are employed in this experiment. Since the $PrRi$ peer selection approach employs the same trust models as the proposed trust-aware P2PIR system in the previous chapters, the same experimental settings for trust in Section 5.4.1 are also used. Therefore, the percentage of malicious peers is initially set to 20%, randomly selected from the network. Good peers provide positive feedback of the trustworthy documents and negative feedback of the untrustworthy documents. On the other hand, malicious peers submit positive feedback for malicious peers, and submit negative feedback for the documents provided by the good peers. The credibility values of good peers and malicious peers are randomly set to 90% and 20% in the experiments.

One challenge to the evaluation of the $PrRi$ peer selection model is to acquire the parameters in mapping functions and peer recall-precision curves. A training data set of relevant documents is needed for parameters learning [NF04], and the pseudo-relevant judgement files are proposed for this purpose. The top-30 ranked documents are selected for each query retrieved by VSM in cooperative P2PIR environments as being the relevant documents for parameters learning. This is because: *(i)* the cut-off value top-30 are extensively used in the Information Retrieval literature [BYRN99], such as the average precision at given document cut-off values; *(ii)* for the same query, the retrieval performance of cooperative P2PIR is better than that of uncooperative P2PIR, which can be observed by comparing Figures 4.2 and 5.2. This phenomenon is same as in cases of Web search and distributed information retrieval in traditional IR; *(iii)* VSM provides the best overall retrieval performance of other approaches in cooperative P2PIR environments (as shown in Figure 4.2 (b)). Considering these three reasons, the top-30 ranked documents retrieved by VSM in cooperative P2PIR are the best candidates for a training data set for parameters learning. Since DTF-rp is applied to estimate the number of relevant documents in the results set, its methodologies for learning the parameters in mapping functions and recall-precision curves [NF04] are employed in this section. In the process of parameters learning, the nonlinear least-squares (NLLS) Marquardt-levenberg algorithm is employed [AFS89]. The Gnuplot implementation of the nonlinear

least-squares (NLLS) Marquardt-levenberg algorithm and the training data set are used to explore the parameters in the mapping functions and the peer recall-precision curves. Although, there are four types of parameters in DTF-rp, namely, *linear relevant mapping functions + linear recall-precision curves, linear relevant mapping functions + quadratic recall-precision curves, logistic relevant mapping functions + linear recall-precision curves,* and *logistic relevant mapping functions + quadratic recall-precision curves*, for simplicity, the linear relevant mapping functions + linear recall-precision curves are implemented in this chapter. The others can be done in the further evaluations if it is necessary. The $PrRi$ peer selection model is implemented in Matlab and Java. In the experiments, both the $PrRi$ peer selection model and the heuristic-based peer selection approach select 1000 documents for comparisons.

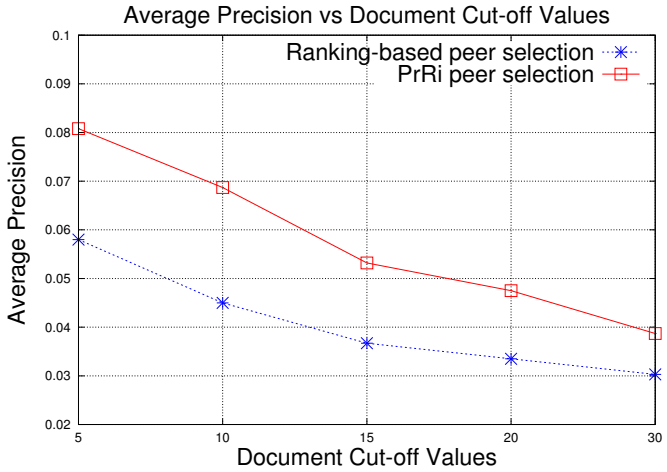
6.6.2 Experimental Results

This section focuses on the experimental results of the $PrRi$ peer selection on retrieval accuracy and effectiveness of trust, compared to the heuristic-based (i.e., ranking-based in Figure 6.3) peer selection approach. Two sections are devoted to the experimental results with regard to evaluations: *(i)* the retrieval accuracy of the two peer selection models; *(ii)* the effectiveness of the two peer selection approaches in protecting untrustworthy documents in the results set. Figure 6.3 shows the experimental results of retrieval accuracy of the two peer selection approaches for the TREC topics 451-550 short query set in the 1000 peer-sized network. Figure 6.3 (a)-(b) depicts the results of two evaluation metrics, which are (a) the average precision vs document cut-off values, and (b) 11-point interpolated recall vs. average precision. The higher the average precision in both figures, the better the retrieval accuracy that the peer selection approach can achieve.

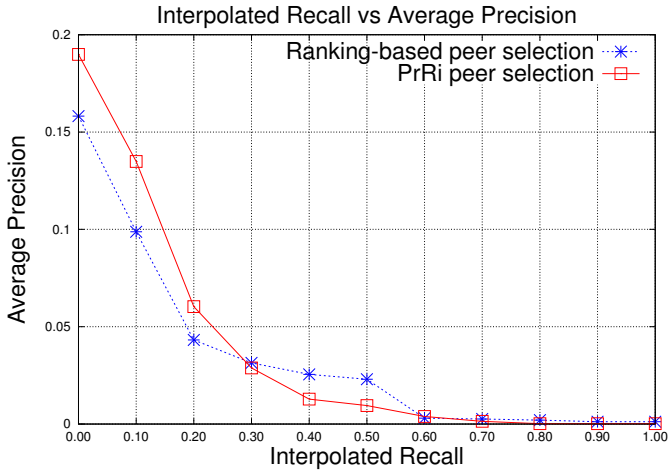
6.6.2.1 Retrieval Accuracy of the Two Peer Selection Approaches

This experiment evaluates the $PrRi$ peer selection model, compared to the heuristic-based peer selection approach (as described in Section 5.2.2) on retrieval accuracy.

As can be seen from Figure 6.3 (a), the $PrRi$ peer selection model outperforms the heuristic-based peer selection approach and the differences between the two curves are significant, indicating that $PrRi$ peer selection can favour user satisfaction much more than the heuristic-based approach. In Figure 6.3 (b), when



(a) average precision vs document cut-off values



(b) average precision vs interpolated recall

Figure 6.3: Retrieval accuracy of the two peer selection approaches for the TREC 451-550 short query set in the 1000 peer-sized network.

the recall levels are between 0 and 0.3, $PrRi$ can achieve a better retrieval accuracy than the heuristic-based peer selection. This demonstrates that, in the lower recall levels, the $PrRi$ peer selection model can uses fewer numbers of documents to achieve the same precision, compared to the heuristic-based peer selection. For example, there are a total of 200 relevant documents for a query in the network, when the recall level is 0.1, which means that 20 relevant documents have been seen. Since the precision of the $PrRi$ peer selection is higher than that of the heuristic-based peer selection, the $PrRi$ peer selection may retrieve 50 documents to achieve the recall level 0.1, but the heuristic-based peer selection may need 150 documents. Lower recall levels always indicate top-ranked documents in the results set. Therefore, this shows that the $PrRi$ peer selection model can obtain the better retrieval accuracy in the top-ranked results list, which is same as the result and conclusion in Figure 6.3 (a). When the recall levels are between 0.3 and 0.6, the heuristic-based peer selection can provide a better retrieval performance than the $PrRi$ peer selection, and after the recall level 0.6 both approaches are tight. Since it is not easy to distinguish between which approach can achieve a better overall retrieval performance in Figure 5.3 (b), the average precision and recall are employed. The $PrRi$ peer selection and the heuristic-based peer selection are 0.0328 and 0.0276, respectively. The recalls for both peer selection approaches are 0.0036 for $PrRi$, and 0.0025 for the heuristic-based one. Therefore, the $PrRi$ peer selection can yield a better overall retrieval performance than the heuristic-based peer selection.

In summary, on a theoretically founded basis, $PrRi$ can achieve the better retrieval accuracy on both user satisfaction and the overall retrieval performance, compared to the heuristic-based peer selection strategy, which selects a fixed and pre-determined number of peers and an equal number of documents from each of these selected peers. When users specify the number of documents to be retrieved, in order to achieve the maximum number of relevant documents in the results set, the $PrRi$ peer selection model computes a clear cut-off for which peers to search and the number of document to be retrieved from each of these selected peers. The relevant and trustworthy peer returns the specific number of documents based on how many estimated relevant documents are in it for the given query. This makes the $PrRi$ peer selection model achieve a better retrieval performance. This also indicates that it is not appropriate to use a fixed and pre-determined threshold to decide the number of peers to search, and an equal

number of documents to retrieve from each selected peer for any queries.

6.6.2.2 Effectiveness of Trust of the Two Peer Selection Approaches

This experiment explores the effectiveness of trust in protecting untrustworthy documents by the two peer selection approaches. Figure 6.4 shows the experimental results of the percentage of untrustworthy documents appearing in the top-ranked results list for the TREC 451-550 short query set, which are generated by two peer selection approaches.

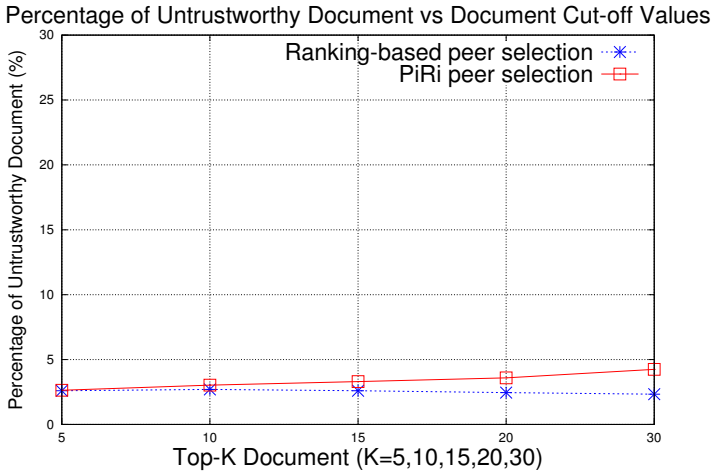


Figure 6.4: Effectiveness of the two peer selection approaches to protect untrustworthy documents for the TREC 451-550 short query set in the 1000 peer-sized network.

It can be observed from the figure, that the heuristic-based (i.e., ranking-based in Figure 6.4) approach is little bit better than $PrRi$ peer selection. The reason for this is that, in order to achieve the maximum number of relevant documents in the results set, the $PrRi$ peer selection model contacts more peers to retrieve relevant documents for a given query, and some peers are untrustworthy. This reduces the effectiveness of trust. To address this problem, the $PrRi$ model may be optimised by setting different conditions, for example, achieving the maximum retrieval accuracy after minimising the risk value of the results set.

6.7 A Variant

In this section, a variant of the $PrRi$ peer selection model is discussed. The selection criterion used so far is to retrieve the maximum number of relevant documents with the minimum risk of reviewing or downloading untrustworthy documents in the results set. Now, a simpler case is introduced whereby users only want to minimise the risk values of documents in the results set subject to the number of documents which have been seen. To address this problem, the $PrRi$ model can be simplified by removing the estimated precision part and modifying the risk part. The risk value of the results set n_{p_k} retrieved from peer p_k should be the sum of the risk value of every single document. It is assumed that the risk value of the document is equally distributed in the peer, and therefore, the risk value of documents retrieved from peer p_k is proportional to the peer risk value. Thus, the estimated risk value of the results set retrieved from peer p_k is estimated by:

$$ERisk_{p_k}(n_{p_k}) \approx Risk_{p_k} * \frac{n_{p_k}}{nd_{p_k}}, \tag{6.20}$$

where $ERisk_{p_k}(n_{p_k})$ is the estimated risk value of peer p_k for a given query q_i, n_{p_k} is the number of documents retrieved from peer p_k, and the nd_{p_k} is the number of documents in peer p_k.

Assuming that there are k different peers in the network, the results set is the aggregation of the documents retrieved from those peers and a corresponding vector $n = (n_{p_1}, n_{p_2}, ..., n_{p_k})$ is the number of documents to be retrieved from each of them, then:

$$ERisk(n) \approx \sum_{1}^{k} Risk_{p_k}(n_{p_k}). \tag{6.21}$$

To

$$minimise\ ERisk(n), \tag{6.22}$$

subject to

$$M = \{(n_{p_1}, n_{p_2}, ..., n_{p_k}) : n_{p_1} + n_{p_2} + n_{p_3} + ... + n_{p_k} = n\}. \tag{6.23}$$

Lagrange multipliers can be employed to compute the values by determining

which peers need to be searched and the number of documents retrieved from these selected peers.

6.8 Summary

Peer selection is an important problem in uncooperative P2PIR. In contrast to the heuristic-based peer selection approach, a theoretical-based peer selection model $PrRi$ is proposed in this chapter. The proposed $PrRi$ precision-risk model can compute a clear cut-off for which peers to select and the number of documents needed to be retrieved from each of the selected peers for a given query. These values can be computed by maximising the number of relevant documents and minimising the risk values of documents in the results set. Moreover, a system architecture and data management protocols are designed to implement the $PrRi$ peer selection approach in structured P2P networks. The experimental results show that $PrRi$ peer selection can achieve the better retrieval performance on both user satisfaction and the overall retrieval performance, compared to the heuristic-based peer selection strategy.

