ABSTRACT

PEOPLE-SEARCH: SEARCHING FOR PEOPLE SHARING SIMILAR INTERESTS FROM THE WEB

by
Quanzhi Li

On the Web, there are limited ways of finding people sharing similar interests or background with a given person. The current methods, such as using regular search engines, are either ineffective or time consuming. In this work, a new approach for searching people sharing similar interests from the Web, called People-Search, is presented. Given a person, to find similar people from the Web, there are two major research issues: person representation and matching persons. In this study, a person representation method which uses a person's website to represent this person's interest and background is proposed. The design of matching process takes person representation into consideration to allow the same representation to be used when composing the query, which is also a personal website. Based on this person representation method, the main proposed algorithm integrates textual content and hyperlink information of all the pages belonging to a personal website to represent a person and match persons. Other algorithms, based on different combinations of content, inlink, and outlink information of an entire personal website or only the main page, are also explored and compared to the main proposed algorithm. Two kinds of evaluations were conducted. In the automatic evaluation, precision, recall, F and Kruskal-Goodman Γ measures were used to compare these algorithms. In the human evaluation, the effectiveness of the main proposed algorithm and two other important ones were evaluated by human subjects. Results from both evaluations show that the People-Search algorithm integrating content and link

information of all pages belonging to a personal website outperformed all other algorithms in finding similar people from the Web.

PEOPLE-SEARCH: SEARCHING FOR PEOPLE SHARING *SIMILAR INTERESTS FROM THE WEB*

by
Quanzhi Li

A Dissertation
Submitted to the Faculty of
New Jersey Institute of Technology
in Partial Fulfillment of the Requirements for the Degree of
Doctor of Philosophy in Information Systems

Department of Information Systems

May 2007

APPROVAL PAGE

PEOPLE-SEARCH: SEARCHING FOR PEOPLE SHARING SIMILAR INTERESTS FROM THE WEB

Quanzhi Li

Dr. Yi-fang Brook Wu, Dissertation Advisor — Date
Assistant Professor of Information Systems, NJIT

Dr. Murray Turoff, Committee Member — Date
Distinguished Professor of Information Systems, NJIT

Dr. Julian Scher, Committee Member — Date
Associate Professor of Information Systems, NJIT

Dr. Vincent Oria, Committee Member — Date
Associate Professor of Computer Science, NJIT

Dr. Il Im, Committee Member — Date
Assistant Professor of The School of Business, Yonsei University, Korea

BIOGRAPHICAL SKETCH

Author: Quanzhi Li

Degree: Doctor of Philosophy

Date: May 2007

Graduate Education:

- Doctor of Philosophy in Information Systems,
  New Jersey Institute of Technology, Newark, NJ, 2007

- Master of Science in Computer Engineering,
  University of Minnesota, Twin Cities, USA, 2001

Major: Information Systems

Publications:

Li, Quanzhi and Wu, Y. B. People Search: Searching People Sharing Similar Interests from the Web, Journal for American Society of Information Science and Technology (JASIST), forthcoming.

Li, Quanzhi and Wu, Y. B. (2006). Identifying Important Concepts from Medical Documents, Journal of Biomedical Informatics, 39(6), pp. 668-679.

Wu, Y. B., Li, Quanzhi, Bot, R. and Chen, X. (2006). Finding Nuggets in Documents: A Machine Learning Approach, Journal for American Society of Information Science and Technology (JASIST), 57(6), pp. 740-752.

Li, Quanzhi and Wu, Y. B. (2006). Information Mining -Integrating Data Mining and Text Mining for Business Intelligence, in Proceedings of AMCIS'06, Acapulco, Mexico.

Li, Quanzhi (2006). Searching People Sharing Similar Interests from the Web, ACM SIGIR Doctoral Consortium, Seattle, WA.

Li, Quanzhi, Wu, Y. B., Bot, R. and Chen, X. (2005). Automatically Finding Significant Topical Terms from Documents, in Proceedings of AMCIS'05, Omaha, Nebraska.

Li, Quanzhi, Wu, Y. B., Chen, X. and Bot, R. (2005). Extracting Conceptual Terms from Medical Documents, in Proceedings of AMCIS'05, Omaha, Nebraska.

Wu, Y. B., Li, Quanzhi, Bot, R. and Chen, X. (2005). Domain-specific Keyphrase Extraction, in Proceedings of ACM Fourteenth Conference on Information and Knowledge Management (CIKM05). Bremen, Germany.

Wu, Y. B., Bot, R., Chen, X. and Li, Quanzhi (2005). Improving Concept Hierarchy Development for Web Returned Documents Using Automatic Classification, in Proceedings of International Conference on Internet Computing (ICOMP'05), Las Vegas, NV.

Wu, Y. B., Li, Quanzhi, Chen, X. and Bot, R. (2005). Learning by Examples: Identifying Key Concepts from Text Using Pre-Defined Inputs, in Proceedings of International Conference on Artificial Intelligence (ICAI'05), Las Vegas, NV.

Bot, R., Wu, Y. B., Chen, X. and Li, Quanzhi (2005). Generating Better Concept Hierarchies Using Automatic Document Classification, in Proceedings of ACM Fourteenth Conference on Information and Knowledge Management (CIKM05). Bremen, Germany.

Li, Quanzhi, Wu, Y. B., Bot, R., and Chen, X. (2004). Incorporating Document Keyphrases in Search Results, in Proceedings of AMCIS'04, New York City, NY.

Wu, Y. B., Li, Quanzhi, Bot, R., and Chen, X. (2004). KIP: A Keyphrase Identification program with learning Function, in Proceedings of International Conference on Information Technology (ITCC), Las Vegas, NV.

Bot, R., Wu, Y. B., Chen, X., and Li, Quanzhi (2004). A Hybrid Classifier Approach for Web Retrieved Documents Classification, in Proceedings of International Conference on Information Technology (ITCC), Las Vegas, NV.

This dissertation is dedicated to my beloved family

ACKNOWLEDGMENT

First I would like to express my deepest appreciation to my dissertation advisor, Dr. Yi-fang Brook Wu, for her valuable guidance, support, encouragement, kindness and enthusiasm during the course of this work and my entire graduate study at NJIT. I would also like to thank the rest of my thesis committee members, Dr. Murray Turoff, Dr. Julian Scher, Dr. Il Im and Dr. Vincent Oria, for their insightful comments, hard questions and actively participating in my committee.

Also thanks to my fellow graduate students at Earth Lab and CoLab for interesting discussions and being fun to be with during my graduate study at NJIT.

Finally, I also want to thank all the people who graciously participated in my experiment and provided valuable suggestions and comments.

TABLE OF CONTENTS

TABLE OF CONTENTS
(Continued)

TABLE OF CONTENTS
(Continued)

TABLE OF CONTENTS
(Continued)

Chapter **Page**

TABLE OF CONTENTS
(Continued)

Chapter **Page**

LIST OF TABLES

Table **Page**

LIST OF TABLES
(Continued)

Table **Page**

LIST OF FIGURES

Figure **Page**

LIST OF FIGURES
(Continued)

Figure **Page**

CHAPTER 1

INTRODUCTION

1.1 Research Motivations and Objectives

Since the inception of the World Wide Web, information has become more accessible than before. One of the popular web search needs is person/people related search: users like to search information related to a specific person or people who are specialized in a subject; they also want to find other people possessing certain interest/expertise or sharing similar interests or background with them from the Web.

Previous studies on person/people related search mainly focus on two directions: 1. "person search" -- searching web pages authored by a specific person or containing information about this person, given this person's name as the query, and 2. "people search" -- finding a list of people similar to the given one, in terms of their interests. In this study, the two concepts "person search" and "people search" are differentiated. The focus of this study is the latter.

On the Web, to find other people having similar interests, the simplest way is to browse through who's who directories or other similar directories. The problems with this method are that such directories might not be updated regularly, and the scope may be limited to only certain popular domains. Many users utilize regular search engines to find people by sending keywords to search engines and then browsing through the results to see who authored the web pages of their interest. However, regular search engines are not specialized for the task of finding similar people; users using this approach would find it laborious and ineffective. Other existing methods, such as the online dating

systems and social matching systems (discussed in Chapter 2), also have various limitations. They usually require a lot of user involvement and efforts. For example, online dating systems need to build user profiles by getting users to answer a long list of questions on topics such as their religious beliefs, professions, physical appearance, etc. Some other systems need users' browsing history to build their profiles. Furthermore, these systems are only available to the registered users and usually the searchable people in their database are limited to certain groups or domains.

This study attempts to find a people search solution that requires no manual user involvement in building searchable people profiles, is able to search people from various domains, and has access to a large body of people. The people search method proposed in this study is about specifying characteristics of a person automatically and finding other persons who share the similar characteristics with the given person. To design such a system, two major research issues need to be solved: how to represent a person and how to match persons on the Web. The first question is how to profile users - what type of information does a system use to represent its users, and how does it acquire this information? The second question is how to find matches - what is the system's model of a good match? And how does the system compute matches (Terveen & McDonald, 2005)? To address these problems, in this study, a framework for people search is first defined. Then under this framework, a main algorithm is proposed for people search, and a few other algorithms are also explored and compared to the main algorithm. Finally, an automatic evaluation and a user study are conducted to evaluate these algorithms to see if the main algorithm is the best one.

1.2 Research Overview

In this study, an approach to representing a person online is proposed: a person's personal website (personal home page) is used to represent this person. A person's personal website usually contains information about this person's background and interest, and it can be used to represent this person. Many previous studies have indicated that a person's personal website can be considered as this person's identity and self-presentation on the Web, and it can be used to represent this person (De Saint-Georges, 1997; Doring, 2002; Papacharissi, 2002a & 2002b). For example, a professor's website usually has information about her/his research interest, publications, research projects, etc., which well represents this professor. There is a huge number of personal websites available. Therefore, by using personal websites to represent people, there will be a tremendous number of people available for search. The owners of these websites are from various domains. This means, by using personal websites as searchable people profiles, people available for search are diversified, unlike certain social matching or online dating systems, where people available for search are limited to only certain domains or registered users. Personal websites already exist online, so users of the system do not need to explicitly provide their information to the system, in order for them to be searched by other users.

To solve the people search problem, a framework is first defined to completely specify what kind of information may be used in person representation and the matching persons process. The following attributes together define the proposed people search framework:

- A person's personal website can be used to represent this person, in terms of his/her interests and background.

- If persons can be represented by their own websites, then the search query can be represented by the personal website of a person as well. Therefore, in the people search process, a query is a personal website, and the returned results are a list of personal websites. This means searching people for a given person becomes searching people's personal websites for a given personal website.

- All documents belonging to a person's website may be used to compare two persons.

- Both content and link information of the web pages of a person's website could be used for representing this person.

Based on this people search framework, the ultimate research question becomes: given a person's website, how can the system find other people's websites semantically related to the provided one?

Under this person representation method, a main algorithm is proposed, and 13 other algorithms are also explored and compared to the main algorithm. The main algorithm, called People-Search algorithm, integrates content and link information of all the pages belonging to a personal website to represent a person and match persons. The other algorithms are based on different combinations of content, inlink, and outlink information of an entire personal website or only the main page. It is hypothesized that the People-Search algorithm will outperform the other 13 algorithms. To find similarity between two personal websites, in the People-Search algorithm, the content similarity between the two sites and also the link similarity between them are first calculated, and then these two kinds of similarities are linearly combined together to get the integrated similarity between the two sites (two persons). To evaluate these algorithms and test the hypothesis, two kinds of evaluations were conducted: an automatic evaluation and a human evaluation. In the automatic evaluation, precision, recall, F and Kruskal-Goodman Γ measures were used to compare these algorithms. In the human evaluation,

the effectiveness of the People-Search algorithm and two other important ones were evaluated by human subjects. Several prototype systems were developed for the evaluations.

1.3 Research Assumptions and Scope

As mentioned before, the people search method proposed in this study should be able to search people from various domains. In other words, the system implemented based on this solution is a general purpose people search system, not a domain-dependant one. It is not designed for a specific domain or group, though it can be tailed to a specific domain or group.

People search and person search are two concepts that sometimes are used interchangeably in other places. However, in this study, they are considered two different concepts. In this study, person search refers to finding web pages related to a specific person given this person's name as the query. In this case, the query is a person's name, and usually the returned pages contain this person's name. On the other hand, people search is to find a list of people that are similar to the given one, in terms of their interests. In this study, the query of people search is a personal website (the address of a person's home page), not a person's name or a set of terms, which are usually used in the regular search engines. This study addresses the problem of people search, not person search.