Chapter 7

An Analysis of the Trade-off Study between Relevance and Trustworthiness

7.1 Introduction

In the previous chapters, document ranking and peer selection algorithms were developed for the proposed trust-aware P2PIR systems in cooperative and uncooperative environments. The purpose of these algorithms was two fold: *(i)* to retrieve more relevant documents; and at the same time *(ii)* to prevent untrustworthy documents in the results set. However, both objectives often in conflict with each other. For a given query, some documents may be relevant but untrustworthy, and some may be trustworthy but irrelevant. If a bigger weight is given to relevance, more untrustworthy documents may be retrieved. Conversely, if weighting trustworthiness more, more irrelevant documents may be involved. Therefore, the proposed document ranking algorithm and peer selection algorithms may lose one objective, such as retrieval accuracy or effectiveness of trust, in return for meeting the other. In Chapters 4, 5 and 6, to simplify the problems, the relative weights between relevance and trustworthiness were assumed to be equal in the document ranking algorithm (i.e., Equation 4.3) and peer selection algorithms (i.e., Equations 5.1 and 6.1). Actually, there is a tradeoff between relevance and trustworthiness in these algorithms. Then, how to find the trade-off surface between relevance and trustworthiness in these algorithms, and how to select a preferred weight from a set of compromising weights on the trade-off

surface are interesting problems for the proposed trust-aware P2PIR systems.

To address the above problems, the major contributions of this chapter are:

- A heuristic-based search process to find the near optimal points of relative weights. The near optimal points (i.e., a weight with the corresponding values of relevance and trustworthiness) will be visualised to represent a trade-off surface. Any two near optimal points on the trade-off surface have the same property, in which from one solution to another, the algorithm (e.g., the document-ranking algorithm Equation 4.3) has to sacrifice either retrieval accuracy or the effectiveness of trust in order to gain the benefit in another objective.

- A ranking approach to sort the different near optimal points by the ratios of how much the changing percentage in one objective relates to the changing percentage of another one. This approach may help users to make a decision as to which weight to choose. It should be noted that the selection of a preferred solution from a set of compromising weights is not an easy question to answer. This contains some high-level information such as the decision maker's preferences. Different preferred solutions should be chosen according to various situations.

In the remainder of this chapter, Section 7.2 introduces the definition of a multi-objective optimisation problem and general solutions. Section 7.3 describes a proposed heuristic-based search process to find the near optimal points of the relative weights. Section 7.4 represents a case study, which employs a heuristic-based approach to find the near optimal points and a trade-off surface for the document ranking algorithm of the proposed trust-aware P2PIR system in cooperative P2PIR environments. The near optimal points and the trade-off surface are also analysed in this section. Section 7.5 presents a ranking approach to help users to select a preferred solution from the near optimal points on the trade-off surface. This chapter is summarised in Section 7.6.

7.2 Multi-objective Optimisation

In the proposed trust-aware P2PIR system, retrieval accuracy and the effectiveness of trust rely on the relative weight selection. Basically, users may prefer the

proposed trust-aware P2PIR system to achieve the *maximum retrieval accuracy* and the *maximum effectiveness of trust*. However, these both in conflict with each other. Such a problem is generally known as a multi-objective optimisation problem, which can minimise or maximise more than one objective at the same time [Ste86]. A trade-off surface is generated with a set of optimal solutions, and then the decision-maker chooses the most preferred solution as the final answer [BDMS08]. The solution process of the multi-objective optimisation problem provided the inspiration to resolve the problems proposed in this chapter. The basic definitions and general solutions of the multi-objective optimisation problem are introduced in the following section.

7.2.1 Background of Multi-objective Optimisation

In the real world, a number of decisions to resolve problems involve more than one conflicting objective function, which need to be considered simultaneously. For example, in the design of a motor engine, an engineer may wish to maximise the strength of an engine component and minimize the financial cost. Unlike a single objective optimisation problem which only has a unique solution, the solution of a multi-objective optimisation problem is a set of Pareto-optimal points, which needs further analysis to achieve a single preferred solution by the decision maker [Ste86, CS03]. In the multi-objective optimisation problem, the objective functions are to be either minimised or maximised, and in mathematical terms, the multi-objective optimisation problem is defined by [Ste86, CS03, BDMS08]:

Definition 1

$$Minimise/Maximise\ f(x_i) = [f_1(x_i), f_2(x_i), ..., f_n(x_i)]^T\ n = 1, 2, ..., N,$$

subject to:

$$g_j(x_i) \leq 0,\ j = 1, 2, ..., J,$$

$$h_k(x_i) = 0\ k = 1, 2, ..., K,$$

$$x_i \leq x \leq x_q,\ i = 1, 2, ..., m.$$

where, $f_n(x_i)$ is the $n-th$ objective function (or so-called optimisation criterion), $g(x_i)$ is unequal constraints and $h(x_i)$ is equal constraints. The set of vectors x_i is known as the decision variables.

Before giving an explanation of the Pareto-optimal points, the Pareto solution, Pareto point and non-dominated point are described. The Pareto solution is one of the solutions of a multi-objective problem, but it is not optimal. A Pareto point is a Pareto solution with the associated objective functional values. A set of Pareto points consist of the objective space for the solutions of the multi-objective problem. The non-dominated point is defined as follows:

Definition 2 *A Pareto solution x_1 is said to* dominate *the other Pareto solution x_2, if both of the following conditions are met:*

- *Pareto solution x_1 is no worse than x_2 in all objectives. Pareto solutions x_1 and x_2 are compared based on their objective function values.*
- *Pareto solution x_1 is strictly better than x_2 in at least one objective.*

Definition 2 can be used to compare any two Pareto points in the objective space to identify which one dominates the other. If there are a number of points which are not dominated by the others in the objective space, then these points are called *non-dominated points*. The solutions of a multi-objective optimisation problem are *Pareto-optimal points*, which are the locations of the corresponding *non-dominated* points (or so-called Pareto-optimal points) in the objective space [Ste86, CS03, BDMS08]. These Pareto-optimal points are often visualised to represent a trade-off surface, which involves sacrificing the quality of one objective in return for gaining the quality of other objectives [CS03, BDMS08]. After a set of Pareto-optimal points is found, the decision-maker will consider the different optimal points and select a single preferred solution as the final answer.

7.2.2 Optimisation Methods

The multi-objective optimisation problem has been extensively studied for several decades. The idea of resolving multi-objective optimisation is to find a set of Pareto-optimal points while simultaneously considering a number of objectives. There are a number of surveys on the methods developed for multi-objective optimisation [MA04, CS03, BDMS08]. For example, in [CS03], the methods are classified into three categories, based on the involvement of a decision-maker in the multi-objective optimisation solution process. These are *a priori, progressive* and *a posteriori* methods.

In the a priori method, decision makers should firstly specify their preference, and then the optimisation solution process is to find one Pareto-optimal point according to that preference. The advantage of the a priori method is that the search process of Pareto-optimal points is only performed once. The disadvantage is that the specified preference may be misleading and the end optimisation solution may not satisfy the decision maker's request. Alternatively, the progressive method is an iterative solution, whereby the optimal search process is repeated until the decision-maker finds the most preferred solution. In the progressive method, the decision-maker needs to become involved in the optimisation solution process by specifying the preferred information. The advantage of this method is that the decision-maker can direct the solution process and only a part of Pareto-optimal points need to be found. The disadvantage of this method is that the decision-maker has to be involved in every single step, which may take a long time. The third classification is the a posteriori method, which produces a full set of Pareto-optimal points first, and then the decision-maker can select the most preferred solutions by comparing various solutions. The advantage of this method is that the decision-maker has a global view of the full set of Pareto-optimal points, which can be comprehensive. The disadvantage is that the a posteriori method is computationally expensive and time consuming, especially if there are more than three objectives. Different methods have their own strengths and weaknesses, which is why different approaches are needed. A large number of mathematical approaches have been proposed to find the Pareto-optimal points, such as aggregate objective functions (AOF), e-constrained functions and evolutionary algorithms.

7.3 A Heuristic-Based Search Process for Near Optimal Solutions

To find the trade-off surface for relevance and trustworthiness in the proposed document ranking and peer selection algorithms, the Pareto-optimal points should be located first. Typically, to obtain Pareto-optimal points, the objective functions need to be formalised, and then a mathematical approach (e.g., fuzzy method, e-constrained, and evaluation algorithm) can be used to find the Pareto-optimal solutions. However, in the proposed trust-aware P2PIR system, it is hard to present relevance and trustworthiness (i.e., objective functions) with respect to

the relative weight (i.e., decision variable) in mathematical expressions. This is because: *(i)* the ranked results lists are used for comparison with the relevant and trustworthy judgement files to evaluate the retrieval accuracy and the effectiveness of trust for the proposed document ranking and peer selection algorithms (as described in Sections 4.4.2, 5.4.1 and 6.5.1). Therefore, the rank of a document or peer is an essential factor for the evaluation of relevance and trustworthiness. However, the relative weight does not have a straightforward relationship with the ranks; *(ii)* a document with a high rank does not mean that it is relevant to a given query. It only indicates that the document has a greater possibility of being a relevant document for a given query. Actually, the relevance and trustworthiness of a document for a given query is decided by users. Since people may have very different backgrounds and knowledge of relevance and trustworthiness, it is hard to express a subjective-based judgement as a mathematical function. With both of these difficulties, it is hard, or even impossible, to represent the retrieval accuracy and effectiveness of trust with respect to the relative weight in any mathematical function. Thus, mathematical methods cannot be employed to find the Pareto-optimal points for the proposed document ranking and peer selection algorithms.

To address this problem, a heuristic-based search process is proposed to find a set of solutions approximating the Pareto-optimal points. These points are defined as *near optimal points* in this dissertation. In this chapter, the progressive approach of the multi-objective optimisation solution process is used to find the near optimal solutions. This is because: *(i)* the decision-maker's preferences are not known in advance, so the a priori approach is not an appropriate choice for this problem; *(ii)* since the values of the weights are infinite, it is extremely computationally expensive and time consuming to find the global near optimal solutions by a heuristic-based search process. Thus, the a posteriori method may not be a good option either. The progressive method could be a better choice for a heuristic-based search process because only a part of the global near optimal solutions are found. This can make a large saving on computational cost and timing. To find the near optimal points in the progressive approach, the proposed heuristic-based search process computes a set of Pareto points by changing the relative weights in the proposed document ranking or peer selection algorithms first. Then, the near optimal points are selected from the objective space based on Definition 2. The selected near optimal points are connected and represented

as a trade-off surface for the relative weight study.

In the heuristic-based search process, two evaluation metrics are selected as the value of relevance $APre_r$ (i.e., *average precision at given document cut-off values*), and the value of trustworthiness $APer_t$ (i.e., *average percentage of untrustworthy documents at given document cut-off values*). To compute the relevance and trustworthiness values with respect to the corresponding relative weight w in the proposed document ranking or peer selection algorithms, the heuristic-based search process employs the same evaluation methodologies as in Section 4.4.2 (for the proposed document ranking algorithm), and in Sections 5.4.1 and 6.5.1 (for the proposed peer selection algorithms). When setting a relative weight w in these algorithms, the corresponding $APre_r$ and $APer_t$ values are generated by the evaluation methodologies.

Algorithm 1 A heuristic-based search process for the near optimal points of relative weights

Initialising the relative weight w and searching range, e.g., set $w = 0.1$, range from 0 to 1 and step size 0.1
for $w = 0$ to 1 in step of 0.1 **do**
  Computing document scores by the proposed document ranking or peer selection algorithms;
  Comparing document scores and sorting documents by the decreasing order;
  Comparing top-k documents in the ranked results list with the standard relevant and trustworthy judgement files;
  Computing $APre_r$ and $APer_t$;
  Plotting the values of $APre_r$ and $APer_t$ on a plane with w;
end for
if the decision maker is satisfied with the Pareto solutions **then**
  the search process will terminate
else
  repeat
    the decision maker needs to specify the new w, searching range and step size;
    back to the **for** loop;
  until the decision maker is satisfied with the generated Pareto solutions
end if
Selecting the non-dominated points (near optimal points) in the objective space and drawing a trade-off surface.

The objective of the heuristic-based search process is to find a set of near optimal points by changing the weights in the proposed document ranking algorithm (i.e., Equation 4.1) and peer selection algorithms (i.e., Equations 5.1 and

6.1), which is given in Algorithm 1. The first step, in Algorithm 1 is to initialise the relative weight in the algorithm, as well as the searching range and step size. Since this approach is to find the near optimal points in the objective space, the precision of the near optimal points relies significantly on the precision of the initial value w and step size. These values are selected by the decision-maker. The second step is to make a *for* loop to find the Pareto solutions by computing a set of values of $APre_r$ and $APer_t$ with respect to the w values. When the Pareto solutions are generated, the decision-maker needs to analyse the results and decide whether the Pareto solutions are satisfied or not. The satisfaction of the Pareto solutions is subjective, since it is dependent on the decision-maker's preference. If the decision-maker is satisfied with the solutions, the search process should be stopped. Otherwise, the decision-maker needs to specify a new relative weight, searching range and step size, which are in the *repeat* loop in Algorithm 1. The new Pareto solutions are generated according to the new settings. The search process will be repeated until the Pareto solutions satisfy the decision-maker's requirements. Thereafter, all of the Pareto points in the objective space should be compared with each other to identify the non-dominated points. The non-dominated points refer to the near optimal points, and then these points can be visualised to represent a trade-off surface. The trade-off surface represents a set of compromising relative weights between relevance and trustworthiness in the proposed document ranking algorithm or the proposed peer-selection algorithms. The decision-maker can study the near optimal points on the trade-off surface, and then select a preferred solution as the final answer.

7.4 Case Study: A Document Ranking Algorithm of the Proposed Trust-Aware P2PIR System

In the previous section, the heuristic-based search process to find the near optimal points and trade-off surface is described. This section represents a case study, which uses the heuristic-based search process to find the near optimal points of relative weights and the trade-off surface for the document ranking algorithm of the proposed trust-aware P2PIR system in cooperative environments (as described in Chapter 4). Moreover, the generated near optimal value and trade-off

surface are analysed in this section.

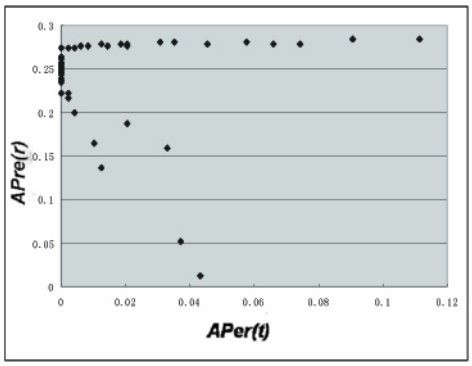
Since the cut-off values of 5, 10, 15, 20 and 30 documents in the ranked results list are extensively studied in the IR literature [BYRN99], the near optimal points and trade-off surfaces can be found for each of them. Equation 4.3 in Chapter 4 is the proposed document ranking algorithm for trust-aware P2PIR in cooperative P2PIR environments, with the relative weight w, the score $S(d_j, q_i, p_k)$ of the document d_j for a given query q_i provided by a peer p_k is given by

$$S(d_j, q_i, p_k) = \sqrt{w * R^2(d_j, q_i) + (1 - w) * T^2(d_j, q_i, p_k)}. \tag{7.1}$$

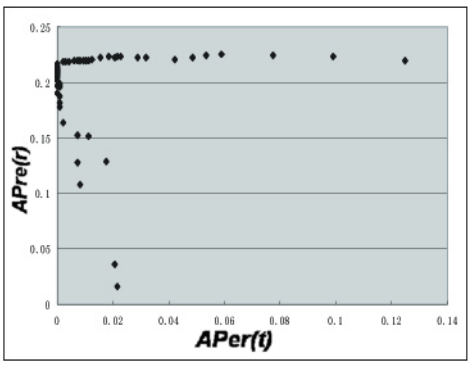
where $R(d_j, q_i)$ is the relevance degree between a document d_j and a query q_i, and $T(d_j, q_i, p_k)$ is the trust value of the document d_j provided by the peer p_k for the given query q_i, w is the relative weight between relevance and trustworthiness.

To compute the relevance value (i.e., $APre_r$) and trustworthiness value (i.e., $APer_t$) with regard to the relative weight w, a set of experiments needs to be set up. The same experimental settings are used as in Section 4.4. A 1000 peer testbed is selected with the TREC 451-550 short query set. Two types of peers are simulated in the network, which are good peers and malicious peers. The termination of the heuristic-based search process is assumed to find at least p (e.g., 50) different Pareto points so that the decision-maker may have a sufficient objective space with a number of Pareto points to study. The w is sampled for the variant searching ranges and step sizes. For example, w is from 0 to 1 in step of 0.1, or w is between 0.001 and 0.009 in step of 0.001. A set of Pareto points can be generated from the different w values by Equation 7.1.

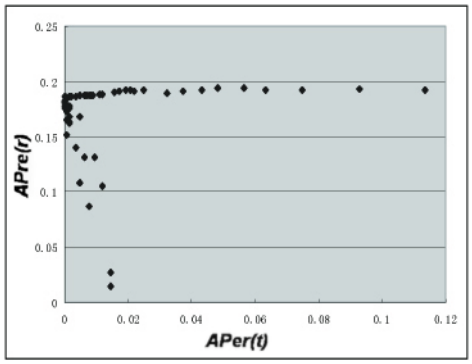
To start the heuristic-based search process for Equation 7.1 by Algorithm 1, w to 0.1 is firstly initialised and the searching range is from 0 to 1 in step of 0.1. It was found that the $APre_r$ and $APer_t$ values did not change when w was between 0.1 to 0.9. This indicates that the sensitive areas of w should be in 0 to 0.1 and 0.9 to 1. Thereafter, w was set to 0.01 with a searching range from 0.01 to 0.09 in step of 0.01, and is set to 0.91 with the searching range from 0.91 to 0.99 in step of 0.01. The new Pareto points can be generated according to the new w settings. After the stepwise searching, it was found that the sensitive ranges of w were between 0 and 0.003, and 0.9971 and 1, which could generate more than 50 different Pareto points for each of the top-k ranked documents (i.e., k=5,10,15,20 and 30). Then, these Pareto points were put into the different figures for each of the top-k ranked documents, as shown in Figure 7.1 (a)-(e). The figure depicts



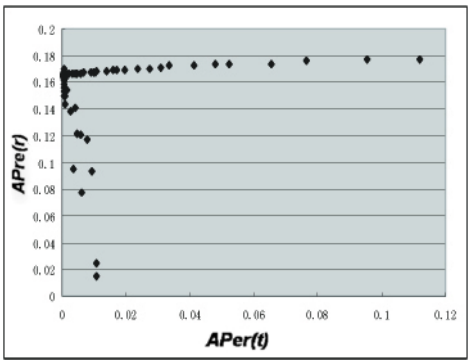
(a) top-5 ranked documents



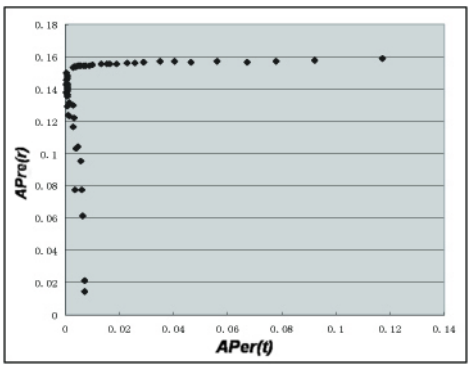
(b) top-10 ranked documents



(c) top-15 ranked documents



(d) top-20 ranked documents



(e) top-30 ranked documents

Figure 7.1: Pareto points of top-k ranked documents in the objective space for the TREC 451-550 short query set in the 1000 peer-sized network

the results of $APre_r$ and $APer_t$ with the corresponding weight w at the top-5 ranked documents (Figure 7.1 (a)), top-10 ranked documents (Figure 7.1 (b)), top-15 ranked documents (Figure 7.1 (c)), top-20 ranked documents (Figure 7.1 (d)), and top-30 ranked documents (Figure 7.1 (e)). In Figure 7.1, x-axis is $APer_t$, y-axis is $APre_r$, and the points in the figure are the relative weights w. For example, in Figure 7.1 (a), there is a Pareto point A, its $APer_t$ is 0.0907, $APre_r$ is 0.2845, and the corresponding w is 0.9999.

A pay-off table is generated according to the Pareto points in Figure 7.1 (a)-(e). The pay-off table is a form, which consists of the minimum and maximum $APre_r$ with the corresponding w, and the minimum and maximum $APer_t$ with the corresponding w for each of the top-k ranked documents in Figure 7.1. An example in Table 7.1 is, for the top-5 ranked documents, the minimum $APre_r$ value is 0.0124 when $w = 0$, the maximum $APre_r$ value is 0.2845 when $w = 1$. The pay-off table can provide an approximation of the complete Pareto solutions and the potential value ranges of $APre_r$ and $APer_t$ for analysis. The following observations can be made from Table 7.1: *(i)* when $APre_r$ achieves the minimum values in top-k ranked documents, w is always 0. This is as expected, because the document ranking algorithm Equation 7.1 only takes trustworthiness into consideration; *(ii)* when $APre_r$ achieves the maximum values in top-k ranked documents, most w is 1 except 0.9995 in top-10 ranked documents and 0.9997 in top-15 ranked documents. This is because when w is set to 1 in Equation 7.1, the document ranking algorithm does not calculate trustworthiness. Therefore, some untrustworthy but relevant documents for a given query can be ranked in the higher ranks in the results list.

Table 7.1: The pay-off table for the near optimal points of relative weights w in the objective space for the proposed document ranking algorithm

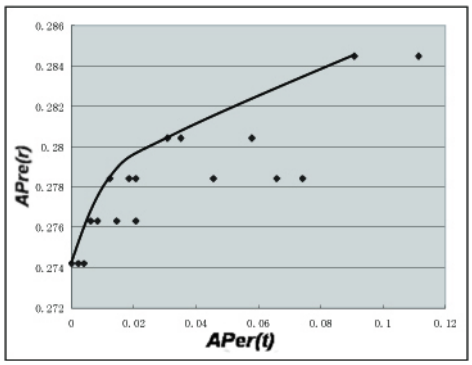
	minimum $APre_r$	w	maximum $APre_r$	w	minimum $APer_t$	w	maximum $APer_t$	w
Top 5	0.0124	0	0.2845	1	0	0.0007, 0.001∼0.003	0.1113	1
Top 10	0.0165	0	0.2258	0.9997	0	0.0013,0.0014 0.0016∼0.003	0.1247	1
Top 15	0.0144	0	0.1938	0.9995	0	0.0017,0.0019 0.0022,0.0024 0.0026, 0.0028∼0.003	0.1134	1
Top 20	0.0149	0	0.1768	1	0	0.0022,0.0024, 0.0029	0.1119	1
Top 30	0.0144	0	0.1588	1	0.0003	0.0029	0.1172	1

Since the Pareto points have been obtained and studied, the next step is to select the non-dominated points from the objective space for each of the top-k documents in Figure 7.1 (a)-(e). According to Definition 2, all of the points in Figure 7.1 can be compared with each other to identify which ones are not dominated by any of the other points. For example, in the top-5 plot of Figure 7.1, points (0,0.2742), (0.0062, 0.2763) and (0.0124,0.2784) are non-dominated points. These non-dominated points are connected to represent the trade-off surfaces for each of the top-k ranked documents, as shown in Figure 7.2. Then, Figure 7.2 is enlarged and some dominated points are removed to arrive at Figure 7.3. The lines in Figure 7.3 are the trade-off surfaces for the relative weights between relevance and trustworthiness in the proposed document ranking algorithm Equation 7.1 for each of the top-k ranked documents. In addition, a set of points in the trade-off surface is near optimal points, which are compromising weights for Equation 7.1.

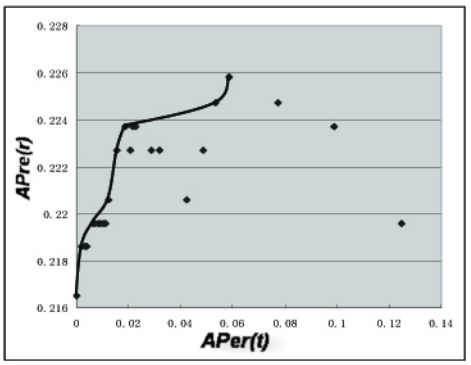
7.5 Decision Making

Since a set of near optimal points and a trade-off surface are found for the relative weight w in Equation 7.1, the obvious question which must arise for a decision-maker is how to select a solution from the near optimal points in the trade-off surface as the final answer. This is not a easy question, since the answer often involves some high-level information such as the decision-maker's preferences. For example, if the decision-maker is trustworthiness-orientated, according to the pay-off table (i.e., Table 7.1), the preferred solution should be w=0.003, in which $APer(t)$ can achieve the minimum values in all of the top-k ranked documents. Conversely, if the decision-maker is relevance-orientated, then the preferred relative weight may be 1 based on Table 7.1. However, if the decision-maker does not show any preferences for either relevance or trustworthiness, it is hard to know which solution may be chosen as the final answer. For example, in the trade-off surface of the top-5 ranked documents, the near optimal point (0.0907,0.2845) with w=0.9999 is the preferred solution or the near optimal point (0.0062, 0.2763) with w=0.9978 is the one. The problem is how to select one near optimal point from the trade-off surface when the decision-maker does not show any preferences between relevance and trustworthiness.

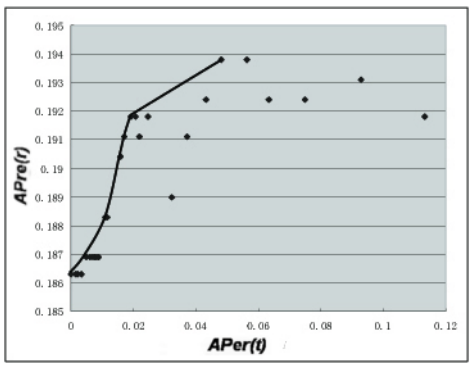
To address this problem, a ranking approach is proposed to help users to make



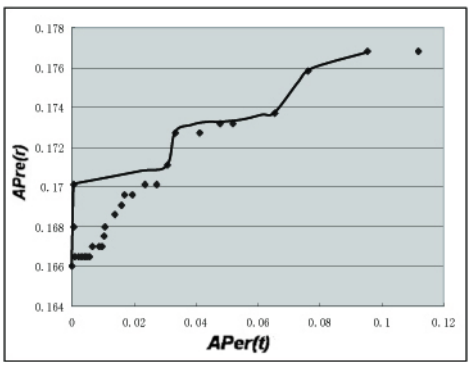
(a) top-5 ranked documents



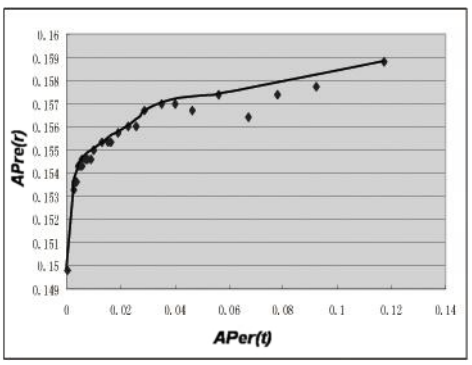
(b) top-10 ranked documents



(c) top-15 ranked documents



(d) top-20 ranked documents



(e) top-30 ranked documents

Figure 7.2: Trade-off surfaces of each of the top-k ranked documents for the TREC 451-550 short queries in the 1000 peer-sized network

decisions. This sorts the different near optimal solutions by the ratios of changing percentages between $APre_r$ and $APer_t$. This is because the property of the near optimal points in the trade-off surface is giving up $APre_r$ or $APer_t$, which allows the other objective to be improved. However, the changing percentages of both objectives are not same. Therefore, how much the document ranking algorithm (i.e., Equation 7.1) gives up the quality of one objective in order to improve the quality of another objective with different weighting can be measured. In the proposed document ranking approach, all of the near optimal points are compared with a base point to calculate the ratio of the changing percentage for $APre_r$ and $APer_t$. The ratio of changing percentages is defined as follows:

Definition 3 *Considering two near optimal points w^{base} and w^i with the corresponding values $APre_r$ and $APer_t$ on the trade-off surface, the ratio of changing percentage between $APre_r$ and $APer_t$ is denoted by $Rcp(w^{base}, w^i)$, where*

$$Rcp(w^{base}, w^i) = \frac{|APre_r(w^{base}) - APre_r(w^i)|}{APre_r(max) - APre_r(min)} \Big/ \frac{|APer_t(w^{base}) - APer_t(w^i)|}{APer_t(max) - APer_t(min)} \quad (7.2)$$

where w^i is the relative weight of a near optimal point and w^{base} is the relative weight of the base near optimal point on the trade-off surface, the $APre_r(max)$, $APre_r(min)$, $APer_t(max)$ and $APer_t(min)$ are the maximum trustworthiness value, minimum trustworthiness value, maximum relevance value and minimum relevance value in the trade-off table for each of the top-k ranked documents.