In this study, a person's website is used to represent this person, in terms of his/her interests or background. Although previous studies have pointed out that personal websites can be used as people's online identities and to represent people online, in

reality, some people's websites may not have sufficient information to represent them on the Web. For example, a person's website has only one main page (home page), and this main page contains only contact information. In this case, this person's website does not have enough information to describe and represent this person. Therefore, the chance that this website is retrieved as a relevant returned hit to a query is very small. Therefore, one assumption of this study is that a person's website contains enough information to represent this person.

The framework and algorithms proposed in this study require that the system should be able to access the personal websites on the Web and index them. One issue raised by this requirement is how to automatically obtain these personal websites. In the prototype systems, the personal websites are collected from the ODP personal website directory (Open Directory Project, http://www.dmoz.org/), which is not enough for a commercial system. It is better for a practical system to be able to automatically, continuously crawl the Web to index personal websites. How a web crawler intelligently distinguishes personal websites from non-personal websites (e.g., university web sites, company web sites, etc.) exceeds the scope of this study. In this study, it is assumed that people's websites can be obtained from the Web. How to automatically, intelligently crawl the web to index only personal websites will be one of the future research topics.

1.4 Dissertation Organization

The remainder of this dissertation is organized in the manner described below. Chapter 2 presents related previous studies. Chapter 3 describes the proposed framework and the 14 algorithms in detail. Chapter 4 presents the evaluation method, including dataset

selection, the automatic evaluation, and the human evaluation. Chapter 5 presents experimental results and data analysis. The limitations of this study, contributions and future research directions are discussed in Chapter 6.

CHAPTER 2

BACKGROUND INFORMATION AND RELATED STUDIES

The topic of this dissertation falls in the field of web-based information retrieval (IR), which involves the traditional IR and the new development of web search technologies. To find similar personal websites, the content similarity and link similarity between two sites need to be calculated. The algorithms for content similarity calculation are basically based on the traditional IR techniques, which mainly deal with textual information. The link similarity calculation is mainly based on the link analysis of the Web. Therefore, part of the focus of this chapter will be on previous studies of the traditional IR and the web-based IR. Section 2.1 presents existing methods and current development of online person/people related search. In Section 2.2, previous studies about personal websites are described. In Section 2.3, the background knowledge and previous studies of the traditional IR are introduced. Section 2.4 describes some popular link-based web search algorithms and the link-based similarity methods used for web pages.

2.1 Related Research on Person/People Search

In this section, previous studies about people/person search are introduced. Section 2.1.1 introduces the commercial systems related to people/person search. Previous research on person search is described in Section 2.1.2. Social matching systems are discussed in Section 2.1.3.

2.1.1 Existing Commercial Systems Related to People/Person Search

The two terms *people search* and *person search* have been used in many places, and their meanings may vary in different contexts. They have been used in many places. The following two paragraphs describe two kinds of online commercial systems that also use these two terms.

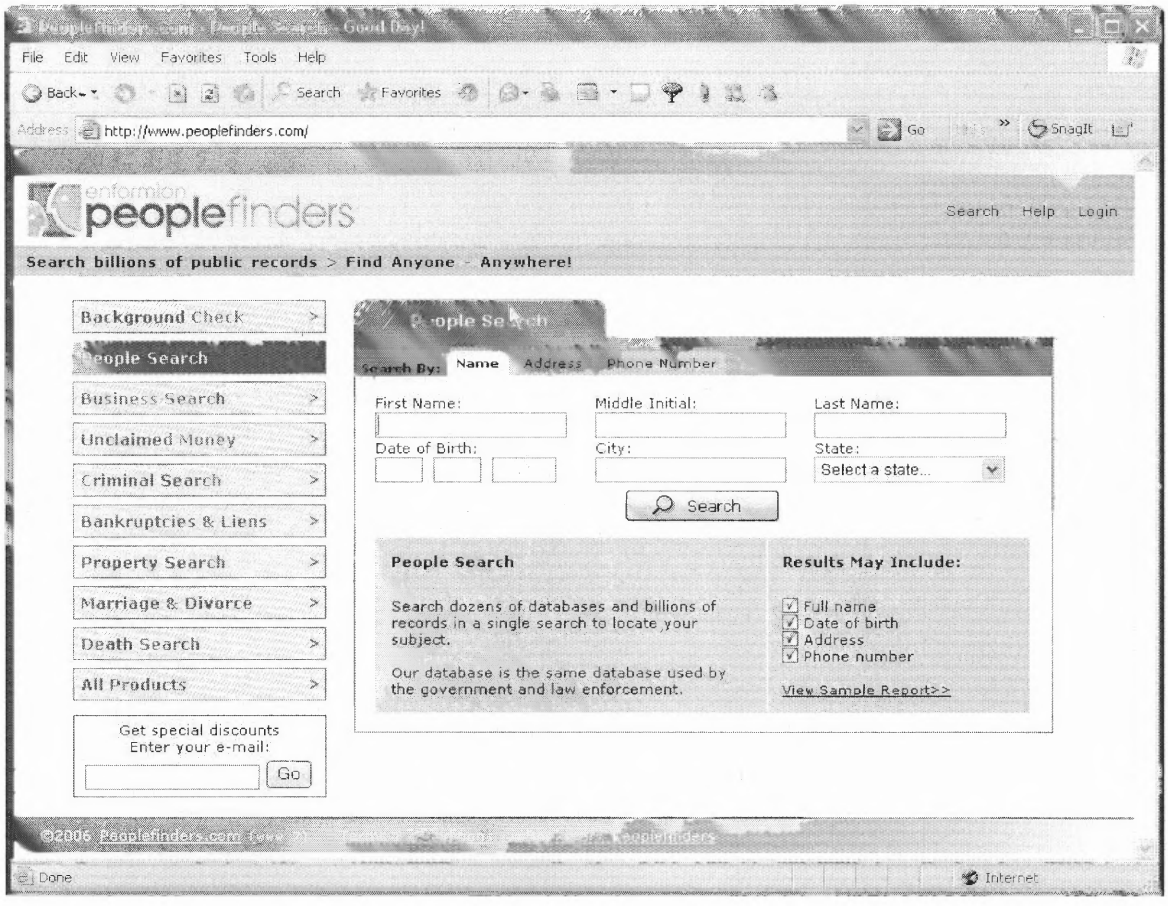

Figure 2.1 A screenshot from Peoplefinder.

In the first kind of online commercial systems, the meanings of the two concepts are different from that used in this study. In these systems, these two terms mean searching for a person's public record, such as phone number, criminal record, marital status, email address, etc., given a person's name or other information, such as a person's address. These systems usually obtain the structured personal information from some

agents and store them in their database for customers to search. The personal information is not automatically collected from the Web; they are manually classified and well structured, and are limited only to the structured personal records. Examples of such kind of commercial systems are: http://www.publicbackgroundchecks.com/, http://www.peoplefinders.com/, http://www.usa-peoplesearch.com/, and http://people.yahoo.com/. Figure 2.1 is a screenshot taken from http://www.peoplefinders.com. Although these types of systems also use the term *person search* or *people search*, they have different meanings from that used in this study.

Another kind of commercial systems are the ones like online dating systems. Sometimes these systems also use the term *people search* or *person search*. Examples of such kind of commercial systems are: http://www.americansingles.com/, http://www.eharmony.com/, http://www.match.com/ and http://personals.yahoo.com/. To use these systems, users must first pay and register. These systems usually ask the registered users to provide some personal information, such as gender, height, weight, age, location, education and hobbies. This type of structured personal information is stored in their system database. Users can find other people they are interested in by searching the database based on some search criteria. The matching persons process in these systems is basically done by searching their databases using structured SQL queries. The personal information in their system database is not collected from the Web, but provided by the registered users. Figure 2.2 is a screenshot taken from http://personals.yahoo.com. The search criteria are listed in the left side frame. Usually, this kind of system also has a search criterion called "interests." Users can use it to

roughly specify some interests that should be possessed by people they are interested in. Figure 2.3 shows the items in the "interests" category in http://personals.yahoo.com.

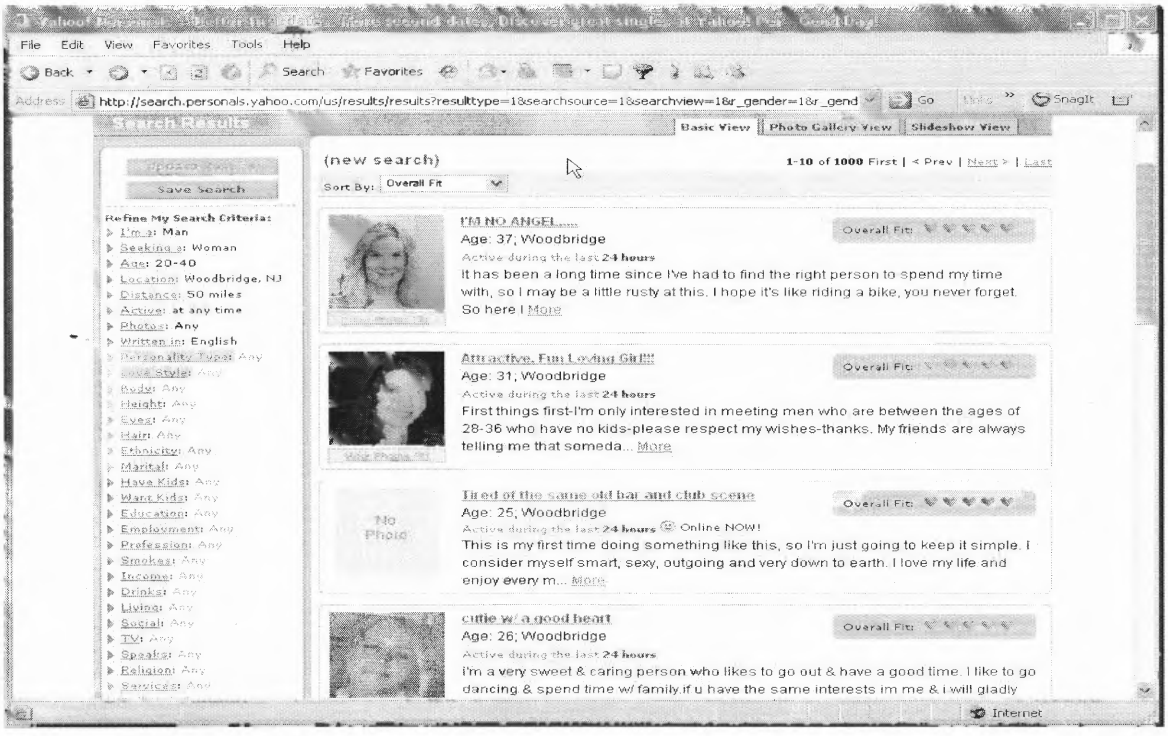

Figure 2.2 A screenshot taken from Yahoo Personal.

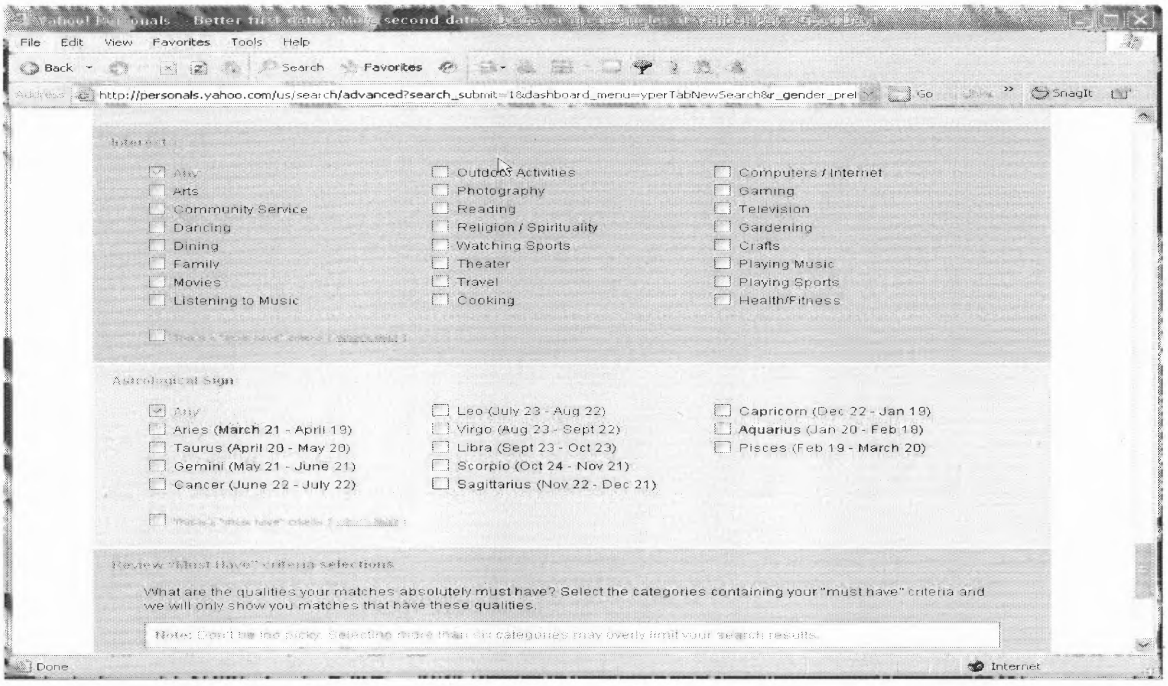

Figure 2.3 Interests criteria in Yahoo Personal.