Basically, any point on the trade-off surface can be selected as the base point. In this chapter, the points with the minimum trustworthiness values are chosen. For example, (0, 0.2742) with w=0.0028 in top-5 ranked documents, and (0, 0.2165) with w=0.0029 in top-10 ranked documents. The $Rcp(w^{base}, w^i)$ is the ratio of improvement of the retrieval accuracy and the deterioration of the effectiveness of trust. The objective of $APer_t(max) - APer_t(min)$ and $APre_r(max) - APre_r(min)$ is to normalise the trustworthiness and relevance values in the objective space for each of the top-k ranked documents. The higher $Rcp(w^{base}, w^i)$ score means that the near optimal point w^i can obtain more improvement on $APre_r$ and less sacrifice on $APer_t$.

Table 7.2: Ranking table of the near optimal points for the study of the optimal weight w between relevance and trustworthiness in the proposed document ranking algorithm

	Rank	1	2	3	4	5
Top-5	base solution	(0, 0.2742), 0.0028				
	near optimal solution	(0.0062, 0.2763)	(0.0124, 0.2784)	(0.0309, 0.2804)	(0.0907, 0.2845)	(0.0062, 0.2763)
	$Rcp(w^{base}, w^i))$	0.1385	0.1385	0.0821	0.0465	0.1385
	w	0.9978	0.9982	0.9992,0.9993	0.9999	0.9979
Top-10	base solution	(0, 0.2165), 0.0029				
	near optimal solution	(0.0021,0.2186)	(0.0062, 0.2196)	(0.0155, 0.2227)	(0.0186, 0.2237)	(0.0124, 0.2206)
	$Rcp(w^{base}, w^i)$	0.5958	0.2979	0.2383	0.2306	0.1969
	w	0.9971,0.9972	0.9978	0.9987	0.9988	0.9986
Top-15	base solution	(0,0.1863), 0.003				
	near optimal solution	(0.0192, 0.1918)	(0.0172, 0.1911)	(0.0158, 0.1904)	(0.011, 0.1883)	(0.0481, 0.1938)
	$Rcp(w^{base}, w^i))$	0.1811	0.1764	0.1640	0.1149	0.0985
	w	0.9988	0.9987	0.9986	0.9984	0.9995
Top-20	base solution	(0,0.166), 0.0024				
	near optimal solution	(0.0005,0.1701)	(0.0335,0.1727)	(0.0309,0.1711)	(0.0763,0.1758)	(0.0954,0.1768)
	$Rcp(w^{base}, w^i)$	5.6675	0.1382	0.1140	0.0887	0.07824
	w	0.003	0.9993	0.9992	0.9998	0.9999
Top-30	base solution	(0.0003, 0.1498), 0.003				
	near optimal solution	(0.0027,0.1533)	(0.0031,0.1536)	(0.0048, 0.1543)	(0,0058, 0.1546)	(0.01, 0.155)
	$Rcp(w^{base}, w^i)$	1.1806	1.0986	0.8095	0.7065	0.4339
	w	0.9971	0.9972	0.9976	0.9978	0.9984

The ranking results are shown in Table 7.2. Only the best five near optimal points are presented in the table. It can be observed from Table 7.2 that the top-ranked near optimal points are not same for each of the top-k ranked documents, which means that no unique point can prefer the proposed document ranking algorithm in all cases. For example, if the decision-maker is only interested in the top-5 ranked documents, then the preferred w is 0.9978 or 0.9982. If the decision-maker does not specify any preference, a heuristic-based weighting approach is proposed, which is to rank all of the near optimal points in Table 7.2 by calculating their occurrences with the corresponding weights. For example, if rank 1 is set to 0.5 point, rank 2 to 0.4 point, rank 3 to 0.3 point, rank 4 to 0.2 point and rank 5 to 0.1 point, the best near optimal solution is 0.9978.

7.6 Limitations

The proposed heuristic-based search process to find the near optimal points of relative weights suffer from two drawbacks, namely, time consuming and computation expensive. The reasons for this are as follows:

- To find a set of near optimal points, users need to specify the weight manually, and weights can be infinite.
- For each individual weight, the proposed heuristic-based approach needs to compute $APre_r$ and $APer_t$, which involves retrieving documents from the network, ranking them, and comparing the ranked results with evaluation files.

It is difficult to employ the proposed heuristic-based search process to find the near optimal points of relative weights in practice for the above two reasons.

7.7 Summary

In this chapter, the relative weights between relevance and trustworthiness in the proposed document ranking and peer selection algorithms are studied. A heuristic-based search process is proposed to find the near optimal points and the trade-off surface for analysis. The limitation of this approach is that the precision of the near optimal points is heavily dependent on the precision of the relative

weights and step sizes. The heuristic-based search process is computationally expensive and time consuming. It should be noted that, if $APre_r$ and $APer_t$ with respect to the relative weight w can be represented by mathematical functions, then a number of mathematical approaches can be employed to find the exact Pareto-optimal points. Moreover, a ranking approach is proposed to sort the near optimal solutions by the ratios of changing percentages between $APre_r$ and $APer_t$. This may help users to select a preferred solution as the final answer.

Chapter 8

Conclusions and Future Work

Section 8.1 of this chapter summarises the research problem of this dissertation, and outlines the proposed solutions. The major contributions of the thesis and their impact are discussed in Section 8.2, while Section 8.3 represents a critique of the thesis. Section 8.4 discusses the direction for future work.

8.1 Problem and Summary

Information Retrieval in P2P networks (P2PIR) has become an active field of research in the last decade. P2P networks rely on voluntary peers to exchange information and accomplish tasks. Without prior experience of the network, peers run the risk of reviewing and downloading untrustworthy documents, even if these documents are relevant. Most of the existing P2PIR systems only focus on finding relevant documents for given queries, but ignore the trustworthiness of documents and document providers. Thus, it is necessary to integrate the feature of trustworthiness into P2PIR systems. However, current trust management systems in P2P networks focus on entity trust, which is not sufficient for P2PIR (as discussed in Sections 1.2.2 and 2.4). In this context, the work presented in this dissertation provides the first integrated framework for retrieving not only *relevant* but also *trustworthy* documents upon request in cooperative and uncooperative P2PIR environments.

In Chapter 4, a trust-aware P2PIR system is proposed in cooperative P2PIR environments. A method is designed to estimate global term statistics, and the proposed statistic is integrated with the K-L retrieval algorithm to compute relevance-base document scores for the proposed trust-aware P2PIR system.

This approach can facilitate effective and practical Information Retrieval in real P2P environments which are highly dynamic and distributed. Moreover, a set of content trust factors is identified in the context of P2PIR, and the content trust models are proposed to calculate the trust values of a document or document provider for a given query. A system architecture and data management protocols are designed to implement the proposed trust-aware P2PIR system in structured P2P networks. The proposed system architecture is an extension of the PeerTrust architecture in the context of trust-aware P2PIR. The proposed data management protocols are adaptations of the data management protocols of structured P2P networks in the context of statistical information and reputation data routing. A set of testbeds is developed to evaluate the performance of the proposed trust-aware P2PIR system in terms of retrieval accuracy, effectiveness of trust, and scalability of network size.

Chapter 5 addressed the problem of trust-aware P2PIR in uncooperative P2PIR environments, including trust-based peer description, peer selection, result merging and implementation. The proposed estimated global term statistics in Chapter 4 are integrated with a traditional resource selection algorithm CORI to compute relevance-based peer scores in the proposed trust-aware P2PIR system. In order to merge the retrieved results, a heuristic-based estimation function is proposed, which calculates the merged document scores by combining the document scores provided by peers and the peers' scores. To implement the proposed trust-aware P2PIR system in uncooperative P2P environments and structured P2P networks, a system architecture and data management protocols are developed. Moreover, a set of testbeds is developed to evaluate the performance of the proposed trust-aware P2PIR system in uncooperative environments. Preliminary experimental results show the retrieval accuracy, effectiveness of trust and scalability of network size.

Peer selection is an important problem for P2PIR in uncooperative environments. In Chapter 5, a heuristic-based peer selection approach is proposed to use a fixed and pre-determined threshold to decide the number of top-ranked peers to be selected, and the equal number of documents retrieved from each selected peer for any queries. On the contrary, a theory-based $PrRi$ peer selection model is proposed in Chapter 6. The proposed $PrRi$ precision-risk model can compute a clear cut-off for which peers to select, and the number of documents needed to be retrieved from each of the selected peers for a given query. These values

are computed by maximising the number of relevant documents in the results set, and minimising the risk of reviewing untrustworthy documents. Moreover, a system architecture and data management protocols are proposed to implement the $PrRi$ peer selection model in uncooperative P2PIR environments, as well as structured P2P networks. The experimental results show that the $PrRi$ peer selection model can achieve better retrieval accuracy in terms of both user satisfaction and the overall retrieval performance, compared to the heuristic-based peer selection approach.

In the previous chapters, the document ranking algorithm and peer selection algorithms were proposed for trust-aware P2PIR in cooperative and uncooperative environments. The objectives of retrieval accuracy and effectiveness of trust in one algorithm are often in conflict with each other. Therefore, the document ranking and peer selection algorithms may lose one objective, such as retrieval accuracy or effectiveness of trust, in return for meeting the other objective. In Chapter 7, the relative weights between relevance and trustworthiness in the proposed document ranking and peer selection algorithms are studied. A heuristic-based search process is proposed to find the near optimal points and the trade-off surface for analysis. Moreover, a ranking approach is proposed to sort the near optimal solutions by the ratios of changing percentages between $APre_r$ and $APer_t$. This may help users to select a preferred solution as the final answer.

8.2 Contributions and Impact

A large amount of text-based information is shared in distributed environments (e.g., Internet, P2P, Grid). Effective and practical technologies are required to retrieve information from distributed information providers to satisfy users' requirements (e.g., relevance). The reason for employing P2P architectures as the basic infrastructure for IR is that P2P networks may be an attractive alternative to current centralised search engines for both technical and economic reasons (as discussed in Sections 1.1 and 2.2.1). The objective of this dissertation has been to study of trust-aware P2PIR from both the P2PIR perspective and the security viewpoint, in order to propose new techniques to complement existing approaches in P2P networks. To be specific, it aims to develop a trust-aware P2PIR system which can retrieve not only *relevant* but *trustworthy* documents for queries in both cooperative and uncooperative P2PIR environments. This section presents

the contributions and the impact of the proposed trust-aware P2PIR system in various scenarios.

From the Information Retrieval perspective, the proposed trust-aware P2PIR system can be applied in cooperative and uncooperative P2PIR environments. Application areas of the proposed trust-aware P2PIR system in cooperative P2PIR environments (as described in Chapter 3) include two ways: *(i)* P2P Web search and *(ii)* P2P file sharing systems.

- In response to the issues of bottleneck, a single point of failure, scalability and high maintenance costs in traditional centralised Web search engines, P2P networks (especially, structured P2P networks) can offer the attractive solution of selecting a number of stable, powerful and good network connectivity peers as server peers to replace the centralised server in the Web search engine. This is referred to as *P2P Web search*. The proposed trust-aware P2PIR system in cooperative P2PIR environments can provide more effective, practical and security solutions to P2P Web search by retrieving and ranking not only relevant but also trustworthy documents in the top-ranked results list of Web documents.

- Vast numbers of P2P networks exist today for file sharing music, videos, software and text documents, such as Mule [eMu], aMule [aMu], MLDonkey [mld] and etc. For text document sharing, most of the existing file sharing systems only retrieve the document name or the meta-data which matches the query terms. The contents and trustworthiness of documents are not taken into account. Basically, a full-text based search has been common practice for Information Retrieval in a number of application areas. The proposed trust-aware P2PIR system in cooperative P2PIR environments can benefit the existing P2P file sharing applications, particularly for text document sharing, in an effective, practical and secure way. With this effective and practical solution to estimate global term statistics, the full-text based document retrieval algorithms can be easily employed in structured P2P networks (e.g., aMule, MLDonkey). Furthermore, the content trust model can be integrated into structured P2P networks without changing the network structures and routing protocols. The proposed trust-aware P2PIR system in cooperative P2PIR environments can be employed by the existing P2P file sharing networks for retrieving relevant and trustworthy documents upon request.

The proposed trust-aware P2PIR system in uncooperative P2PIR environments (as described in Chapters 5 and 6) can find its uses in applications of digital libraries search. To date, a large number of text digital libraries are available on the Internet, and they only permit documents to be retrieved through their own individual search engines, which cannot be crawled by the public Web search engines. Most of the existing uncooperative P2PIR systems use a P2P network to organise these text digital libraries into an overlay with P2P network protocols such as Chord. To conduct a search, a query is forwarded to a number of text digital libraries relevant to the given query with a single interface, and then a list of results is retrieved from each of the selected text digital libraries. These are merged together and presented to users. However, the existing P2PIR systems in uncooperative environments lack mechanisms to prevent the entry of malicious peers and untrustworthy documents. The proposed trust-aware P2PIR system in uncooperative P2PIR environments can provide effective, practical and secure solutions for text digital libraries' search by selecting a set of relevant and trustworthy text digital libraries to search and merging the results returned from them.

Not only can P2PIR benefit from the proposed trust-aware P2PIR system, but also some applications of trust management systems in P2P networks. Security issues are recognised as being a critical problem for information sharing in P2P networks. More and more trust management systems have been developed so far to protect users from being attacked by malicious peers. However, the existing trust management systems only focus on entity trust, which is insufficient for P2PIR. Many relevant and irrelevant documents may be assigned to the same trust value, which makes it hard to select one as the final answer. The proposed trust-aware P2PIR system builds upon both entity trust and relevance to complement existing trust management systems which are limited in text-based trust applications.

In addition to the proposed trust-aware P2PIR systems, this dissertation also proposes a set of testbeds and methodologies for the evaluation of existing P2PIR systems with security issues, or existing trust management systems with IR issues. Since the experimental results have demonstrated the effectiveness of the testbeds and evaluation methodologies in both cooperative and uncooperative environments, they can also be used by other research applications.

8.3 Critique of the Thesis

A critique of the thesis is discussed as follows:

- This thesis identifies a crucial problem which has never been researched before, which is trust-aware P2PIR. In fact, work relating to trust-aware P2PIR is extremely limited. In this thesis, the literature review mainly focuses on P2PIR and trust in P2P. Currently, trust in the Semantic Web has been an active research field. It is considered that it would be desirable to pay some attention to the trust models in the Semantic Web, so a better content trust model may be proposed for P2PIR.

- The experimental settings and results may not be sufficient because different situations are not taken into account. For example, a high percentage of the malicious peers in the network are out of research. It would be interesting to obtain an in-depth experimental evaluation to study the performance of the proposed trust-aware P2PIR system in different situations.

- Since P2P networks provide more document selection criteria than traditional IR environments, multi-objective optimisation would be interesting. For example, when retrieving a set of documents, a user may want to maximise retrieval accuracy, minimum time and money cost, and maximise the effectiveness of trust.

- In Chapter 7, a heuristic-based search process of near optimal solutions is proposed because relevance and trustworthiness cannot be presented in mathematical functions. Alternatively, the estimated relevance and estimated trustworthiness may be used to determine the optimal solutions by modifying the functions in Chapter 6.

8.4 Future Work

This section discusses some open questions. The proposed work in this dissertation can be extended and improved in the following ways:

- Unlike most P2PIR systems which rank documents and peers based on relevance, the proposed trust-aware P2PIR system uses both relevance and trustworthiness as selection criteria to retrieve documents. Basically, P2P

networks offer more selection criteria, such as messages routing costs, which are normally assumed to be perfect in traditional IR environments, but are important factors in P2P networks [LZL06]. These essential factors should be taken into account when retrieving relevant documents in P2P networks. In future work, a set of new selection criteria should be chosen based on the unique characteristics of P2P networks as distinguished from traditional IR environments.

- Since structured P2P networks are regarded as being a significant improvement on unstructured P2P networks for scalability, efficiency and reliability, the proposed trust-aware P2PIR systems are designed based on the characteristics of structured P2P networks (e.g., routing table size). However, a number of unstructured P2P networks still exist, such as Gnutella. Although the proposed estimating global statistics cannot be applied to unstructured P2P networks, the content trust models in the proposed trust-aware P2PIR system offer sufficient flexibility to be integrated in unstructured P2P networks. However, new implementation strategies are needed. It would be interesting to integrate the proposed content trust models to P2PIR applications in unstructured P2P networks.

- It is difficult to do research in academic environments without thousands of computers and real users, and therefore, the simplification of the evaluation and network settings may ignore some real-world problems. For example, one of the important factors of employing applications in real P2P networks should be the cost of message routing. The statistical information in the proposed trust-aware P2PIR system should be large in size, which may produce a significant amount of network traffic and search latency overheads. Currently, researchers have developed a number of methods to reduce the information size of full-text based descriptions in P2PIR, which could be employed for studying the retrieval performance and effectiveness of trust in the proposed trust-aware P2PIR system.

- Chapter 6 demonstrates four kinds of parameters in the DTF-rp approach, linear relevant mapping functions + linear recall-precision curves, linear relevant mapping functions + quadratic recall-precision curves, logistic relevant mapping functions + linear recall-precision curves, and logistic relevant mapping functions + quadratic recall-precision curves. For simplicity,

only the parameters for linear relevant mapping functions + linear recall-precision curves have been studied. The remaining parameters can be generated in future work.

- The metric for the evaluation of effectiveness of trust used so far is the average percentage of untrustworthy documents at given document cut-off values, which only focuses on the top-ranked results list. More evaluation metrics should be developed, such as metrics to evaluate the effectiveness of trust for the overall results list. Moreover, in the experiments, the percentage of malicious peers is initially set to 20% and the credibility of good peers and malicious peers is set to 90% and 20%, respectively. A number of experimental settings could be changed for evaluation in further studies, for example, the percentage of malicious peers in the network, the credibility of good peers and malicious peers, the percentage of untrustworthy documents in a good peer, and the percentage of trustworthy documents in a malicious peer.

- Almost all of the approaches in P2PIR only consider the individual scenarios of either cooperative or uncooperative Information Retrieval. There is no system capable of addressing the problem of Information Retrieval in a combined scenario. For example, a P2P network may mix with text digital libraries (with access limitations) and personal users (without access limitations), and it would be challenging to design and develop a trust-aware P2PIR system in such an environment.

Bibliography

[ACM] ACM Digital Library. `http://portal.acm.org/dl.cfm`. Last accessed: May 2010.

[ACMD+03] Karl Aberer, Philippe Cudré-Mauroux, Anwitaman Datta, Zoran Despotovic, Manfred Hauswirth, Magdalena Punceva, and Roman Schmidt. P-Grid: A Self-organising Structured P2P System. *SIGMOD Record*, 32(3):29–33, 2003.

[ADDV+06] Roberto Aringhieri, Ernesto Damiani, Sabine De Capitani Di Vimercati, Stefano Paraboschi, and Pierangelo Samarati. Fuzzy Techniques for Trust and Reputation Management in Anonymous Peer-to-Peer Systems: Special Topic Section on Soft Approaches to Information Retrieval and Information Access on the Web. *The American Society for Information Science & Technology*, 57(4):528–537, 2006.

[AFS89] Serge Abiteboul, Patrick Fischer, and Hans-Jorg Schek, editors. *Nested Relations and Complex Objects in Databases*. Springer-Verlag, New York, USA, 1989.

[AG07] Donovan Artz and Yolanda Gil. A Survey of Trust in Computer Science and the Semantic Web. *Web Semantics*, 5(2):58–71, 2007.

[ALPH01] Lada A. Adamic, Rajan M. Lukose, Amit R. Puniyani, and Bernardo A. Huberman. Search in Power-Law Networks. *Physical Review E*, 64(4):046135, Sep 2001.

[AMCB04] Nazareno Andrade, Miranda Mowbray, Walfredo Cirne, and Francisco Brasileiro. When Can an Autonomous Reputation Scheme Discourage Free-riding in a Peer-to-Peer System? In *Proceedings of*

the IEEE International Symposium on Cluster Computing and the Grid, pages 440–448, 2004.

[aMu] aMule. `www.amule.org/`. Last accessed: May 2010.

[ARH99] Alfarez Abdul-Rahman and Stephen Hailes. Relying on Trust to Find Reliable Information. In *Proceedings of the 1999 International Symposium on Database, Web and Cooperative Systems*, 1999.

[Ati02] Yacine Atif. Building Trust in E-Commerce. *IEEE Internet Computing*, 6(1):18–24, 2002.

[ATS04] Stephanos Androutsellis-Theotokis and Diomidis Spinellis. A Survey of Peer-to-Peer Content Distribution Technologies. *ACM Computing Surveys*, 36(4):335–371, 2004.

[BCM+03] Franz Baader, Diego Calvanese, Deborah L. McGuinness, Daniele Nardi, and Peter F. Patel-Schneider. The Description Logic Handbook: Theory, Implementation, and Applications. *Description Logic Handbook*, 2003.

[BDMS08] Jurgen Branke, Kalyanmoy Deb, Kaisa Miettinen, and Roman Slowinski. *Multiobjective Optimization: Interactive and Evolutionary Approaches*. Springer-Verlag, Berlin, Heidelberg, 2008.

[BDOS05] Piero Bonatti, Claudiu Duma, Daniel Olmedilla, and Nahid Shahmehri. An Integration of Reputation-Based and Policy-Based Trust Management. In *Proceedings of the Semantic Web Policy Workshop*, 2005.

[BK00] K. Suzanne Barber and Joonoo Kim. Belief Revision Process Based on Trust: Agents Evaluating Reputation of Information Sources. In *Trust in Cyber-societies*, pages 73–82, 2000.

[Blo70] Burton H. Bloom. Space/time Trade-offs in Hash Coding with Allowable Errors. *Communications of the ACM*, 13(7):422–426, 1970.

[BO05] Piero Bonatti and Daniel Olmedilla. Driving and Monitoring Provisional Trust Negotiation with Metapolicies. In *Proceedings of the 6th IEEE International Workshop on Policies for Distributed Systems*

and Networks, pages 14–23, Washington, DC, USA, 2005. IEEE Computer Society.

[BP98] Sergey Brin and Lawrence Page. The Anatomy of a Large-scale Hypertextual Web Search Engine. *Computer Networks and ISDN Systems*, 30(1-7):107–117, 1998.

[Bun97] Peter Buneman. Semistructured Data. In *Proceedings of the sixteenth ACM SIGACT-SIGMOD-SIGART symposium on Principles of Database Systems*, pages 117–121, New York, USA, 1997. ACM.

[BV04] Chris Buckley and Ellen M. Voorhees. Retrieval Evaluation with Incomplete Information. In *Proceedings of the 27th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 25–32, New York, USA, 2004. ACM.

[BYRN99] Ricardo Baeza-Yates and Berthier Ribeiro-Neto. *Modern Information Retrieval*. Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA, 1999.

[Cal00] Jamie Callan. Distributed Information Retrieval. *Advances in Information Retrieval*, 5:127–150, 2000.

[CAN02] Francisco Matias Cuenca-Acuna and Thu D. Nguyen. Text-Based Content Search and Retrieval in Ad-hoc P2P Communities. In *Proceedings of the Networking Workshop*, pages 220–234, 2002.

[CBH00] Nick Craswell, Peter Bailey, and David Hawking. Server Selection on the World Wide Web. In *Proceedings of the Fifth ACM Conference on Digital Libraries*, pages 37–46, 2000.

[CC01] James P. Callan and Margaret E. Connell. Query-Based Sampling of Text Databases. *ACM Transactions on Information Systems*, 19(2):97–130, 2001.

[CC05] Tom Chothia and Konstantinos Chatzikokolakis. A Survey of Anonymous Peer-to-Peer File-Sharing. In *Proceedings of the International Conference on Embedded and Ubiquitous Computing*, pages 744–755, 2005.

[CCB95] James P. Callan, W. Bruce Croft, and John Broglio. TREC and TIPSTER Experiments with INQUERY. *Information Processing Management*, 31(3):327–343, 1995.

[CCL01] Charles L. A. Clarke, Gordon V. Cormack, and Thomas R. Lynam. Exploiting Redundancy in Question Answering. In *Proceedings of the 24th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 358–365, New York, USA, 2001. ACM.

[CF04] Jamie Callan and Norbert Fuhr. SIGIR Peer-to-Peer Information Retrieval Workshop. *ACM SIGIR Forum*, 38(2):37–40, 2004.

[Cha81] David L. Chaum. Untraceable Electronic Mail, Return Addresses, and Digital Pseudonyms. *Communications of the ACM*, 24(2):84–90, 1981.

[Che05] Yan Chen. A Scalable Semantic Indexing Framework for Peer-to-Peer Information Retrieval. In *Proceedings of the SIGIR 2005 Workshop: Heterogeneous and Distributed Information Retrieval*, 2005.

[CJL+09] Hanhua Chen, Hai Jin, Xucheng Luo, Yunhao Liu, and Lionel M. Ni. BloomCast: Efficient Full-Text Retrieval over Unstructured P2Ps with Guaranteed Recall. In *Proceedings of the IEEE International Symposium on Cluster Computing and the Grid*, pages 52–59, Los Alamitos, CA, USA, 2009. IEEE Computer Society.