The limitations of this kind of commercial systems are: in order to search or be searched by others, people need to subscribe and manually provide their personal information to these systems, which requires lots of efforts. Moreover, only the registered users can search other people in the system. These systems will not search the Web to find people/person that users are interested in, instead they use the information stored in their database, which are explicitly and manually provided by users.

2.1.2 Person Search

People search and person search are two different concepts in this study. Person search tries to find a list of pages related to a person given this person's name as the query. Previous studies about person search are described below.

WebHawk is a person search system developed by Wan et al. (2005). They claim that, given a list of pages obtained by submitting a person's name to a search engine, their system can cluster these pages into different clusters (groups), each of which corresponds to one specific person. The main purpose of their systems is to handle the multi-referent ambiguity problem. Their system has three steps:

1. A filter is used to remove pages containing no information about any person (called junk pages). The junk pages are retrieved because person names may refer to non-person entities, such as products or companies. The features used to identify junk pages include: lexical features, such as title words and words adjacent to query words (person's name), and query-relevant possessive features, such as the occurrences of "'s" after query words.

2. A clustering technique is used to group the remaining pages into different clusters, each for one specific person. The agglomerative clustering algorithm is employed to cluster the filtered pages from the previous step. Cosine similarity measure (defined in Section 2.3.2) is used to calculate the similarity between two pages.

3. An extractor is used to extract useful information for each cluster, mainly the name and title of the person corresponding to the cluster. For a query name, the final

search results of WebHawk are a list of groups of returned hits. Each group has a name and a title to represent it.

The person search problem can also be treated from the aspect of disambiguating web appearance of people in a social network (Bekkerman and McCallum, 2005). In Bekkerman and McCallum's study, the query is a set of person names, instead of a single person name. These persons are in the same social network, e.g., in the same email list or online community. Their study tries to find pages that are related to any person of the social network concurrently, excluding the pages related to namesakes of the people of the social network. Two kinds of methods are used to solve the web appearance disambiguation problem. The first one is based on the link structure of the returned pages, assuming that the pages related to the people in the same social network are interconnected in some way. Two web pages are considered linked to each other if they have same inlinks (these two pages are pointed to by the same web page), same outlinks (these two pages point to the same web page), or one page can be reached within three link hops from the other. The interconnected pages are clustered into the same cluster. The largest cluster is considered the central cluster. Then the distances between this central cluster and other clusters are calculated. The cosine similarity measure and TF.IDF term weighting method (defined in Section 2.3.1) are used in the clustering process. The central cluster and clusters close to it are considered relevant clusters. All the pages in these clusters are considered relevant pages.

Their second method is based on agglomerative/conglomerative double clustering model. In this method, only the terms of the returned pages are used; links are not considered. This method is based on the text classification principle - similar documents have similar distribution over words, while similar words are similarly distributed over

documents. Based on this principle, the returned pages are clustered into clusters. The clustering process is iterated until a predefined number of clusters are obtained. They choose the largest cluster as the relevant one and all the pages in it are considered as relevant pages to the query names. Their experiment shows that these two methods perform equally. Bekkerman and McCallum's methods address the problem of web appearance of people in a given social network. These people are already known to each other. Their methods do not address the problem of how to find new people to a given social network.

An earlier study about person search is the work by Shakes et al. (1997). They develop a system called Ahoy!. The query for this system is also a person's name, plus some other kinds of personal information of this person, such as phone number, email address, or name of the institution this person belongs to. Based on the provided information (name, email, etc.), Ahoy! can find home pages for this person. They primarily use heuristics and pattern matching techniques for recognizing URLs of homepages.

The studies discussed in this section mainly focus on person search. Their purpose is to find web pages related to a given person, instead of finding a list of people having similar interests with the given person, which is different from the goal of this study.

2.1.3 Social Matching Systems

One of the closest areas to the people search problem is social matching. Social matching systems bring people together in both physical and online spaces, based on certain criteria; therefore, social matching systems are a kind of people search system. They can

increase social interaction and foster collaboration among users within organizational intranet or on the Internet. Terveen and McDonald (2005) survey several social matching approaches. They point out that social matching systems are not a well-established field, and there is not even a generally recognized name for it. In the following paragraphs, previous studies on social matching systems are introduced.

The space of social matching systems has been explored by both commercial systems and research prototypes. The online dating systems discussed in Section 2.1.1 are also a kind of social matching system (Terveen and McDonald, 2005). They ask people to provide information about themselves and what they are looking for in a romantic partner, and then apply algorithms to find matches and provide ways for matched people to communicate.

In the remainder of this section, three kinds of social matching research prototypes are introduced. The first type of social matching research prototype matches people based on their social relationship and information need. ReferralWeb (Kautz et al. 1997) and Expertise Recommender (McDonald and Ackerman, 2000; McDonald, 2001) are two examples. To match people based on their specific information needs, both systems need two kinds of profiles, one representing a person's expertise and the other concerning social relations between persons.

ReferralWeb (Kautz et al. 1997) mines the public web documents to identify expertise for users. Names are first identified from documents, and then the main topics of these documents are identified to represent the expertise of people whose names appear in these documents. ReferalWeb uses the co-occurrence of names in documents as evidence of a relationship. The main sources to obtain the co-occurrence information of

names are: lists of co-authors in technical papers and citations of papers, exchanges between individuals recorded in Netnews archives and organization charts. The construction process of relationships is incremental. When a user registers with the system, the system uses search engines to retrieve documents related to this user. Names of other people are extracted from these documents, too; these people will have a relation with this user, since they co-occur in the same document. Gradually, a whole relationship network of users will be built in the system. When a registered user (information seeker) wants to find an expert in a certain field from the system, the system will use knowledge of the topic expertise to identify people who are likely to be able to answer the question, and it will also try to find the experts with the closest social relation to the information seeker. This is based on the assumption that a person usually considers the answer more credible if it is given by others with closer social relation with him/her (e.g., a friend of a friend).

The Expertise Recommender system (McDonald and Ackerman, 2000; McDonald, 2001) works within an origination. It acquires knowledge about who knows what by mining the work products and byproducts within an organization, such as project reports, technical support documents, and software source control systems. In Expertise Recommender system, the social network information is obtained manually. Human experts familiar with the searchable people in the system database are employed to identify the social relations between people in the organization. Similar to ReferralWeb, when an information seeker wants to find an expert for a specific topic, Expertise Recommender system will try to find an expert who also has the closest social relationship to the information seeker.

The two social matching systems described above focus on information seeking, and they both need to build user profiles in advance. In their systems, in order to search others and to be searched by others, people need to register. Another limitation of the first system is that the search space is limited to only the users who have used the system, and it has the "cold start" problem (at the beginning, there are very few users in the system available for search). The second system can only work for people in the same organization, and its social relationships are built manually.

Another kind of social matching system focuses on helping people navigate information spaces to find desired facts and providing information about who can help if users need information beyond what is recorded in the system (Terveen and McDonald, 2005). Examples of such systems are the Designer Assistant (Terveen et al. 1995), Answer Garden (Ackerman, 1994; Ackerman and McDonald, 1996) and PHOAKS (Hill and Terveen, 1996; Terveen et al. 1997). They are briefly introduced below.

The Designer Assistant is for software developers. It works within an origination or special interest group (Terveen et al. 1995). It organizes software design knowledge as a hierarchical series of pieces of advices. Each piece of advice in this software development knowledge repository is tagged with an owner, an individual in the organization who is most familiar with this specific topic. Users of the system can traverse the hierarchical structure to get advice, and if they want more information they may contact the owner of that piece of advice. Answer Garden (Ackerman, 1994; Ackerman and McDonald, 1996) is similar to the Designer Assistant. It organizes knowledge around a hierarchy of questions and answers. Questions and answers are also tagged with experts who are in charge of this topic. Users can get more information about

a specific topic by contacting the expert who is responsible for it. PHOAKS stores recommended web pages from Usenet news messages (Hill and Terveen, 1996; Terveen et al. 1997). When a user is browsing a web page, it shows the user the message in which this web page was recommended and the contact information of the person who recommended this page. If a user is interested in a web page and wants to discuss similar topics with the recommender, they may make contacts.

The main limitation of these three social matching approaches (Designer Assistant, Answer Garden and PHOAKS) is that they mainly work for users in certain domains or organizations. They are similar to the online community systems. A lot of human efforts are needed in order to build such a system (e.g., in Answer Garden, experts are needed to design questions and provide answers).

The last type of social matching systems are more close to the people search problem addressed in this study, focusing on finding people sharing similar interests. Unlike the systems described above, this kind of systems is independent of a specific user information-seeking request. The matching process is based on users' interests. These systems infer users' interests from the record of their browsing activities (Terveen and McDonald, 2005). I2I (Budzik et al. 2002) is one example.

I2I (Budzik et al. 2002) tries to find users sharing similar interests based on the documents they have viewed or are reading. It attempts to provide informal collaboration by providing its users with opportunities to become aware of the activities of others who share common interests, as represented by the documents they interact with. It tries to build communities of common interest on the fly. For example, if a user is reading a document in I2I system, the system can find other users who are browsing

similar documents and recommend them to this user on the fly. I2I uses the document (or several documents if they are opened by one user at the same time) a user is browsing to represent this user's interests. It calculates the similarity between two active users to see if they have similar interests. The similarity calculation involves the documents currently viewed by the two users. I2I exploits only the content information of a document to represent a user's interests. The terms appearing in a document are used to represent the content of this document. Term stemming is applied to get the stems of terms, and TF.IDF method (defined in Section 2.3.1) is used to calculate term weights. Cosine similarity (defined in Section 2.3.2) is used to calculate the similarity between two documents. After similar people are found for a user, the user can have an active chat with them or contact them by other ways.

Other systems similar to I2I are Kalsa (Svensson et al. 2001) and LiveMaps (Cohen et al. 2002). Kalas is a social navigation system for recipes. It organizes recipes into collections, and users can gather around a collection they are interested in and chat with each other. LiveMaps also matches people based on their browsing behaviors. Users who are browsing the same web page can chat with each other.

The systems mentioned above (I2I, Kalas and LiveMaps) are similar to online community services. One of the limitations of this kind of systems is: in order to chat with other people or to be searched by others, a user must register and join the online community, and users' activities also need to be recorded. In other words, in such systems, a registered user is both a user of the system and a searchable item in system database.

In their study, Adamic and Adar (2003) analyze information stored in personal home pages and mailing lists to predict relationships between individuals. Three kinds of information are extracted from home pages: text, inlinks, and outlinks. These three kinds of information and mailing lists, which are obtained from a mailing list server, are used to predict whether one person is a friend of another. These sources of information are compared to see which one is the most predictive. The users (home page owners and people appearing in the mailing lists) are ranked based on their similarity to a given person to predict whether they are friends of this person. Similarity is measured by sum of the number of items two users have in common, including words, links, and mailing lists. They evaluate their methods using home pages in the domains of MIT.edu and Stanford.edu, and they find that inlinks are the most predictive in finding friends, followed by mailing lists, outlinks, and finally text.

In this section, previous studies related to people/person search have been introduced. Their applications and limitations are also discussed. The current methods used for people search are either not specialized for this task, or have some limitations. Therefore, there is a need to find an innovative solution for people search. This study tries to find a people search solution that requires no user involvement in building searchable people profiles, is able to search people from various domains, and has access to a large, diverse body of people. The proposed people search solution tries to satisfy these requirements. It uses personal websites to represent people. In the next section, previous studies on personal websites are introduced.