[CR01] Chowdhury and Abdur Rashid. *On the Design of Reliable Efficient Information Systems*. PhD thesis, Illinois Institute of Technology, Chicago, IL, USA, 2001. Adviser-Frieder, Ophir.

[CS03] Yann Collette and Patrick Siarry. *Multiobjective Optimization: Principles and Case Studies*. Springer, 2003.

[CSB+05] Sergey Chernov, Pavel Serdyukov, Matthias Bender, Sebastian Michel, Gerhard Weikum, and Christian Zimmer. Database Selection and Result Merging in P2P Web Search. In *Proceedings of the Databases, Information Systems, and Peer-to-Peer Computing Workshop*, pages 26–37, 2005.

[DA06] Zoran Despotovic and Karl Aberer. P2P Reputation Management: Probabilistic Estimation vs. Social Networks. *Computer Networks*, 50(4):485–500, 2006.

[DdVP+02] Ernesto Damiani, Sabrina De Capitani di Vimercati, Stefano Paraboschi, Pierangela Samarati, and Fabio Violante. A Reputation-Based Approach for Choosing Reliable Resources in Peer-to-Peer Networks. In *Proceedings of the 9th ACM conference on Computer and Communications Security*, pages 207–216, New York, USA, 2002. ACM.

[DdVPS03] Ernesto Damiani, Sabrina De Capitani di Vimercati, Stefano Paraboschi, and Pierangela Samarati. Managing and Sharing Servents' Reputations in P2P Systems. *IEEE Transactions on Knowledge and Data Engineering*, 15(4):840–854, 2003.

[DGM+03] Neil Daswani, Philippe Golle, Sergio Marti, Hector Garcia-Molina, and Dan Boneh. Evaluating Reputation Systems for Document Authenticity. *Technical report, Computer Science Department, Stanford University*, 2003.

[DM01] Inderjit S. Dhillon and Dharmendra S. Modha. Concept Decompositions for Large Sparse Text Data Using Clustering. *Machine Learning*, 42(1/2):143–175, 2001.

[DMP+09] Patrizio Dazzi, Matteo Mordacchini, Raffaele Perego, Pascal Felber, Lorenzo Leonini, Martin Rajman, and Etienne Riviere. Peer-to-Peer Clustering of Web-browsing Users. In *Workshop on Large-Scale Distributed Systems for Information Retrieval*, 2009.

[DSM03] Rajan Lukose Kiran Nagaraja Jim Pruyne Bruno Richard Sami Rollins Zhichen Xu Dejan S. Milojicic, Vana Kalogeraki. Peer-to-Peer Computing. *Technical Report, HPL-2002-57*, 2003.

[eBa] eBay. `www.ebay.com/`. Last accessed: May 2010.

[EJ] Donald E. Eastlake and Paul E. Jones. US Secure Hash Algorithm 1 (SHA1). `http://www.ietf.org/rfc/rfc3174.txt?number=3174`.

[eMu] eMule. `www.emule.com/`. Last accessed: May 2010.

[FC00] Rino Falcone and Cristiano Castelfranchi. The Socio-cognitive Dynamics of Trust: Does Trust Create Trust? In *Trust in Cybersocieties*, pages 55–72, 2000.

[FC04] Rino Falcone and Cristiano Castelfranchi. Trust Dynamics: How Trust Is Influenced by Direct Experiences and by Trust Itself. In *Proceedings of the Third International Joint Conference on Autonomous Agents and Multiagent Systems*, pages 740–747, 2004.

[FFF99] Michalis Faloutsos, Petros Faloutsos, and Christos Faloutsos. On Power-law Relationships of the Internet Topology. In *Proceedings of the Conference on Applications, Technologies, Architectures, and Protocols for Computer Communication*, pages 251–262, New York, USA, 1999. ACM.

[FM87] Peter Frankl and Hiroshi Maehara. The Johnson-Lindenstrauss Lemma and the Sphericity of Some Graphs. *Journal of Combinatorial Theory, Series A*, 44(3):355–362, 1987.

[FPC+99] James C. French, Allison L. Powell, James P. Callan, Charles L. Viles, Travis Emmitt, Kevin J. Prey, and Yun Mou. Comparing the Performance of Database Selection Algorithms. In *Proceedings of the Research and Development in Information Retrieval Conference*, pages 238–245, 1999.

[Fuh99] Norbert Fuhr. A Decision-theoretic Approach to Database Selection in Networked IR. *ACM Transactions on Information Systems*, 17(3):229–249, 1999.

[GA07] Yolanda Gil and Donovan Artz. Towards Content Trust of Web Resources. *Web Semantics: Science, Services and Agents on the World Wide Web*, 5(4):227 – 239, 2007.

[GEBR+03] Luis Garcés-Erice, Ernst W. Biersack, Keith W. Ross, Pascal Felber, and Guillaume Urvoy-Keller. Hierarchical Peer-To-Peer Systems. *Parallel Processing Letters*, 13(4):643–657, 2003.

[GF07] Dion Goh and Schubert Foo. *Social Information Retrieval Systems:*

Emerging Technologies and Applications for Searching the Web Effectively. Information Science Reference - Imprint of: IGI Publishing, Hershey, PA, 2007.

[GGMP04] Zoltán Gyöngyi, Hector Garcia-Molina, and Jan Pedersen. Combating Web Spam with Trustrank. In *Proceedings of the Thirtieth International Conference on Very Large Databases*, pages 576–587. VLDB Endowment, 2004.

[GKN09] Fausto Giunchiglia, Uladzimir Kharkevich, and Sheak Noori. P2P Concept Search: Some Preliminary Results. In *Proceedings of the Semantic Search Workshop in WWW*, 2009.

[GMH04] Sakaryan German, Wulff Markus, and Unger Herwig. Search Methods in P2P Networks: A Survey. In *Proceedings of the 4th International Workshop on Innovative Internet Community Systems*, pages 59–68, 2004.

[GMS06] Christos Gkantsidis, Milena Mihail, and Amin Saberi. Random Walks in Peer-to-Peer Networks: Algorithms and Evaluation. *Performance Evaluation*, 63(3):241–263, 2006.

[gnu] Gnutella. `www.gnutella.com`. Last accessed: May 2010.

[goo] Google. `www.google.com/`. Last accessed: May 2010.

[Hir05] J. E. Hirsch. An Index to Quantify an Individual's Scientific Research Output. *National Academy of Sciences of the United States of America*, 102(46):16569–16572, November 2005.

[Hof99] Thomas Hofmann. Probabilistic Latent Semantic Analysis. In *Proceedings of Conference on Uncertainty in Artificial Intelligence*, pages 289–296, 1999.

[Hor05] Ian Horrocks. OWL: A Description Logic Based Ontology Language. In *Proceedings of the International Logic Programming Conference*, pages 1–4, 2005.

[HZM+08] Quirin Hofstätter, Stefan Zöls, Maximilian Michel, Zoran Despotovic, and Wolfgang Kellerer. Chordella - A Hierarchical Peer-to-Peer Overlay Implementation for Heterogeneous, Mobile Environments. *IEEE International Conference on Peer-to-Peer Computing*, pages 75–76, 2008.

[JDXB03] Xuxian Jiang, Yu Dong, Dongyan Xu, and Bharat Bhargava. Gnustream: A P2P Media Streaming System Prototype. In *Proceedings of the International Conference on Multimedia and Expo*, pages 325–328, 2003.

[JH97] Joemon M. Jose and David J. Harper. A Retrieval Mechanism for Semi-Structured Photographic Collections. In *Proceedings of DEXA 97*, pages 276–292. Springer, 1997.

[JNC06] Hai Jin, Xiaomin Ning, and Hanhua Chen. Efficient Search for Peer-to-Peer Information Retrieval Using Semantic Small World. In *Proceedings of the International World Wide Web Conference*, pages 1003–1004, 2006.

[JT99] Catholijn M. Jonker and Jan Treur. Formal Analysis of Models for the Dynamics of Trust Based on Experiences. In *Proceedings of the 9th European Workshop on Modelling Autonomous Agents in a Multi-Agent World*, pages 221–231, 1999.

[JYF07] Yuh-Jzer Joung, Li-Wei Yang, and Chien-Tse Fang. Keyword Search in DHT-Based Peer-to-Peer Networks. *IEEE Journal on Selected Areas in Communications*, 25(1):46–61, 2007.

[KA05] Fabius Klemm and Karl Aberer. Aggregation of a Term Vocabulary for Peer-to-Peer Information Retrieval: A DHT Stress Test. In *Proceedings of the Third International Workshop on Databases, Information Systems and Peer-to-Peer Computing*, 2005.

[KGM95] Steven P. Ketchpel and Hector Garcia-Molina. Making Trust Explicit in Distributed Commerce Transactions. In *Proceedings of the 16th International Conference on Distributed Computing Systems*, pages 694–701, Washington, DC, USA, 1995. IEEE Computer Society.

[Kir] S. Kirsch. Document Retrieval over Networks Wherein Ranking and Relevance Scores Are Computed at the Client for Multiple Database Documents. *U.S. Patent*, 5,659,732.

[KJ03] Iraklis A. Klampanos and Joemon M. Jose. An Architecture for Peer-to-Peer Information Retrieval. In *Proceedings of the 26th Annual International ACM SIGIR Conference on Research and Development in Informaion Retrieval*, pages 401–402, New York, USA, 2003. ACM.

[KJ04] Iraklis A. Klampanos and Joemon M. Jose. An Architecture for Information Retrieval over Semi-collaborating Peer-to-Peer Networks. In *Proceedings of the ACM Symposium on Applied Computing*, pages 1078–1083, New York, USA, 2004. ACM.

[KLK07] O-Hoon Kwon, So Young Lee, and Jong Kim. FileTrust: Reputation Management for Reliable Resource Sharing in Structured Peer-to-Peer Networks. *IEICE Transactions*, 90-B(4):826–835, 2007.

[KMF08] Gabriella Kazai and Natasa Milic-Frayling. Trust, Authority and Popularity in Social Information Retrieval. In *Proceeding of the 17th ACM conference on Information and Knowledge Management*, pages 1503–1504, New York, USA, 2008. ACM.

[KPJD05] Iraklis A. Klampanos, Victor Poznanski, Joemon M. Jose, and Peter Dickman. A Suite of Testbeds for the Realistic Evaluation of Peer-to-Peer Information Retrieval Systems. In *ECIR*, pages 38–51, 2005.

[KSGM03] Sepandar D. Kamvar, Mario T. Schlosser, and Hector Garcia-Molina. The EigenTrust Algorithm for Reputation Management in P2P Networks. In *Proceedings of the Twelfth International World Wide Web Conference*, pages 640–651. ACM Press, 2003.

[KT06] Eleni Koutrouli and Aphrodite Tsalgatidou. Reputation-Based Trust Systems for P2P Applications: Design Issues and Comparison Framework. *Trust and Privacy in Digital Business*, pages 152–161, 2006.

[KTR05] Michael Kinateder, Ralf Terdic, and Kurt Rothermel. Strong Pseudonymous Communication for Peer-to-Peer Reputation Systems. In *Proceedings of the 2005 ACM Symposium on Applied Computing*, pages 1570–1576, New York, USA, 2005. ACM.

[KWTA07] Hisashi Kurasawa, Hiromi Wakaki, Atsuhiro Takasu, and Jun Adachi. Data Allocation Scheme Based on Term Weight for P2P Information Retrieval. In *Proceedings of the Web Information and Data Management Workshop*, pages 33–40, 2007.

[LC03] Jie Lu and James P. Callan. Content-Based Retrieval in Hybrid Peer-to-Peer Networks. In *Proceedings of the ACM Conference on Information and Knowledge Management*, pages 199–206, 2003.

[LC04a] Jie Lu and Jamie Callan. Merging Retrieval Results in Hierarchical Peer-to-Peer Networks. In *Proceedings of the 27th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 472–473, New York, USA, 2004. ACM.

[LC04b] Jianming Lv and Xueqi Cheng. WonGoo: A Pure Peer-to-Peer Full Text Information Retrieval System Based On Semantic Overlay Networks. In *Proceedings of the IEEE International Symposium on Network Computing and Applications*, pages 47–54, 2004.

[LC05] Jie Lu and Jamie Callan. Federated Search of Text-Based Digital Libraries in Hierarchical Peer-to-Peer Networks. In *Proceedings of the Annual European Conference on Information Retrieval*, pages 52–66, 2005.

[LC06] Jie Lu and Jamie Callan. Full-text Federated Search of Text-Based Digital Libraries in Peer-to-Peer Networks. *Information Retrieval*, 9(4):477–498, 2006.

[LC07a] Jie Lu and Jamie Callan. Content-Based Peer-to-Peer Network Overlay for Full-Text Federated Search. In *Proceedings of the International Conference on Adaptivity, Personalisation and Fusion of Heterogeneous Information*, 2007.

[LC07b] Jie Lu and Jamie Callan. Full-text Federated Search of Text-Based Digital Libraries in Peer-to-Peer Networks. *PhD dissertation, CMU*, pages 477–498, 2007.

[LCC+02] Qin Lv, Pei Cao, Edith Cohen, Kai Li, and Scott Shenker. Search and Replication in Unstructured Peer-to-Peer Networks. In *Proceedings of the 16th International Conference on Supercomputing*, pages 84–95, New York, USA, 2002. ACM.

[LCP+05] Keong Lua, J. Crowcroft, M. Pias, R. Sharma, and S. Lim. A Survey and Comparison of Peer-to-Peer Overlay Network Schemes. *IEEE Communications Surveys & Tutorials*, pages 72–93, 2005.

[Lee97] Joon Ho Lee. Analyses of Multiple Evidence Combination. In *Proceedings of the 20th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 267–276, New York, NY, USA, 1997. ACM.

[lem] The Lemur Toolkit. `http://www.lemurproject.org/`. Last accessed: May 2010.

[LJT07] Yingguang Li, H. V. Jagadish, and Kian-Lee Tan. SPRITE: A Learning-Based Text Retrieval System in DHT Networks. In *Proceedings of the International Conference on Data Engineering*, pages 1106–1115, 2007.

[LKZ+06] Toan Luu, Fabius Klemm, Ivana Podnar Zarko, Martin Rajman, and Karl Aberer. ALVIS Peers: A Scalable Full-text Peer-to-Peer Retrieval Engine. In *Proceedings of the International Workshop on Information Retrieval in Peer-to-Peer Networks*, pages 41–48, New York, USA, 2006. ACM.

[LLH+03] Jinyang Li, Boon Thau Loo, Joseph M. Hellerstein, M. Frans Kaashoek, David R. Karger, and Robert Morris. On the Feasibility of Peer-to-Peer Web Indexing and Search. In *Proceedings of the International Workshop on Peer-To-Peer Systems*, pages 207–215, 2003.

[LLQ+04] Zhiguo Lu, Bo Ling, Weining Qian, Wee Siong Ng, and Aoying Zhou. A Distributed Ranking Strategy in Peer-to-Peer Based Information Retrieval Systems. In *Proceedings of the International Asia-Pacific Web Conference*, pages 279–284, 2004.

[Lua05] Eng Keong Lua. Hierarchical Peer-to-Peer Networks Using Lightweight SuperPeer Topologies. In *Proceedings of the 10th IEEE Symposium on Computers and Communications*, pages 143–148, Washington, DC, USA, 2005. IEEE Computer Society.

[Lyn01] Clifford A. Lynch. When Documents Deceive: Trust and Provenance as New Factors for Information Retrieval in a Tangled Web. *Journal of the American Society for Information Science and Technology*, 52(1):12–17, 2001.

[LZBT03] Chu Yee Liau, Xuan Zhou, Stéphane Bressan, and Kian-Lee Tan. Efficient Distributed Reputation Scheme for Peer-to-Peer Systems. In *Human.Society@Internet 2003*, pages 54–63, 2003.

[LZL06] Dik Lun Lee, Dyce Jing Zhao, and Qiong Luo. Information Retrieval in a Peer-to-Peer Environment. In *Proceedings of the 1st International Conference on Scalable Information Systems*, page 48, New York, USA, 2006. ACM.

[MA04] R.T. Marler and J.S. Arora. Survey of Multi-objective Optimization Methods for Engineering. *Structural and Multidisciplinary Optimization*, 262004(6):369–395, 2004.

[Mai] Siddharth Maini. A Survey Study on Reputation-Based Trust Management in P2P Networks. *Technical Report, Kent State University.*

[Mal03] Robin Jan Maly. Comparison of Centralised (Client-Server) and Decentralised (Peer-to-Peer) Networking, March 2003. Semester Thesis, ETH Zurich, Switzerland.

[Mau96] Ueli M. Maurer. Modelling a Public-Key Infrastructure. In *Proceedings of the European Symposium on Research in Computer Security*, pages 325–350, 1996.

[MBR03] Gurmeet Singh Manku, Mayank Bawa, and Prabhakar Raghavan. Symphony: Distributed Hashing in a Small World. In *Proceedings of the 4th USENIX Symposium on Internet Technologies and Systems*, pages 127–140, 2003.

[MGM06] Sergio Marti and Hector Garcia-Molina. Taxonomy of Trust: Categorising P2P Reputation Systems. *Computer Networks*, 50(4):472–484, 2006.

[MKA06] Tsunenori Mine, Akihiro Kogo, and Makoto Amamiya. Agent-community Based Peer-to-Peer Information Retrieval: An Evaluation. In *Proceedings of the Fifth International Joint Conference on Autonomous Agents and Multiagent Systems*, pages 1323–1325, New York, USA, 2006. ACM.

[mld] MLDonkey. `http://mldonkey.org/`. Last accessed: May 2010.

[MM02] P. Maymounkov and D. Mazieres. Kademlia: A Peer-to-Peer Information System Based on the XOR Metric. In *Proceedings of International Workshop on Peer-To-Peer Systems*, 2002.

[MM09] Hamid Mousavi and Ali Movaghar. Challenges in Using Peer-to-Peer Structures in Order to Design a Large-Scale Web Search Engine. *Advances in Computer Science and Engineering*, 6:461–468, 2009.

[MMKA04] Tsunenori Mine, Daisuke Matsuno, Akihiro Kogo, and Makoto Amamiya. Design and Implementation of Agent Community Based Peer-to-Peer Information Retrieval Method. In *Workshop CIA-2004 on Cooperative Information Agents*, pages 31–46, 2004.

[MR04] David S. H. Rosenthal TJ Giuli Petros Maniatis Jeff Mogul Mema Roussopoulos, Mary Baker. 2 P2P or not 2 P2P? In *Proceedings of the Third International Workshop on Peer-to-Peer Systems*, 2004.

[MS05] Kieran McDonald and Alan F. Smeaton. A Comparison of Score, Rank and Probability-Based Fusion Methods for Video Shot Retrieval. In *Proceedings of the 4th International ConferenceImage and Video Retrieval*, pages 61–70, 2005.

[myp] MyP2P: Living TV Programmes. `www.myp2p.eu/`. Last accessed: May 2010.

[nap] Napster. `www.Napster.com`. Last accessed: May 2010.

[NF03] Henrik Nottelmann and Norbert Fuhr. Evaluating Different Methods of Estimating Retrieval Quality for Resource Selection. In *Proceedings of the 26th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 290–297, New York, USA, 2003. ACM.

[NF04] Henrik Nottelmann and Norbert Fuhr. Combining CORI and the Decision-theoretic Approach for Advanced Resource Selection. In *Proceedings of the Annual European Conference on Information Retrieval*, pages 138–153. Springer, 2004.

[NF07a] Henrik Nottelmann and Gudrun Fischer. Search and Browse Services for Heterogeneous Collections with the Peer-to-Peer Network Pepper. *Information Processing and Management*, 43(3):624–642, 2007.

[NF07b] Henrik Nottelmann and Norbert Fuhr. A Decision-theoretic Model for Decentralised Query Routing in Hierarchical Peer-to-Peer Networks. In *Proceedings of the Annual European Conference on Information Retrieval*, pages 148–159, 2007.

[NFTN05] Henrik Nottelmann, Gudrun Fischer, Alexej Titarenko, and Andre Nurzenski. An Integrated Approach for Searching and Browsing in Heterogeneous Peer-to-Peer Networks. *SIGIR Workshop on Heterogeneous and Distributed Information Retrieval*, 2005.

[NJYF07] Linh Thai Nguyen, D. Jia, Wai Gen Yee, and Ophir Frieder. Analysis of Query Logs in Gnutella Peer-to-Peer Network. In *Proceedings of the ACM Thirtieth Conference on Research and Development in Information Retrieval*, 2007.

[NYF08a] Linh Thai Nguyen, Wai Gen Yee, and Ophir Frieder. Adaptive Distributed Indexing for Structured Peer-to-Peer Networks. In *Proceeding of the 17th ACM Conference on Information and Knowledge Management*, pages 1241–1250, New York, USA, 2008. ACM.

[NYF08b] Linh Thai Nguyen, Wai Gen Yee, and Ophir Frieder. Query Workload Driven Summarization for P2P Query Routing. In *Proceedings of the 2008 Eighth International Conference on Peer-to-Peer Computing*, pages 63–72, Washington, DC, USA, 2008. IEEE Computer Society.

[OLT03] Beng Chin Ooi, Chu Yee Liau, and Kian-Lee Tau. Managing Trust in Peer-to-Peer Systems Using Reputation-Based Techniques. In *Proceedings of the International Conference on Web-Age Information Management*, pages 2–12, 2003.

[ORMN05] Daniel Olmedilla, Omer F. Rana, Brian Matthews, and Wolfgang Nejdl. Security and Trust Issues in Semantic Grids. In *Semantic Grid*, 2005.

[Pap08] Odysseas Papapetrou. Full-text Indexing and Information Retrieval in P2P Systems. In *Ph.D. '08: Proceedings of the 2008 EDBT Ph.D. workshop*, pages 49–57, New York, USA, 2008. ACM.

[PFC+00] Allison L. Powell, James C. French, James P. Callan, Margaret E. Connell, and Charles L. Viles. The Impact of Database Selection on Distributed Searching. In *Proceedings of the Research and Development in Information Retrieval Conference*, pages 232–239, 2000.

[PJS+98] Chengxiang Zhai Peter, Peter Jansen, Emilia Stoica, Norbert Grot, and David A. Evans. Threshold Calibration in CLARIT Adaptive Filtering. In *Proceedings of the Seventh Text Retrieval Conference*, pages 149–156, 1998.

[PLPZR08] Maroje Puh, Toan Luu, Ivana Podnar Zarko, and Martin Rajman. Scalable Content-Based Ranking in P2P Information Retrieval. In *Proceedings of the 12th International Conference on Knowledge-Based Intelligent Information and Engineering Systems, Part I*, pages 633–640, Berlin, Heidelberg, 2008. Springer-Verlag.

[PMB06] Josiane Xavier Parreira, Sebastian Michel, and Matthias Bender. Size Doesn't Always Matter: Exploiting PageRank for Query Routing in Distributed IR. In *Proceedings of the International Workshop*

on Information Retrieval in Peer-to-Peer Networks, pages 25–32, New York, USA, 2006. ACM.

[Por97] M. F. Porter. An Algorithm for Suffix Stripping. *Readings in Information Retrieval*, pages 313–316, 1997.

[RD01] Antony Rowstron and Peter Druschel. Pastry: Scalable, Decentralised Object Location, and Routing for Large-Scale Peer-to-Peer Systems. In *Proceedings of the Middleware Conference*, pages 329–350, 2001.

[rdf] Resource Description Framework (RDF). `http://www.w3.org/RDF/`. Last accessed: May 2010.

[RFH+01] Sylvia Ratnasamy, Paul Francis, Mark Handley, Richard M. Karp, and Scott Shenker. A Scalable Content-addressable Network. In *Proceedings of the ACM SIGCOMM*, pages 161–172, 2001.

[RGKK09] Avi Rosenfeld, Claudia V. Goldman, Gal A Kaminka, and Sarit Kraus. PHIRST: A Distributed Architecture for P2P Information Retrieval. *Information Systems*, 34(2):290–303, 2009.

[RK06] Thomas Repantis and Vana Kalogeraki. Decentralised Trust Management for Ad-hoc Peer-to-Peer Networks. In *Proceedings of the 4th International Workshop on Middleware for Pervasive and Ad-Hoc Computing*, page 6, New York, USA, 2006. ACM.

[RP08] Paraskevi Raftopoulou and Euripides G. M. Petrakis. iCluster: A Self-organising Overlay Network for P2P Information Retrieval. In *Proceedings of the Annual European Conference on Information Retrieval*, pages 65–76, 2008.

[RPTW08] Paraskevi Raftopoulou, Euripides G. M. Petrakis, Christos Tryfonopoulos, and Gerhard Weikum. Information Retrieval and Filtering over Self-organising Digital Libraries. In *Proceedings of the European Conference on Research and Advanced Technology for Digital Libraries*, pages 320–333, 2008.

[RV03] Patrick Reynolds and Amin Vahdat. Efficient Peer-to-Peer Keyword Searching. In *Proceedings of the Middleware Conference*, pages 21–40, 2003.

[RWHB95] Stephen E. Robertson, Steve Walker, and Micheline Hancock-Beaulieu. Large Test Collection Experiments on an Operational, Interactive System: Okapi at TREC. *Information Processing and Management*, 31(3):345–360, 1995.

[SC03] Luo Si and Jamie Callan. A Semi-supervised Learning Method to Merge Search Engine Results. *ACM Transactions on Information Systems*, 21(4):457–491, 2003.

[Sch01] Rüdiger Schollmeier. A Definition of Peer-to-Peer Networking for the Classification of Peer-to-Peer Architectures and Applications. In *Proceedings of the IEEE International Conference on Peer-to-Peer Computing*, pages 101–102, 2001.

[SFWC04] Natalia Stakhanova, Sergio Ferrero, Johnny S. Wong, and Ying Cai. A Reputation-Based Trust Management in Peer-to-Peer Network Systems. In *Proceedings of the Workshop on Security in Parallel and Distributed Systems*, pages 510–515, 2004.

[SGG02] Stefan Saroiu, Krishna P. Gummadi, and Steven D. Gribble. A Measurement Study of Peer-to-Peer File Sharing Systems. *Multimedia Computing and Networking*, 2002.

[SJCO02] Luo Si, Rong Jin, Jamie Callan, and Paul Ogilvie. A Language Modeling Framework for Resource Selection and Results Merging. In *Proceedings of the Eleventh International Conference on Information and Knowledge Management*, pages 391–397, New York, USA, 2002. ACM.

[SKT08] Junghwa Shin, Taehoon Kim, and Sungwoo Tak. TrustRRep: An Improved Reputation Management Scheme for Reliable Resource Sharing in Peer-to-Peer Networks. In *Proceedings of the Asia-Pacific Network Operations and Management Symposium*, pages 112–122, 2008.

[SLZ+07] Gleb Skobeltsyn, Toan Luu, Ivana Podnar Zarko, Martin Rajman, and Karl Aberer. Web Text Retrieval with a P2P Query-driven Index. In *Proceedings of the 30th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 679–686, New York, USA, 2007. ACM.