2.2 Related Studies on Personal Websites

Personal websites (or personal home pages) have been a focus of many studies. De Saint-Georges (1997) provides a tentative definition of a personal website as a "presentation of the self in digital (hypertextual) form, authored by one individual, and which (i) emphasizes a person (minimally, by a name or picture); and/or (ii) a person's current activities; and/or (iii) professional experience; and/or (iv) displays a person's interest (in the body of the text and/or through hyperlinks to other sites)." Some other previous studies about personal websites have discussed what kinds of web sites are considered personal websites in their studies, though they do not explicitly give a definition for personal websites. In Papacharissi's study (2002a & 2002b), the personal websites are picked up from some personal home page providers, such as Yahoo! Geocities. If the chosen websites are affiliated with or constructed by a commercial organization or other institutions, then they are excluded from their study. Weaver (2000) conducts a survey to determine if the viewing of personal web pages is part of a reference librarian's duties. In the study, the websites that are "wholly under the control of individuals, and not functioning as official library pages" are considered as personal websites. Dominick (1999) defines a personal website as a website published and maintained by an individual who may or may not be affiliated with an institution.

In this people search study, the adopted definition of De Saint-Georges's is used - A website is considered a personal website if it is thought to have been authored by a person with the purpose of presenting that person's interests and persona (Narsesian, 2004; De Saint-Georges, 1997). This definition is similar to the one defined by Dominick (1999). Dominick's definition is also used by Doring (2002). Doring points out that the

ownership status of personal websites can almost always be determined from page titles and headings (e.g., "Home page of George W. Bush," "Tom's World," "Jerry's Little Palace," and "My Home Page"). Those websites maintained by organizations, institutions or formal groups are to be distinguished from personal websites. Doring also points out that the ownership status of the "personal" website is independent of how private or intimate the contents actually are. If a person's website is restricted to professional activities, then according to the definition suggested here, there is still a personal (that is person-related) website.

Personal websites have some advantages over the commercial-style sites: for topics where commercial rewards cannot or have not yet been reaped, information is more likely to be found on personal sites of enthusiasts; and personal websites provide the ability to contact the authors (Narsesian, 2004). Many studies have examined the characteristics and attributes of personal websites, and the purposes that personal websites serve for their authors. They are introduced in the following paragraphs.

Papacharissi (2002a & 2002b) conducts a study that tries to examine the purposes and motives as to why people create personal home pages. The research design of the study involves an online survey of web page authors and a content analysis of respondents' web sites. One thousand personal websites are randomly chosen from Geocities, EarthLink Homepages and other homepage services providers. Their authors are contacted, and 260 of them finally answer the survey. Besides other findings, the study results show that personal websites owners create their home pages primarily for information, entertainment, self-expression and social interaction. Thirty four percent of the respondents say when creating home pages they focus on their general interests (this

is the most popular response). The authors think they create an "online portrait" of themselves by using personal home pages. Based on the study results, Papacharissi points out that a person's personal website is the presentation of the self in virtual life.

By analyzing 319 personal websites, Dominick (1999) attempts to examine how web page authors use personal home pages to project themselves to the rest of the world. Among other things, this study tries to see the differences between real self-presentation and 'virtual' self-presentation by analyzing the demographics and the contents of the pages. This study has the following findings: strategies of online self-presentation by using personal websites are employed with the same frequency as they are in interpersonal settings; gender differences in online self-presentation are consistent with research findings from social psychology; personal websites are useful as an information resource; and seventy five percent of the examined personal websites contain information about either "likes" or "dislikes." A final conclusion from the results of this research is that personal websites are a tool of self-expression, which are used by their authors to create pages tailored to a specific audience.

Bates and Lu (1997) carry out a survey with 114 personal home pages. The aspects they want to look at include the purpose of home pages, their structure and physical features. The study finds that even though certain elements and design features are present often on personal websites, there is no one feature which is ever-present. For example, although personal email address is the most frequent element appearing in personal websites, only 92.1% of the surveyed personal websites have it. Forty five percent of the site owners think the primary purpose of their websites is to present their professional capabilities or background to others, which is the most popular response.

According to Buten's survey (1996), the top two reasons for creating personal home pages are "means of expression" (account for 49% of the responses) and "distribute information to people I don't know with similar interests (43%)." Buten's survey also gives an interesting statistic about the expected audiences of personal websites. Sixty three percent of respondents think that "browsers"(i.e., random surfers) would visit their websites, and 52% of them think that "fellow enthusiasts for a topic/hobby" would visit intentionally.

According to Doring (2002), the application of theoretical constructs by social scientists who are interested in studying personal home pages revolves around the fact that personal websites involve personal identity construction and self-presentation issues, via computer-mediated communication. As a medium of self-expression and self-construction, a personal website represents important and beneficial variants of our intrapersonal communication. It is especially well suited for an elaborate self-presentation, and, as a kind of rich and evocative source of information, it can surpass other types of self-presentation (Doring, 2002; Chandler, 1998; Karlsson, 1998; Miller, 1995; Wynn & Katz, 1997). With computer-mediated communication, control of one's verbal statements is enhanced: people can present themselves more deliberately and selectively than in face-to-face scenarios, and are not placed under intense pressures of confrontation and pressures to act. This can encourage heightened self-disclosure and authenticity on the one hand (e.g., self-outing on one's own home sites), but also abet conscious masquerade and deception on the other (e.g., omissions on one's home sites). Based on the systematic review of the diverse theoretical and empirical literature on

personal home pages, Doring considers a personal website an effective personal identity and self-presentation on the Web (2002).

Other studies also point out that the contents of the personal homepages reflect a range of purposes, but a unifying purpose is that of self-presentation (Miller, 1995; Erickson, 1996; Walther, 1996; Vazire and Gosling, 2005). Dillon and Gushrowski (2000) consider a personal website the "first truly digital genre." Chandler and Roberts-Young's (1999) study shows that not all personal homepages are overtly or primarily about their authors, but such pages do reveal their authors' interests to the readers.

The previous studies described above show that a personal website can be considered as a person's online identity and its content reflects a person's interests. Based on these studies, it is reasonable to use a person's personal website to represent the person on the Web. This person representation method is used in this people search study.

Based on the proposed method, to find similar people, the content-based similarity and the link-based similarity between personal websites need to be calculated. This requires the traditional information retrieval (IR) technology, on which the textual content similarity is based, and the web-based IR technology, on which the link similarity is based. In the next section, previous studies about traditional IR technologies are described. Precious studies of the web-based IR will be discussed in Section 2.4

2.3 Traditional Information Retrieval

IR is a broad interdisciplinary field which draws on many other disciplines. It stands at the junction of many established fields, such as information science, natural language processing, artificial intelligence, computer human interaction, library science, and

computer science. From a broad point of view, IR is the art and science of searching for text, sound, images or data within database, Intranet or Internet. From a narrow point of view, when talking about IR, people refer to searching for textual information. In this study, the narrow point of view is used.

IR systems are often related to objects and queries. Queries are formal statements of information needs that are entered to an IR system by the user. An object is an entity which stores information. The entities to be searched are usually textual documents, such as web pages, news articles, and scientific papers.

In the IR research community, there are two different points of view about the relationship between queries and documents (Salton, 1989; Brauen, 1969; Korfhage, 1997). Some researchers consider that a query is also a document, in spite of its difference from the real documents, since both of them can be used to address topics and represent users' interests. A more common reason for this is that in many instances it is possible to identify a specific document as being of interest to the user and to use that document as the model for the query. Other researchers take the opposite point of view. They think a query should not be considered as a document, since it is sufficiently different from the documents (Bollmmann-Sdorra & Raghavan, 1993). The distinction between the two kinds of views about the relationship between queries and documents affects different retrieval methods. If a query is considered as a document, then the retrieval process would be a process of matching between one document and another. In contrast, if a query is considered to be different from documents, then the retrieval process becomes a mapping process between a query and documents. A query can be distinct from the documents being retrieved in many ways:

- A query may not satisfy the normal syntax rules
- Most queries are very brief
- Word frequency, which is usually used to indicate the importance of a word in a regular document, is barely useful for queries, since most of the words in a query appear only once.

Usually, a query can be in one of two forms: a sentence or a list of terms. Many query models and matching processes have been proposed and implemented in real retrieval systems in the last three decades. The two most popular ones are Boolean query model (also called Boolean retrieval model, and sometimes Boolean matching process) and vector space model. The Boolean model is briefly introduced here, and the vector space model, on which the proposed people search solution is based, is described in detail in Section 2.3.1.

The Boolean retrieval model is the simplest and earliest one of the retrieval methods. It is based on the logic of Boolean algebra. The query terms are joined together using AND, OR or NOT Boolean operators (Heaps, 1978). Many old retrieval systems are based on the Boolean retrieval model, and now most retrieval systems still have Boolean retrieval functions, though natural language queries are much more popular.

Despite the effectiveness and simplicity of Boolean queries, this method has a number of problems: most ordinary users are not well trained in Boolean algebra, and the composed Boolean queries may not reflect what they want; the user has to have some knowledge about the search topic for the search to be efficient, e.g., a wrong word in a query could make a relevant document non-relevant; the strict interpretation required by Boolean queries often exclude information that is relevant to users' interests; and the retrieved documents are all equally ranked with respect to relevance, though some

systems may rank the returned documents based on the frequencies of the matched terms in the documents (Korfhage, 1997).

Nowadays, natural language queries are becoming widely used. A natural language query is a query that is expressed using normal conversational syntax. Users can phrase their query as if they are making a spoken or written statement to another person. A natural language query is different from a Boolean query. Unlike a Boolean query, there are no conventions or syntax rules for users to learn in natural language queries. Users may enter a query in the form of a sentence or question, or just a set of keywords. A natural language query is usually treated as a short document due to its nature described above, and most systems will treat the relationship between a natural language query and the documents being searched as a document-to-document relationship. With the popularity of the natural language query, and the increasing demand for effective document classification, clustering and other textual document processing activities, the vector space model (or vector model) has been used more and more.

The algorithms proposed in this study use the vector space model to calculate the content similarity between two personal websites. In Section 2.3.1, the vector model is described in detail. In Section 2.3.2, the similarity measures used for the vector space model are discussed.

2.3.1 Vector Space Model

In a vector model, each document is represented by a vector of terms, or an ordered list of terms. The underlying set of terms is the same for both the vector model and the Boolean query model. The main differences between the vector model and the Boolean query

model are the term representation method and the approaches of measuring the similarity between a query and a document (Korfhage, 1997). For the Boolean query model, the terms are usually represented by their presence and absence in the document. For the vector model, usually the terms are represented by some measures showing their degree of importance in the document. In the Boolean query model retrieval, the similarity between a query and a document or between two documents is based upon the presence of terms in both the documents and the query. However, in the vector model retrieval, the similarity between a query and a document or between two documents is calculated using more complicated measures.

In order to reduce the complexity of the documents and make them easier to process, a document has to be transformed from the full text version to a format suitable for processing. Representing documents using vectors is the most accepted and commonly used method in IR systems and other fields. In the vector space model, each document vector is an element in the vector space. Each item of the vector represents a term from the document. The term may be a single word or a phrase. The vector space model relies on the premise that the meaning of a document can be derived from the document's constituent terms (Salton, 1989). Sometime, people also call a document vector a *bag-of-words (BOW).* In the past decade, a lot of efforts have been put on attempting to come up with a document representation which is richer than the simple *bag-of-words*. However, despite the numerous attempts to introduce more sophisticated techniques for document representation, the vector model method remains very effective and popular. A document vector *d* can be expressed as follow:

$$d = \{(w_1, t_1), \ldots, (w_k, t_k)\},$$

where w_i ($1 \leq i \leq k$) is a term from this document, and t_i is a non-negative value denoting the degree of importance of term w_i in this document, more often, the document vector is represented as: $d = (t_1, \ldots, t_k)$, where t_i ($1 \leq i \leq k$) represents the degree of importance of term i in this document. Similarly, a query can also be represented by the above form (Belkin and Croft, 1987). Each unique term in the document collection corresponds to a dimension in the vector space. A key point of successfully using the vector model is to maintain dimensional compatibility. This means the system must be designed to ensure that the comparison between two documents or between a query and a document must be based on comparing the same terms.

The vector model, by placing terms, documents, and queries in a term-document space and computing similarities between the queries and documents, allows the returned documents to be ranked according to the similarity measure used. Unlike lexical matching techniques, such as the Boolean query model, which provide no ranking or a very crude ranking scheme, the vector space model is able to automatically guide the user to documents more conceptually similar and of greater use than other documents (Letsche and Berry, 1997).

The performance of any retrieval system based on the vector space model depends highly on how well the documents are represented. In the following subsections, three important factors affecting the effectiveness of the vector retrieval models are described: term weighting, stop words removal and term stemming. The performance of the algorithms proposed in this study will also be affected by these three factors. The measures used to calculate the similarities between a query and a document or between two documents are discussed in Section 2.3.2.