[SLZ+09] Gleb Skobeltsyn, Toan Luu, Ivana Podnar Zarko, Martin Rajman, and Karl Aberer. Query-driven Indexing for Scalable Peer-to-Peer Text Retrieval. *Future Generation Computer Systems*, 25(1):89–99, 2009.

[SMK+01] Ion Stoica, Robert Morris, David Karger, Frans Kaashoek, and Hari Balakrishnan. Chord: A Scalable Peer-To-Peer Lookup Service for Internet Applications. In *Proceedings of the 2001 ACM SIGCOMM Conference*, pages 149–160, 2001.

[sop] SopCast:P2P Internet TV. `www.sopcast.com/`. Last accessed: May 2010.

[Ste86] R. E. Steuer. *Multiple Criteria Optimization: Theory, Computation and Application.* John Wiley, New York, 546 pp, 1986.

[STS+04] Girish Suryanarayana, Richard N. Taylor, Girish Suryanarayana, Richard N. Taylor, Girish Suryanarayana, and Richard N. Taylor. A Survey of Trust Management and Resource Discovery Technologies in Peer-to-Peer Applications. *ISR Technical Report UCI-ISR-04-6*, 2004.

[SUP04] Ali Aydn Selcuk, Ersin Uzun, and Mark Resat Pariente. A Reputation-Based Trust Management System for P2P Networks. In *Proceedings of the 2004 IEEE International Symposium on Cluster Computing and the Grid*, pages 251–258, Washington, DC, USA, 2004. IEEE Computer Society.

[SWY75] G. Salton, A. Wong, and C. S Yang. A Vector Space Model for Automatic Indexing. *Communications of the ACM*, 18(11):613–620, November 1975.

[SXL05] Mudhakar Srivatsa, Li Xiong, and Ling Liu. TrustGuard: Countering Vulnerabilities in Reputation Management for Decentralised Overlay Networks. In *Proceedings of the 14th International Conference on World Wide Web*, pages 422–431, 2005.

[SYG05] William Sears, Zhen Yu, and Yong Guan. An Adaptive Reputation-Based Trust Framework for Peer-to-Peer Applications. *IEEE International Symposium on Network Computing and Applications*, pages 13–20, 2005.

[SYYW03] ShuMing Shi, Jin Yu, GuangWen Yang, and DingXing Wang. Distributed Page Ranking in Structured P2P Networks. In *Proceedings of the International Conference on Parallel Processing*, page 179, Los Alamitos, CA, USA, 2003. IEEE Computer Society.

[SZA05] Gayatri Swamynathan, Ben Y. Zhao, and Kevin C. Almeroth. Decoupling Service and Feedback Trust in a Peer-to-Peer Reputation System. In *Proceedings of the International Symposium on Image and Signal Processing and Analysis*, pages 82–90, 2005.

[TD04] Chunqiang Tang and Sandhya Dwarkadas. Hybrid Global-Local Indexing for Efficient Peer-to-Peer Information Retrieval. In *Proceedings of the Symposium on Networked Systems Design and Implementation*, pages 211–224, 2004.

[TD07] Jing Tian and Yafei Dai. Understanding the Dynamic of Peer-to-Peer Systems. In *Proceedings of the Sixth International Workshop on Peer-to-Peer Systems*, 2007.

[TDX04] Chunqiang Tang, Sandhya Dwarkadas, and Zhichen Xu. On Scaling Latent Semantic Indexing for Large Peer-to-Peer Systems. In *Proceedings of the Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 112–121, 2004.

[trea] Text REtrieval Conference (TREC) 10g. `http://ir.dcs.gla.ac.uk/test_collections/wt10g.html`. Last accessed: May 2010.

[treb] Text REtrieval Conference (TREC). gov2. http://ir.dcs.gla.ac.uk/test_collections/gov2-summary.htm. Last accessed: May 2010.

[trec] TREC English Test Questions (Topics). http://trec.nist.gov/data/testq-eng/. Last accessed: May 2010.

[TXD03] Chunqiang Tang, Zhichen Xu, and Sandhya Dwarkadas. Peer-to-Peer Information Retrieval Using Self-organising Semantic Overlay Networks. In *Proceedings of the SIGCOMM Conference*, pages 175–186, 2003.

[TXM03] Chunqiang Tang, Zhichen Xu, and Mallik Mahalingam. pSearch: Information Retrieval in Structured Overlays. *Computer Communication Review*, 33(1):89–94, 2003.

[TZWC06] Huirong Tian, Shihong Zou, Wendong Wang, and Shiduan Cheng. A Group Based Reputation System for P2P Networks. In *Autonomic and Trusted Computing*, pages 342–351, 2006.

[VF95] Charles L. Viles and James C. French. On the Update of Term Weights in Dynamic Information Retrieval Systems. In *Proceedings of the Fourth International Conference on Information and Knowledge Management*, pages 167–174, New York, USA, 1995. ACM.

[WB05] Hans Friedrich Witschel and Thomas Böhme. Evaluating Profiling and Query Expansion Methods for P2P Information Retrieval. In *Proceedings of the 2005 ACM workshop on Information Retrieval in Peer-to-Peer Networks*, pages 1–8, New York, USA, 2005. ACM.

[Wit08a] Hans F. Witschel. Ranking Information Resources in Peer-to-Peer Text Retrieval: An Experimental Study. In *Proceeding of the 2008 ACM Workshop on Large-Scale Distributed Systems for Information Retrieval*, pages 75–82, New York, USA, 2008. ACM.

[Wit08b] Hans Friedrich Witschel. Global Term Weights in Distributed Environments. *Information Processing and Management*, 44(3):1049–1061, 2008.

[WM09] Le-Shin Wu and Filippo Menczer. Diverse Peer Selection in Collaborative Web Search. In *Proceedings of the 2009 ACM Symposium on Applied Computing*, pages 1709–1713, New York, USA, 2009. ACM.

[WN09] Yufeng Wang and Akihiro Nakao. Poisonedwater: An Improved Approach for Accurate Reputation Ranking in P2P Networks. *Future Generation Computer Systems*, page Article in Press, Available online May 2009.

[WS06] Kevin Walsh and Emin Gün Sirer. Experience with an Object Reputation System for Peer-to-Peer File-sharing. In *Proceedings of the 3rd Conference on Networked Systems Design & Implementation*, pages 1–1, Berkeley, CA, USA, 2006. USENIX Association.

[WWJ00] Kent Seamons William Winsborough and Vicki Jones. Automated Trust Negotiation. In *Proceedings of the DARPA Information Survivability Conference and Exposition*, pages 88–102. IEEE Press, 2000.

[XC96] Jinxi Xu and W. Bruce Croft. Query Expansion Using Local and Global Document Analysis. In *Proceedings of the 19th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 4–11, New York, USA, 1996. ACM.

[XC98] Jinxi Xu and James P. Callan. Effective Retrieval with Distributed Collections. In *Proceedings of the 21st International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 112–120, 1998.

[XC99] Jinxi Xu and W. Bruce Croft. Cluster-Based Language Models for Distributed Retrieval. In *Research and Development in Information Retrieval*, pages 254–261, 1999.

[XKY04] Jun Xu, Abhishek Kumar, and Xingxing Yu. On the Fundamental Tradeoffs between Routing Table Size and Network Diameter in Peer-to-Peer Networks. *IEEE Journal on Selected Areas in Communications*, 22(1):151–163, 2004.

[XL04] Li Xiong and Ling Liu. PeerTrust: Supporting Reputation-Based Trust for Peer-to-Peer Electronic Communities. *IEEE Transactions on Knowledge and Data Engineering*, 16(7):843–857, 2004.

[XSD+08] Quanqing Xu, Heng Tao Shen, Yafei Dai, Bin Cui, and Xiaofang Zhou. Achieving Effective Multi-term Queries for Fast DHT Information Retrieval. In *Proceedings of the 9th International Conference on Web Information Systems Engineering*, pages 20–35, Berlin, Heidelberg, 2008. Springer-Verlag.

[Yag88] Ronald R. Yager. On Ordered Weighted Averaging Aggregation Operators in Multi-criteria Decision Making. *IEEE Transactions on Systems, Man, and Cybernetics*, 18(1):183–190, 1988.

[YDRC06] Yong Yang, Rocky Dunlap, Mike Rexroad, and Brian F. Cooper. Performance of Full Text Search in Structured and Unstructured Peer-to-Peer Systems. In *Proceedings of the IEEE International Conference on Computer Communications*, 2006.

[YMA09] Haibo Yu, Tsunenori Mine, and Makoto Amamiya. Agent-Community-Based P2P Semantic MyPortal Information Retrieval System Architecture. *Embedded Computing*, 3(1):63–75, 2009.

[YV07] Lu Yan and Sebastien Venot. Peer-to-Peer Media Streaming Application Survey. In *Proceedings of the International Conference on Mobile Ubiquitous Computing, Systems, Services and Technologies*, pages 139–148, Washington, DC, USA, 2007. IEEE Computer Society.

[YW03] Ting Yu and Marianne Winslett. Policy Migration for Sensitive Credentials in Trust Negotiation. In *Proceedings of the 2003 ACM Workshop on Privacy in the Electronic Society*, pages 9–20, New York, USA, 2003. ACM.

[YWS01] Ting Yu, Marianne Winslett, and Kent E. Seamons. Interoperable Strategies in Automated Trust Negotiation. In *Proceedings of the 8th ACM conference on Computer and Communications Security*, pages 146–155, New York, USA, 2001. ACM.

[ZCLL04] Haizheng Zhang, W. Bruce Croft, Brian Neil Levine, and Victor R. Lesser. A Multi-Agent Approach for Peer-to-Peer Based Information Retrieval System. In *Proceedings of the International Conference on Autonomous Agents and Multiagent Systems*, pages 456–463, 2004.

[ZDN] ZDNet. `http://www.zdnet.com/`. Last accessed: May 2010.

[ZG00] Xiaolan Zhu and Susan Gauch. Incorporating Quality Metrics in Centralised/distributed Information Retrieval on the World Wide Web. In *Proceedings of the 23rd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 288–295, 2000.

[ZH07] Runfang Zhou and Kai Hwang. PowerTrust: A Robust and Scalable Reputation System for Trusted Peer-to-Peer Computing. *IEEE Transactions on Parallel and Distributed Systems*, 18(4):460–473, 2007.

[ZHC08] Runfang Zhou, Kai Hwang, and Min Cai. GossipTrust for Fast Reputation Aggregation in Peer-to-Peer Networks. *IEEE Transactions on Knowledge and Data Engineering*, 20(9):1282–1295, 2008.

[ZHTW08] Christian Zimmer, Johannes Heinz, Christos Tryfonopoulos, and Gerhard Weikum. P2P Information Retrieval and Filtering with MAPS. In *Proceedings of the 2008 Eighth International Conference on Peer-to-Peer Computing*, pages 84–85, Washington, DC, USA, 2008. IEEE Computer Society.

[ZKJ01] Ben Y. Zhao, John D. Kubiatowicz, and Anthony D. Joseph. Tapestry: An Infrastructure for Fault-tolerant Wide-area Location and Routing. Technical Report UCB/CSD-01-1141, UC Berkeley, April 2001.

[ZL06] Haizheng Zhang and Victor Lesser. Multi-agent Based Peer-to-Peer Information Retrieval Systems with Concurrent Search Sessions. In *Proceedings of the Fifth International Joint Conference on Autonomous Agents and Multi-agent Systems*, pages 305–312, New York, USA, 2006. ACM.

[ZL08] Huanyu Zhao and Xiaolin Li. H-Trust: A Robust and Lightweight Group Reputation System for Peer-to-Peer Desktop Grid. In *Proceedings of the 28th International Conference on Distributed Computing Systems Workshops*, pages 235–240, Washington, DC, USA, 2008. IEEE Computer Society.

[ZRL+07] Ivana Podnar Zarko, Martin Rajman, Toan Luu, Fabius Klemm, and Karl Aberer. Scalable Peer-to-Peer Web Retrieval with Highly Discriminative Keys. In *Proceedings of the International Conference on Data Engineering*, pages 1096–1105, 2007.

[ZS05] Jiangong Zhang and Torsten Suel. Efficient Query Evaluation on Large Textual Collections in a Peer-to-Peer Environment. In *Proceedings of the Fifth IEEE International Conference on Peer-to-Peer Computing*, pages 225–233, Washington, DC, USA, 2005. IEEE Computer Society.

[ZS06] Ivana Podnar Zarko and Fabrizio Silvestri. Workshop Report. In *Proceedings of the CIKM 2006 Workshop on Information Retrieval in Peer-to-Peer Networks*, 2006.

[ZTW07] Christian Zimmer, Christos Tryfonopoulos, and Gerhard Weikum. MinervaDL: An Architecture for Information Retrieval and Filtering in Distributed Digital Libraries. In *Proceedings of the European Conference on Research and Advanced Technology for Digital Libraries*, pages 148–160, 2007.

[ZYKG05] Demetrios Zeinalipour-Yazti, Vana Kalogeraki, and Dimitrios Gunopulos. Exploiting Locality for Scalable Information Retrieval in Peer-to-Peer Networks. *Information Systems*, 30(4):277–298, 2005.

[ZYKG07] Demetrios Zeinalipour-Yazti, Vana Kalogeraki, and Dimitrios Gunopulos. pFusion: A P2P Architecture for Internet-Scale Content-Based Search and Retrieval. *IEEE Transactions on Parallel and Distributed Systems*, 18(6):804–817, 2007.