2.3.1.1 Term Weighting. Assigning weights to document terms in a document vector is a complex process. The term weighting for the vector space model has entirely been based on term statistics. A term may be a single word or a multi-word phrase. Many approaches have been proposed for assigning weights to document terms. The two most popular are the TF method and the TF.IDF method.

The TF method. TF means term frequency. It refers to the absolute frequency of a term in a document. It is reasonable to assume that the more frequently that a term occurs in a document, the more important it is to this document (except the non-content-bearing words, such as *that, the, on* and so on. They are also called stop words, which will be discussed later). This is the assumption of this method. Using this method, a term's weight in a document vector is its absolute frequency in this document.

The TF.IDF method. The problem with the TF method is that it does not take into account the document collection size and characteristics. To a specific document, if a term appears in many documents in the collection, then it may not be so important to this document as another term which appears in only few documents but has the same frequency as the first term in this document. Terms appearing in many documents in the collection should be given a lower weight compared with terms appearing in only a few documents. IDF means Inverse Document Frequency. The TF.IDF method assumes that the importance of a term to a specific document decreases with the number of documents the term appears in increases (Salton, 1983). Experimentally, it has been shown that this document discrimination factor, IDF, leads to a more effective retrieval, i.e., an improvement in precision and recall (Salton and Buckley, 1996).

In the TF.IDF method, the frequency of a term in a document is weighted by the number of documents in the collection that contain the term. A term's TF.IDF weight in a document is its absolute frequency in this document multiplied by the value of IDF. To explain how to calculate TF.IDF, the following four variables are defined (Korfhage, 1997).

N: the number of documents in the document collection,

d_k: the number of documents containing the term *k*,

f_{ik}: the absolute frequency of term k in document *i*, and

w_{ik}: the weight of term k in document *i*.

The IDF, inverse document frequency, is defined as:

$$IDF = \log_2 (N/ d_k)+1 = \log_2 N - \log_2 d_k + 1$$

The ratio d_k /N is the fraction of documents in the collection containing the term. The weight of term *k* in document *i* is:

$$w_{ik} = TF * IDF = f_{ik} * (\log_2 N - \log_2 d_k + 1)$$

The TF.IDF weight of a term in a document is its frequency multiplied by a factor depending logarithmically on the proportion of the documents containing that term in the collection. This formula shows that the importance of a term in a document increases when the frequency of this term in the document increases, and decreases when the number of documents containing this term increases.

In previous studies (Lee et al. 1997; Salton and Buckley, 1996; Zobel and Moffat, 1998), different weight schemes have been investigated, and the best results, based on recall and precision, are obtained by using term frequency with inverse document

frequency method, the TF.IDF method. The TF.IDF weighting method is used in the algorithms for estimating term weights.

2.3.1.2 Stop Words Removal. The words in a document can be roughly divided into two categories, the content-bearing words and the non-content-bearing words. Content-bearing words refer to the words conveying topical information. Non-content-bearing words are also called stop words. Stop words are the frequently occurring, insignificant words, for example, the words *the, are, that,* and *into*. These words have a very high frequency in the documents, and in any measure depending on term frequencies, they diminish the impact of frequency differences among less common words. These words carry little information by themselves. If they are not removed, they may result in a quite large amount of unproductive processing. Usually, when processing a document, the stop words are ignored and discarded.

Precious studies show that the most common 300 common words in English may account for 50% or more of any given text (Kucera and Francis, 1967; Korfhage, 1997). There are two kinds of stop words lists, the general one and the subject dependent one. Besides the general one, usually a domain-dependant one is also exploited for a specific domain. Removing stop words will reduce the size of a document vector. The size of the document vector can be reduced more by selecting a subset of most important words according to some criteria, for example, selecting the important words based on TF.IDF values.

2.3.1.3 Term (Word) Stemming. One challenge for every text-processing task is that a word may occur in many different forms. For example, design, designs, designed, and designing all have the same basic form and the same meaning. If a query term is

"designs," it is very possible that the documents containing "designed" are also relevant. It is clearly undesirable if the system treats them as two different words and returns the documents containing only "designs." One method to address this problem is to use a term stemming algorithm (word stemming). Word stemming strips off the word endings, reducing them to a common stem. In the above example, all the four word forms share the same stem "design." The word stemming algorithm can find the stem for the four different word forms. Word stemming brings the various forms of a word together, and results in a higher frequency count for this word. It is used in many applications involving text processing. The most famous two word stemming algorithms are the Lovins algorithm (Lovins, 1968) and the Porter stemming algorithm (Porter, 1980). In this study, word stemming is done by combining a lexical database, called WordNet (Fellbaum, 1998), with the modified Porter's stemming algorithm.

2.3.2 Similarity Measures for the Vector Space Model

When the vector space model is used, two types of similarity measures are used most: the distance-based measure and the angular-based measure. Distance-based measure is based on the philosophy that documents close together in the vector space are likely to be highly similar. Angular-based measure is based on the philosophy that documents in the same direction are likely to be highly similar. Besides these two measures, there are also some other kinds of similarity measures for the vector space model, but they are less used and their performance is not as good as the distance measure or angular-based measure. Examples of other similarity measures are Jaccard coefficient, Overlap formulation, Dice formulation and Inner product (Zobel and Moffat, 1998).

Because a query is also considered as a document in the vector model, the measures described below apply to the similarity calculation between two documents as well as between a query and a document. Given a query, the documents can be ranked based on their similarity with the query. The more similar a document is to this query, the more relevant it is. Documents that are more similar to the query will be ranked higher. Usually the similarity values are normalized, having values between 0 and 1. The distance-based measure and angular-based measure are discussed below.

2.3.2.1 Distance-based Similarity Measure. This kind of measure evaluates how close two documents are in the document space. As mentioned before, a document *D* is represented in the vector model as: $D = (t_{1, \ldots,} t_k)$, where t_i $(1 \leq i \leq k)$ is a non-negative value denoting the degrees of importance of term *i* in this document. t_i can be based on any term weighting schema, such as the TF or TF.IDF method. It is important to mention here that the actual distance between two documents (more specifically, distance between two n-dimensional vectors) is actually a dissimilarity measure. The bigger the distance the more dissimilar the documents are. The similarity of two documents is inversely proportional to the distance between them. The most widely used distance based metrics are L_p metrics (Korfhage, 1997). Suppose there are two documents, D_1 and D_2,, then the General L_p *formula* is:

$$L_p(D_1, D_2) = \left[\sum_{i}^{N} \left| d_{1i} - d_{2i} \right|^p \right]^{1/p}$$

where:

N – vector dimensionality, the number of unique terms in the vector space.

d_{1i} , d_{2i} – The value at the document *1* or *2*'s vector position ***i.***

p – parameter (see below).

When p=1, this measure is called ***Manhattan Distance*** (also called City-block).

When p= 2, it is called ***Euclidean Distance***

When p= ∞, it is called ***Maximal Direction Distance***.

The Euclidean Distance is more popular than the other two. It corresponds to the ordinary straight-line distance.

2.3.2.2 Angular-based Similarity Measure. The angular-based similarity measure is also called cosine similarity measure. It is not a distance measure, rather is based on the cosine of the angle between two document vectors. Two documents are considered similar if they are situated along the same direction in the document space, starting from the origin. Angular measure does not consider the distance of each document from the origin, but only the direction. It is possible to have documents that are similar under the cosine model even if under the distance based model they are very dissimilar, being situated far apart from each other. The formula for a cosine measure is (Rijsbergen, 1979; Wilkinson and Hingston, 1991):

$$\sigma(D_1, D_2) = \frac{\sum_{k=1}^{N} (d_{1k} \times d_{2k})}{\sqrt{\sum_{k=1}^{N} (d_{1k})^2} \times \sqrt{\sum_{k=1}^{N} (d_{2k})^2}}$$

where:

D_1, D_2 – Document *1* and document *2*

d_{1k}– the weight of term k in document D_1

d_{2k}– the weight of term k in document D_2

k – From *1* to N

In mathematical terms, the cosine similarity measure is the inner product of the two documents vectors, normalized by their lengths. The value of cosine similarity measure ranges from 0 for the lowest similarity to 1 for the highest. In this study, the cosine similarity measure is used for calculating the content-based similarity between two personal websites. It is also used to calculate the link similarity between two personal websites.

In this section, the background information of the traditional information retrieval and previous studies related to the algorithms proposed in this study have been introduced. Previous studies and related background knowledge about web link analysis are presented in the following section.

2.4 Web-based Information Retrieval

The Internet is growing with an increasing rate, and it brings new opportunities and challenges to the field of information retrieval. Traditional information retrieval only deals with textual information. Now Internet has introduced a new concept to IR, web link. During the last decade, a lot of research efforts have been put into combining web linkage analysis with the traditional IR. This forms the web-based IR. In this section, previous studies on web-based information retrieval are introduced. Web-based information retrieval exploits the link structure of the Web, as well as the textual content of the web pages. Most of the research on web-based retrieval mainly focuses on the link analysis. This part is divided into two small sub sections. Section 2.4.1 presents several famous previous studies on link-based web search. These studies have greatly affected

the research and development of the WWW; many other studies on web-based retrieval are extensions of these studies. In Section 2.4.2, related studies on web page similarity calculation, especially the link similarity, are presented.

2.4.1 Link-based Web Search Algorithms

With the growth of the importance of the Web as an information source, more and more attention has been paid on how to find information of interest by exploiting the link structure of the Web. Because web links are created by people for the purpose of guidance to the related pages, inside the link structure, a lot of valuable information about the relationship between web pages exists. The Web is considered as a graph with web pages as nodes and hyperlinks as edges. The graph-based (link-based) algorithms do not rely on the textual contents of web pages; they mainly work on the link structure.

The linked-based algorithms are mainly used for two goals: 1. to rank the results from the content-based search algorithms, or 2. to search for the similar web pages by themselves. In the first case, the search system first returns a list of documents that are considered relevant to the query based on a content-based search algorithm, and then the link analysis algorithm ranks these returned documents according to their popularity or similarity to the query, based on the evidence obtained from the link analysis on these documents. In the second case, the system will directly search documents relevant to the query from the Web, based on the link analysis.

A link analysis-ranking algorithm starts with a set of web pages. Depending on how this set of pages is obtained, algorithms can also be divided into two types: *query independent* algorithms, and *query dependent* algorithms. In the first case, these web

pages are obtained without the consideration of the query; usually all the web pages on the Web are obtained if possible. In the second case, only the pages that are related to the query in certain degree are obtained.

Many link-based algorithms have been proposed in previous studies. The two most famous ones are PageRank (Brin and Page, 1998; Page et al. 1998) and HITS (Kleinberg, 1999). These two algorithms and other influential ones, most of which are extensions of these two, are introduced below. Some of the algorithms presented in this section may not be directly related to the methods proposed in this study. The reasons they are also presented here are that these algorithms have great influence on the development of the Web, and the algorithms related to this study are extensions of these algorithms or have been affected by their ideas.

2.4.1.1 PageRank. The PageRank algorithm used by the Google search engine is one of the most successful link-based ranking methods (Brin and Page, 1998; Page et al. 1998). It approximates a page's authority through the sum of its neighbors' authorities. A page's authority is the probability that the surfer visits it. In Google, the search results based on content-based search algorithm are ranked according to their authorities obtained from PageRank. PageRank models the web surfer's random walk behavior over the (entire) web graph. It is independent from a specific topic. Therefore, the PageRank value of a page is a *global*, topic-independent importance rating of that page on the Web.

Consider the Web as a directed graph, where the nodes represent web pages, and the edges between nodes represent the links between web pages. Suppose N is the number of nodes in this graph, P_i represents the set of pages page i links to, and Q_i is the set of pages linking to page i. Also suppose that a web surfer is jumping from web page

to web page, and which link to follow at each step is chosen using a uniform probability. In order to avoid the effect of endless cycles and dead-ends, the surfer will occasionally jump to a random page with a probability of $1-\alpha$, which is very small. Then after a sufficient number of steps, the probability the surfer visits page j is defined by the following formula:

$$PR(j) = \frac{(1-\alpha)}{N} + \alpha \sum_{i \in Q_j} \frac{PR(i)}{P_i}$$

where *PR(j)* is the PageRank value for node j. The above equation is recursive. The final value of *PR(j)* is obtained when *PR(j)* converges. This equation shows that the PageRank value of a page grows with the importance of the pages pointing to it.

PageRank algorithm has some downsides. It suffers from topic drift, which means it may return web pages with high quality, but are only peripherally related to the query. This is because, when computing PageRanks, only the link structure between pages is considered, and the contents of pages are ignored.

2.4.1.2 Google Bombing. One famous problem with Google's search algorithm is Google bomb (also called Google bombing or Google washer). Based on Google's PageRank algorithm, a page will be ranked higher if there are a lot of pages linking to that page. When users' search terms are related to this page, then this page will have a higher rank in Google's return list. This can be explained by Google's PageRank algorithm: the more pages pointing to a page, the higher rank this page has. An interesting thing is that even though this page does not contain the search terms, it may still be returned as the top 1 hit. The reason behind this is that Google also uses anchor text to represent a page. Therefore, even though a page does not contain any of the search terms, as long as the anchor texts of the links that point to this page contain the search

terms, this page will be returned as a search result. And if the number of pages that link to this page is large and these pages all use consistent anchor text, this page will have a very high rank in the returned results, even the first one. Google bombing is also called Google bomb. Usually it is a certain attempt to influence the ranking of a given page in Google's return results, often with humorous intentions. Some people also called Google bomb as "link bombing," since other search engines which heavily rely on web link structure also have similar problems. For example, a search for "miserable failure" will bring the official "George W. Bush" biography website number one and Michael Moore's official website number two on Google, Yahoo and also MSN, although none of the two websites has the term "miserable failure" in their pages. Google's responses and proposed solution for Google bomb are available at (Google, 2005a; Google, 2005b). In this study, the explored algorithms will consider both the content and link information of a person's website in calculating the similarity between two persons' websites, so the link bombing problem will not happen in the system implemented based on the approach proposed in this study.

2.4.1.3 HITS (Hyperlink-Induced Topic Search Algorithm). HITS (Kleinberg, 1999) algorithm is initially proposed to rank the search results from content-based search algorithms. HITS is a topic-specific, *local* ranking algorithm. It operates on a small part of the whole Web. By analyzing the link structure of this web subgraph and assigning hub and authority scores to its pages, the importance of each page in this subgraph is obtained. HITS is one of the milestones for the link structure research.

Kleinberg introduces the authority and hub concepts to web pages. An authority is a web page pointed by many good hubs, and a hub is a web page pointing to many web pages with high authority. HITS builds a link graph for all the search results of a specific topic or query and initializes the authority and hub values of the nodes (pages). The authority and hub values are recursively updated according to the above principle until they converged. The pages with the highest authorities are regarded as the most valuable relevant pages. The process is described in detail below. Given a query, HITS first obtains a set of pages using a traditional search engine. This set of pages is called the root set. Then this set is expanded to include all pages that link to or are linked to by the pages of the root set. Next, each page i is assigned a hub score and an authority score. The hub and authority scores are updated by the following equations (initially, hub(i) and auth(i) are set to 1):

$$hub(i) = \sum_{j \in Q_i} auth(j) \ , \ auth(i) = \sum_{j \in P_i} hub(j)$$

where P_i represents the set of pages page i links to, and Q_i is the set of pages linking to page i. The above two equations are iterated until they converge. This equation shows that a hub is a page that points to many authorities, and an authority is a page pointed to by many hubs.

HITS also has some downsides. It needs to calculate the authorities and hubs at query time, so it is not practical in a large-scale search system. HITS also has the topic drift problem; it is possible that the pages used to expand the root set may not be related to the query.

2.4.1.4 SALSA (Stochastic Approach for Link Structure Analysis). Another graph-based rank algorithm is the SALSA algorithm proposed by Lempel and Moran based on the Markov chain theory (2000). SALSA combines aspects from both HITS and PageRank, but it is mainly based on HITS. It is also a topic-specific, *local* ranking algorithm. It operates on a small portion of the whole Web. The intuition behind this is that a web page with high authority should have high probability to be visited by a random walk. It builds a link graph G for the results from the content-based search algorithm. Then, it builds a bipartite graph G0 in which each non-isolated node in G is represented by two nodes belonging to the hub side and authority side, respectively. Next, they perform two random walks which start from different sides of the bipartite graph. As authorities and hubs should be highly visible, one may expect that the authorities will be amongst the nodes most frequently visited by the random walk starting from the authority side, and the hubs will be amongst the nodes most frequently visited by the random walk starting from the hub side (Lempel and Moran, 2000). More on the relations between HITS and SALSA can be found in Borodin et al. (2001).

2.4.1.5 SimRank. The SimRank algorithm can be classified into the graph-based similarity algorithm category (Jeh and Widom, 2002). It computes a measure based on the assumption that two objects are similar if they are related to similar objects. This method builds a node-pair graph X for the link graph G. Each node in X is an ordered pair

of nodes in *G*. A node *(a, b)* in *X* points to node *(c, d)* in *X*, if in *G a* points to *c* and *b* points to *d*. It initializes the SimRank scores for all the nodes of *X* as follows: SimRank *(a, b)* is 1 if *a* and *b* are the same node; otherwise, it is 0. The SimRank scores are iteratively computed in that a node's SimRank score is the normalized sum of all the SimRank scores of the nodes pointing to it. The SimRank score of a node in *X* gives a measure of the similarity between its node-pairs in *X*.

2.4.1.6 Companion Algorithm. Dean and Henzinger derive their Companion algorithm from the HITS algorithm (1999). This algorithm is based on an observation by Kleinberg (1999) that the authority and hub method can be used not only to rank the results from content-based methods, but also to find similar web pages. Still using the query node as the seed, they apply a different way to build the link graph. Weights are not only assigned to nodes, but also to edges using the edge weighting scheme proposed by Bharat and Henzinger (1998). They performed a user study to compare their Companion algorithm with the "What's Related" algorithm in Netscape 4. The Companion algorithm performs significantly better than the "What's Related" algorithm.

The link algorithms introduced above are the ones having great impacts in the area of web search engines. In the following section, previous studies on similarity calculation for web pages, some of which are derived from the algorithms described above, are introduced.

2.4.2 Similarity Algorithms for Web Pages

Most previous studies in link-based similarity focus on finding similarity between web pages instead of websites. Although this is different from the algorithms explored in this study, they both deal with link structure.

Two web pages may be similar in terms of their semantic content, or in terms of their page structure. One main purpose of finding pages having similar structure is to detect the phishing web pages. Phishing is a criminal trick of stealing personal information by sending victims spoofed emails urging them to visit a forged web page that looks like a true one. One approach to measure the similarity between the phishing page and the target page is to use the following metrics: block level similarity, layout similarity, and overall style similarity. The goal of this study is to find personal websites that are similar in terms of their semantic contents instead of their site or page structure. Therefore, the structural comparison approach, which is usually used for phishing detection, is not used in this study.

Mainly there are three approaches to finding how similar two web pages are in terms of their semantic contents:

1. *A content-based approach.* This approach uses terms appearing in two pages to calculate the similarity between them. It relies solely on the textual information provided by authors of these two web pages, ignoring the opinions of authors of other pages (which are reflected by links between pages). This is a traditional way for finding document similarity, and is usually used for similarity calculation between documents that do not have link information to use (Korfhage, 1997; Salton, 1983). It can be applied to web pages as well as other kinds of documents, like plain text documents. The common similarity measures used for this approach have been introduced in Section 2.3.2.

2. *A link-based approach.* For web page similarity calculation, the link-based similarity approach is more suitable than the content-based. This approach considers the Web a graph with pages as nodes and links as edges, and uses the link structure to estimate similarities between nodes. Several previous studied

have explored this approach (Kleinberg, 1999; Bollacker et al. 1998; Dean and Henzinger, 1999; Jin and Dumais, 2001; Menczer, 2004; Fogaras and Racz, 2004; Jeh and Widom, 2002).

3. *An anchor-based approach.* For every link pointing to the web page under consideration, this approach uses words appearing inside or near the anchor in a web page. For example, for the link *Information Systems Department, NJIT*, "Information Systems Department, NJIT" is the anchor text of this link. It can be used to represent the page *www.is.njit.edu.* An anchor-window is used to specify the size of text around an anchor, and the text information is used to do similarity search. The idea behind this approach is that the anchor-window constitutes a hand-built summary of the target web page (Haveliwala et al. 2002; Jin and Dumais, 2001).

To find similar people from the Web, the People-Search algorithm combines the content-based approach and the link-based approach. The anchor-based approach can provide additional information to represent a web page. It is usually used when the content-based approach is not exploited. Considering that the anchor-based approach needs additional effort to obtain anchor text and its trivial contribution after the content-based approach is already exploited (Haveliwala et al. 2002), the anchor-based approach is not integrated in the proposed algorithms. The content-based approach has been discussed in Section 2.3.2. In the following paragraphs, previous studies on the link-based similarity methods are discussed.

Link-based search and similarity algorithms are based on the graph theory. The graph-based methods are first used in the bibliometrics field which studies research publications and their citation structures to estimate the importance of scientific papers and the similarity between them (Small, 1973; Kessler, 1963; Bachelor and Eaton, 1980; White and Griffith, 1980). Two basic concepts of bibliometrics are Co-citation and Bibliographic Coupling. Co-citation means two papers are referenced by the same paper, and Bibliographic coupling means two papers cite the same paper. If two documents are

co-cited by more other documents, it may indicate that these two documents are more similar. Similarly, the more documents cited by both of the two documents under comparison, the more similar these two documents might be. A famous application based on the concepts of Co-citation and Bibliographic Coupling is Citeseer, which is an autonomous web agent for automatic retrieval and identification of interesting publications (Bollacker et al. 1998). The web page's inlink and outlink structures are similar to Co-citation and Bibliographic Coupling. In this study, web pages' inlink and outlink information is used to calculate the link similarity between two personal websites.

A well-known concept related to web page similarity is search-by-example. The search-by-example approach is the most suitable approach for retrieving images: the user provides an image as a template and the system finds images that are visually similar (Rui et al. 1997). Now it is also becoming popular in retrieving the general web pages. The concept search-by-example in the web-based information retrieval means looking for pages related to a given page. To find the related pages, some algorithms must be exploited to find the similarity between the query page and target pages. Some search engines explicitly use the term "related pages" or "search by example" in their website, others may use the term "find similar pages," "page specific search," or "similar pages."

In Google, the search-by-example function is called "similar pages." In Google's result page, a link called "similar pages" is attached to each returned hit. If this link is clicked, a list of pages, which are similar to the hit to which the "similar pages" link is attached, is returned. Google's "similar pages" function (also called GoogleScout) is a typical search-by-example example. Users also can invoke the "similar pages" function from the search box by typing in "related: a given URL." The URL must be exact. In

other words, "*related:njit.edu* " and "*related:www.njit.edu*" find different results. To find the related pages, some algorithms must be exploited to find the similarity between the query page and target pages. By analyzing link connections, GoogleScout tries to find other pages similar in linkage patterns to the given page and at a similar hierarchical level with the given page (Google, 1999).

Yahoo provides a search tool, called Y!Q, that has the search-by-example function. This tool can be downloaded and installed on Internet Explore or Firefox browser (http://help.yahoo.com/help/us/ysearch/yq/index.html). Y!Q allows a user to submit all or part of a web page that the user is viewing as a search query, rather than the traditional method of typing words into a search box. Users can submit a page, a paragraph or just several sentences. They can use part of a page as the query by highlighting paragraphs or sentences of the page displaying in the browser. Y!Q analyzes the content submitted by users and extracts the most relevant terms from the submitted page, paragraphs or sentences, and then returns results to users accordingly (Sherman, 2005; Yahoo, 2005). The search results look just like normal Yahoo search results, but, at the top of the result list, the search terms extracted from the submitted page (paragraphs or sentences) are displayed in a "context selection box," with a check box next to each term. If users uncheck the box for a search term, this term will be removed from the query, and the results will be automatically updated to reflect the influence of the checked term. Y!Q still has a regular search box to allow users to input their queries. Y!Q actually provides two kinds of functions: "related search" and "contextual search." If users perform a search just by submitting the page that is being viewed, or highlighting one or more paragraphs or sentences, without typing in a query, then it is a "related

search." Y!Q will find results that are related to the query page, paragraphs or sentences. "Related search" is a kind of search-by-example – the submitted page or a portion of the page will serve as the example, and Y!Q will search web pages similar to this example. In addition to submitting a page or highlighting a portion of a page, if a query is also provided, then the search is called "contextual search." The submitted page is considered the search context, and Y!Q will find pages related to the query based on this context.

Dean and Henzinger (1999) propose two algorithms which use only the link structure of the Web to identify related web pages. The first one is called Companion, which is derived from the HITS algorithm proposed by Kleinberg (1999). The Companion algorithm and HITS algorithm have already been described in Section 2.4.1. The second one is called Cocitation. The Cocitation algorithm can find pages that are frequently co-cited with the query page. This means it finds the pages pointed to by other pages that also point to the query page. Two pages are co-cited if they have a common parent (inlink). The number of common parents of the two pages is their degree of co-citation. Sometimes there is an insufficient level of co-citation with the query page to provide meaningful results. If this is the case, in their implementation, the page corresponding to the query page's URL with one path element removed will be used to find common parents. For example, if the query page's URL is *www.abcd.com/X/Y/Z* and an insufficient number of co-cited nodes exist for this URL, then *www.abcd.com/X/Y* is used to represent the original URL of the query page and find parents (inlinks). In their study, they do not consider the common children (outlinks).

Menczer's study considers both the inlinks and outlinks when estimating the similarity between two web pages (2004). The link similarity between two pages is

defined by the Jaccard coefficient: $\sigma_1(p,q) = |U_p \cap U_q| / |U_p \cup U_q|$ where p and q are two pages and U_p is the set containing the URLs of p's inlinks, outlinks and p itself. U_q has a similar meaning as U_p. The outlinks are obtained from the pages themselves, and the inlinks are obtained from a search engine. The Jaccard coefficient measures the degree of clustering between the two pages, with a high value indicating that the two pages are similar.

In this study, the angular-based similarity (cosine similarity) is used to calculate the link similarity, as well as content similarity, between two personal websites. Menczer's Jaccard coefficient method does not consider the weight of each inlink or outlink. The consine similarity method considers the weight of each outlink and inlink. Details will be provided in Section 3.2. Other previous studies have also used the link information to estimate the similarity between two pages (Fogaras and Racz, 2004; Jin and Dumais, 2001).

2.5 Summary

In this chapter, previous studies on people/person related search are first reviewed. Then related studies on personal websites are presented. Finally, the web search technologies, including the traditional IR techniques, on which the content similarity is based, and the web-based IR techniques, on which the link similarity is based, are reviewed. In the next chapter, the research methodology of this study will be presented.

CHAPTER 3

RESEARCH METHODOLOGY

The research methodology is presented in this chapter. First, in Section 3.1, the basic definitions and notations of the important concepts used in this dissertation are introduced. Section 3.2 discusses research scopes and the main research questions, and it is followed by Section 3.3, where the proposed framework for people search is introduced. In this study, based on the proposed framework, fourteen algorithms are explored, and it is hypothesized that the proposed algorithm which integrates both the content and the link information of all web pages within a personal website will outperform other algorithms. The proposed algorithm is called the "People-Search" algorithm. It is presented in Section 3.4. The remaining 13 algorithms are introduced in Section 3.5. These algorithms have some parameters. A genetic algorithm was used to tune these parameters to obtain their optimal values. Section 3.6 introduces the genetic algorithm and how parameters were tuned. Section 3.7 introduces the architecture of the people search system.

3.1 Definitions

Person search and people search: person search is a type of search which finds pages related to a specific person given this person's name as the query. It aims at searching pages authored by a specific person or containing information about this person, and the query is this person's name. People search is to search other people that have similar interests or background with a given person. It is called "people search" because its

purpose is to find a list of people that are similar to the given one, in terms of the interests and background.

Web page and website: in this study, a web page is a single document in a website. A website holds one or more web pages.

Inlink and outlink: to web page W, an inlink is a URL of another web page which contains a link pointing to W. To web page W, an outlink is a link (URL) appearing in W which points to another web page.

Content similarity: the degree of similarity between two websites (or web pages), based on the textual content (terms appearing in them) of the two websites.

Link similarity: the degree of similarity between two websites (or web pages), based on the link information (inlinks and outlinks) of the two websites (or web pages).

Word stemming: a process which strips off the word endings, reducing them to a root form or a common stem. For example, after applying word stemming to words "designed," "designs," and "designing," they have the same root form, "design."

Stop words: words that frequently appear in a textual document but do not convey any meaning. They are also called non-content-bearing words. Examples of stop words are: "the", "of", "who", "why", etc.

Term weight: different terms have different importance in a textual unit, e.g., a document, a document collection, or a website. A term's weight is a value showing the degree of importance a term is in a textual unit. Usually a term's frequency of appearance in the document (document collection, website, etc.) or its TF.IDF value is used as its weight.

TF.IDF: a method to measure the importance of a term in a document or website (or other kinds of textual units). TF refers to a term's absolute frequency in a document or website. IDF means inverse document frequency. IDF decreases when the number of documents containing the term increases.

3.2 Research Questions

This study tries to provide a solution for people search on the web: specifying characteristics of a person and finding other persons who share similar characteristics with the given person. To design such a system, two major research issues need to be investigated: how to represent a person and how to match persons on the web. The matching process needs to take person representation into consideration to allow the same representation when composing the query, which should also represent a person. In other words, as Terveen and McDonald (2005) point out, the first issue is how to profile users - what type of information does a system use to represent its users, and how does it acquire this information? The second issue is how to compute matches - what is the system's model of a good match? and how does the system compute matches? Therefore, in this study, the following main research questions are to be answered:

Research question 1:

> *How to represent a person on the Web? This representation method should reflect this person's characteristics, and can be used for the process of matching persons.*

Research question 2:

Given the person representation method, how to find similar people from the Web for a given person? What kinds of methods/algorithms can we design?

Research question 3:

Among the possible methods/algorithms, which one performs the best?

Research question 4:

How effective is the best algorithm on ranking the returned search results?

3.3 People Search Framework and Algorithms

To solve the people search problem, a people search framework is first outlined. This framework defines the method of representing a person on the Web, and acts as the guidelines of designing algorithms for people search. The following attributes together define the people search framework:

1. A person's personal website can be used to represent this person, in terms of his/her interests and background. A person's personal website usually contains information about a person's background and interests; therefore, it can be used to represent this person. For example, a professor's website usually has information about her/his research interests, publication, research projects, etc, which can be used to represent this professor. In this study, a person's personal websites is used to represent this person. Many previous studies point out that a person's personal website can be considered as this person's identity and self-presentation on the Web, and it can be used to represent a person (Doring, 2002; De Saint-Georges, 1997; Papacharissi, 2002a & 2002b; Chandler and Roberts-Young, 1999; Dillon and Gushrowski, 2000; Vazire and Gosling, 2004). Details

about these studies are described in Section 2.2. Some other reasons that the personal website, instead of other personal information, is used to represent a person on the Web are described below.

- First, the number of personal websites online is huge. Therefore, by using personal websites to represent people, there will be a significant number of people available for search.

- Second, the owners of these personal websites are from various domains. This means, by using personal websites as profiles of searchable people, people available for search are diversified, unlike certain social matching or online dating systems, in which the people available for search are limited to only certain domains or registered users.

- Finally, all these websites can be obtained and processed without their owners' involvement. Personal websites already exist online, so users of the system do not need to explicitly provide their information to the system, in order for them to be searched by other users. Other kinds of people related search systems, such as the online dating systems, need users to individually provide their information to these systems.

2. If personal websites are used to represent persons, then the format of a query and the returned results can be defined accordingly: the search query will be a personal website as well, and the returned search results will be a list of personal websites that are relevant to the query website. This means searching similar people for a given person becomes searching people's personal websites for a given personal website. In a people search system implemented based on this framework, the input to the query box will be the home address of a person's website, and the search results will be a list of home addresses of the relevant personal websites, accompanied by the title, description and other meta information of these websites.

3. All documents belonging to a person's website, i.e., all the pages in the personal website that can be crawled by the crawler, may be used to compare two persons. In other words, pages other than hidden pages or the ones explicitly excluded from crawling by the page owner through setting some tags, such as the "robots" tag, are all collected.
4. All the textual content and link information of a web page within a person's website may be used in the similarity calculation between two persons (two personal websites).

This framework defines how to represent a person on the Web and what kind of information can be used in this representation. Under this framework, fourteen algorithms are explored. Table 3.1 lists all the 14 algorithms and the information they use to calculate the similarity between two personal websites. A symbol √ in a table cell means the corresponding algorithm will exploit the information corresponding to that table cell. These 14 algorithms cover all the possible combinations of 1. type of information: content, inlink and outlink information, and 2. unit of personal representation: an entire website and the main page. The name of each algorithm indicates what type of information it uses to match persons. If the name of an algorithm starts with "Site," it means this algorithm will use information from the entire personal website; if the algorithm name starts with "MainPage," it means this algorithm will use only information from the main page of a personal site. If an algorithm name contains term "Link," it means this algorithm will use both inlink and outlink information. For example, MainPage_Link will integrate inlink and outlink information from only the main page of

a website; Site_Content_Outlink will combine content and outlink information from an entire website, including the main page and all the sub pages.

Table 3.1 The 14 Algorithms and the Information Used in the Similarity Calculation

Algorithm Name	Information used					
	The entire personal website (Include the main page and all sub pages)			Only the main page		
	Content	Inlink	Outlink	Content	Inlink	Outlink
Site_Content_Link (The proposed People-Search algorithm)	√	√	√			
Site_Content_Inlink	√	√				
Site_Content_Outlink	√		√			
Site_Content	√					
Site_Link		√	√			
Site_Inlink		√				
Site_Outlink			√			
MainPage_Content_Link				√	√	√
MainPage_Content_Inlink				√	√	
MainPage_Content_Outlink				√		√
MainPage_Content				√		
MainPage_Link					√	√
MainPage_Inlink					√	
MainPage_Outlink						√

The Site_Content_Link algorithm is the proposed algorithm and is also called People-Search algorithm. It integrates both the content and the link information of all the web pages in a personal website. It is hypothesized that this algorithm will outperform all the other 13 algorithms. This algorithm is introduced in Section 3.4 in detail and the other 13 algorithms are introduced in Section 3.5.

3.4 The People-Search Algorithm

This section presents the main algorithm – the People-Search algorithm.

3.4.1 Integration of the Content Similarity and Link Similarity

In this section the main algorithm, the People-Search algorithm, is presented. Traditionally, in search engines or other IR related applications, when comparing two documents, only the textual contents of the two documents are considered. Usually the content-bearing terms are extracted from the documents, and they are assigned weights according to some kinds of term weighting schemes. These terms are then used to represent the two documents in the similarity calculation. This kind of document similarity is purely based on the terms appearing in documents and is usually called content similarity. There are some problems with using only the content information to calculate similarity between two documents or websites, e.g., word mismatch problem, word sense ambiguity problem and keyword spamming. Word mismatch refers to the problem that two or more words or expressions have the same meaning, and a query containing one of the words will not retrieve documents containing another, e.g., "laptop" and "notebook." Word sense ambiguity means a word may have multiple meanings, such as the word "mouse," which can mean a kind of animal, as well as a kind of computer input device. Keyword spamming means some authors intentionally place some keywords in their pages in order to increase the chances of their pages being indexed or searched. These keywords may not be related to the content of the page at all. The content similarity method purely relies on the terms appearing in the documents, so it is prone to the problems mentioned above.

Previous studies show that the web link structure might be exploited for web page similarity calculation. Hyperlinks encode a considerable amount of latent human judgment. It is assumed that the similarity information is also embedded in the link structure of the Web. The link-based methods are completely insensitive to word content, and therefore can complement content-based methods. However, there are also some problems with the link-based similarity approach. One problem is that pages with very few inlinks or outlinks will not have enough information for calculating the similarity between pages. Another problem is that link structure can also be spammed. For example, in order to increase the ranks of their pages in the search results of search engines, some authors make inter-agreements that they put outlinks in their pages to point to each other's pages. The pages belonging to these different authors may not be relevant to each other at all, but because of this kind of link spamming they are interconnected with each other. One example is the Google bombing (link bombing) problem, which is already discussed in Section 2.4.1. Both content similarity and link similarity have advantages and disadvantages, but they complement each other. Therefore, in the proposed People-Search algorithm, both of them are integrated to find the similarity between two personal websites.

Each personal website usually contains more than one page. It is very possible that using all of these pages together will give a better representation of the owner than using only one of them. In this algorithm, all the pages belonging to the same personal website will be integrated together to represent this person. These pages include the main page (home page), as well as all the sub pages. The mechanism of finding similarity between two websites is different from that of finding similarity between two web pages.

New questions will arise when comparing two websites. For example, do all the web pages of a website have the same degree of importance in computing the similarity between two websites? and how to integrate the link similarity and content similarity to obtain the best performance?

In this study, to integrate content similarity and link similarity, a linear combination of these two kinds of similarities is applied. The similarity between two personal websites is:

$$S = \beta_{site\text{-}link} S_{content} + (1-\beta_{site\text{-}link}) S_{link}$$

where $S_{content}$ is the content-based similarity value, S_{link} is the link-based similarity which combines inlink similarity and outlink similarity, and $\beta_{site\text{-}link}$ is a parameter to adjust the weights of $S_{content}$ and S_{link}. It determines the degree of importance of the two kinds of similarities in this integration. $\beta_{site\text{-}link}$ ranges from 0 to 1. When a linear combination meets the following two requirements, it is also called convex combination: (1) all the coefficients are non-negative and (2) their sum is 1. In this linear combination, both $\beta_{site\text{-}link}$ and 1- $\beta_{site\text{-}link}$ are non-negative, and their sum is 1. Therefore, this combination is also called convex combination. All the linear combinations discussed in the rest of this dissertation are also convex combination. A genetic algorithm is used to find the optimal value for $\beta_{site\text{-}link}$. The details about how to use the genetic algorithm will be discussed in Section 3.6. The value of S ranges from 0 to 1. The higher S is, the more similar the two websites are. The calculation of the content similarity and link similarity is described in Section 3.4.2 and 3.4.3

3.4.2 Content Similarity

Content similarity calculation involves the terms appearing in a personal website. Because this algorithm considers all the pages in a person's website, the terms may come from any one of these pages. The calculation is based on the vector space model (Salton 1989), which was discussed in Chapter 2. A "bag of words" (a vector containing all the important content-bearing words from a website) is used to represent the content of a person's website:

$$B = \{(T_1, W_1), \ldots, (T_k, W_k)\}$$

where B is the bag of words for this website, $T_i (1 \leq i \leq k)$ is a term from this website, and W_i is the weight for term T_i. The content similarity calculation between two websites now becomes the similarity calculation between the two word bags representing the two websites. Natural language processing techniques, such as stop words removal and word stemming, are used to preprocess the terms so that only the important content-bearing terms are included in the bag of terms. This will be discussed in detail later in this section.

3.4.2.1 Similarity Measure. There are several contented-based similarity measures to calculate the similarity between bags of words. Examples are overlap formulation, Dice formulation, inner product, Jaccard coefficient, distance measure, and cosine measure (Rijsbergen, 1979; Wilkinson and Hingston, 1991; Korfage, 1997; Zobel and Moffat, 1998). Previous studies have shown that the cosine similarity measure is the best in calculating the content similarity between two term vectors (Zobel and Moffat, 1998). Therefore, in this algorithm, the cosine similarity measure is used. Based on the cosine similarity measure, the content similarity between two personal websites is:

$$S_{content}(B_1, B_2) = \frac{\sum_{k=1}^{N}(w_{1k} \times w_{2k})}{\sqrt{\sum_{k=1}^{N}(w_{1k})^2} \times \sqrt{\sum_{k=1}^{N}(w_{2k})^2}}$$

$S_{content}$ – the content similarity between two personal websites

B_1, B_2 – bag of words for the two websites

w_{1k} – the weight of word k in bag B_1

w_{2k} – the weight of word k in bag B_2

N – total number of the unique content-bearing words in a person's website.

k – from 1 to N.

The value of $S_{content}$ rages from 0 to 1. The higher $S_{content}$ is the more similar the two websites are, in terms of their textual content. If these two websites have the same content-bearing words and the weight for the same word is the same, then $S_{content}$ will be 1. If these two sites have no common content-bearing word, then $S_{content}$ will be 0.

3.4.2.2 Choosing Content-bearing Terms. First, all the web pages within a person's website are processed. All the HTML tags, HTML comments, JavaScript code, non-alphabetic characters, and other non-content related symbols, except the tags used to identify the unusual terms described in the following several paragraphs, are removed. Then all terms are extracted from these web pages. After all the terms are extracted, a stop words removal process is applied. There are 570 stop words on the stop words list. Section 2.3.1 has introduced the stop words removal process in detail.

3.4.2.3 Word Stemming. After applying the stop words removal process, the remaining terms will be stemmed to find their root forms. One challenge for any text-processing task is that a word may occur in many different forms. For example, terms

"keep," "keeping," and "kept" all have the same basic form and the same meaning. The stemming process tries to find the root form of these words. More details about general word stemming can be found in Section 2.3.1. In this study, the existing stemming algorithms, such as the Lovins algorithm (Lovins, 1968) and the Porter stemming algorithm (Porter, 1980), are not directly used, since they do not consider the special morphological variants of some words, such as the word "knife" and "knives," or "give" and "gave." To solve this problem, in this study, the WordNet lexical database is integrated with a modified Porter's stemming algorithm. The WordNet database provides all the special morphological variants of the common English words (Fellbaum, 1998). Many previous studies have used WordNet in identifying word root forms or extracting noun phrases from textual documents (Wu et al. 2006; Li et al. 2004). In this study, the algorithm first use this database to identify the special words (e.g., kept, knives, running) and get their root forms. After looking up the WordNet database, a modified Porter stemming algorithm is used to identify the root forms for other terms. In the original Porter stemming algorithm, the stemming is a deep stemming, which generates a truncated pseudo-root instead of a word as the stem. For example, the algorithm considers the words "computer," computers," "computed" and "computing" to have the same root form, which is "comput." Too deep stemming may cause topic drift and decrease the stemming accuracy (Kantrowitz, 2000). In the modified Porter stemming algorithm, shallow stemming is used. Using the above example, the modified stemming algorithm considers that the words "computer" and "computers" have a same root form "computer," and the words "computed" and "computing" have a root form "comput."

3.4.2.4 Term Weighting Method. There are many term weighting methods for calculating a term's weight in a document, and TF.IDF is the most popular one (Salton, 1983; Salton and Buckley, 1996; Korfhage, 1997). More details about TF.IDF are in Section 2.3.1. In this study, a modified TF.IDF method, called TF.IWF, is used to represent a term's weight in a personal website. IWF is inverse "website" frequency, because the collection under consideration in this study is a set of websites, rather than a set of documents. Therefore, a term's weight in a personal website is:

$$W = TF.IWF$$

where W is the weight of a term in the website. TF is the absolute frequency of this term in this website, and IWF is the value of this term's inverse website frequency.

In determining the term weight, the following factors are also considered: terms appearing in the main page, capitalization of terms, terms in the title or meta description of a page, terms in bold, and terms in a larger font.

Terms Appearing in the Main Page. A term may appear in a website's main page, sub pages, or both. For two different terms having the same frequency, if one term appears in only the main page and the other one appears in only the sub pages, it is very possible that the first term is more important than the second one. To reflect this potential difference, a parameter called p_m is used to adjust the weight of a term appearing in the main page by adjusting its frequency. If a term appears only in the main page, then its adjusted frequency will be $TF = p_m * f_{main}$, where f_{main} is the frequency of the term in the main page. If a term appears in both the main page and the sub pages, its adjusted frequency will be $TF = p_m * f_{main} + f_{sub}$, where f_{sub} is its frequency in the sub pages. p_m ranges from 1 to 10. It can be any value between 1 and 10, not just integer. If it is near 1,

then that means terms appearing in the main page and terms appearing in sub pages have little difference in terms of their importance in the content similarity calculation. Its optimal value is obtained by applying a genetic algorithm, which will be introduced in Section 3.6

Capitalization of Terms. Whether a term's first letter is capitalized when this term is not the first word of a sentence is also considered. If it is, then maybe it should be given a higher weight, since people usually use capitalization to emphasize concepts. A parameter, called p_c (can be any value from 1 to 10), is used to adjust the weight of this kind of term by adjusting its frequency. The adjusted frequency for this kind of term is $TF = p_c * f_c + f$, where f_c is this term's frequency in capitalization, and f is its frequency in regular form (not in capitalization).

Terms in Bold. In web pages, usually people like to use bold to emphasize the important concepts. A parameter p_b (ranging from 1 to 10) is used to adjust the weight of this kind of term by adjusting its frequency. The adjusted frequency is $TF = p_b * f_b + f$, where f_b is this term's frequency in bold, and f is its frequency in regular fonts.

Terms in a Larger Font. A higher weight is also given to the terms whose font size is relatively larger than its surrounding text, because usually people like to use larger font to emphasize concepts. The adjusted frequency is $TF = p_l * f_l + f$, where f_l is this term's frequency in a larger font, f is its frequency in regular size, and p_l is a parameter whose value ranges from 1 to 10.

Terms in the Title or Meta Description. The terms appearing in a page's title or meta-description will also be given a higher weight, because usually the terms appearing in these places are topical terms. The adjusted weight is $W_t = p_t * W$, where W_t is the

adjusted weight and W is the initial weight. p_t (ranging from 1 to 10) is a parameter used to adjust the weight. Terms appearing in title and meta-description are treated equally important, and they use the same parameter, p_t. Usually a term appears in the title or meta-description only once, so its weight is adjusted directly, instead of its frequency.

All the above factors together determine the weight of a term in a personal website. Some factors may not have noticeable effect on the performance. A genetic algorithm is used to find the importance of these factors. If a factor's effect is trivial, then the corresponding parameter will be equal or close to one, which means it makes no or little change to the original weight.

3.4.3 Link Similarity

As mentioned before, the content similarity method has some drawbacks, such as the word mismatch problem. The link-based methods are completely insensitive to word content, and therefore can complement the content-based similarity. How to calculate the link similarity is discussed in this section.

First, the layer concept in graph theory is introduced (see Figure 3.1). In a graph, the layer concept is used to define the relationship between the nodes. The nodes of interest are the nodes in layer 0. The nodes connected to a node in layer 0 are in layer 1. Nodes in layer 2 are those connected to a node in layer 1, but not connected to a node in layer 0, and so forth. The Web can be considered as a link graph. Let us use Figure 3.1 as an example. Suppose a person's website has page U, V and W, and all the pages of this personal website are in layer 0. Then page A, B, C and D will be in layer 1. Page M, N, O, X, and Y are in layer 2. In this study, only the links between layer 0 and layer 1 are

considered, which are link 1, 2, 3, and 4. Link 1 and link 2 are the inlinks of this personal website, and link 3 and link 4 are the outlinks of this website. The links between layer 1 and layer 2 are not considered for avoiding the topic drift problem (Henzinger, 2000): pages in layer 2 may have different topics from this personal website.

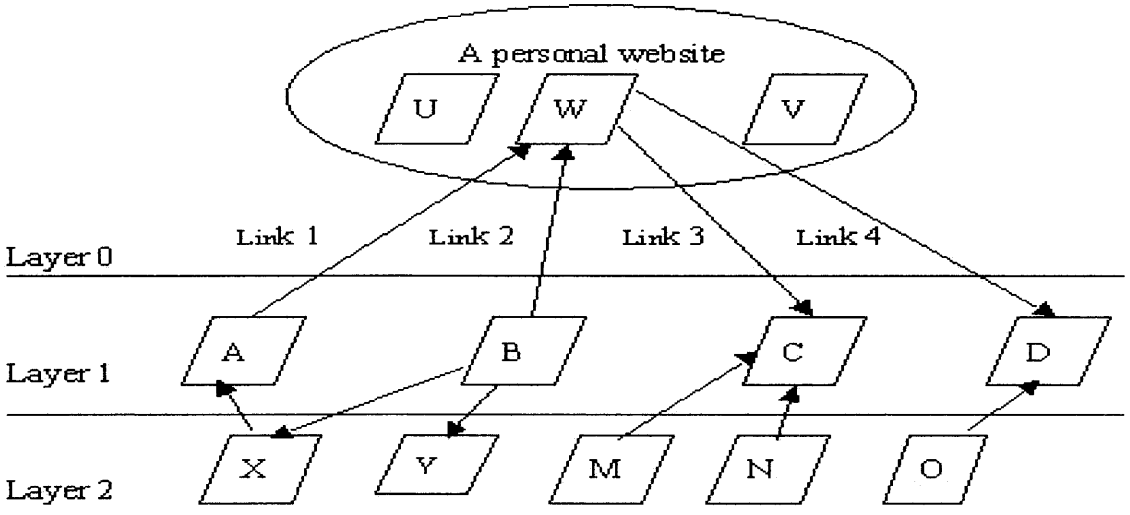

Figure 3.1 A link structure example.

3.4.3.1 Link Representation of a Website. Co-citation analysis and bibliographic coupling have been introduced in Section 2.4.2. They can also be used in the web environment as a link is considered as a display of interest on the target page. By analyzing links, an association can be established between two websites based on the existence of common children (common outlinks or forward links, meaning both websites have links to the same web page) or common parents (common inlinks or backlinks, meaning both websites are pointed to by the same web page). The more common inlinks or common outlinks shared by two websites, the more relevant these two websites are. When calculating content similarity, all pages of a website are considered in the link similarity calculation. For each page, outlinks and inlinks are extracted. Inlink and

outlink may have different degrees of importance in the link similarity calculation. Therefore, the inlink similarity and outlink similarity are first calculated separately, and then they are integrated together to obtain the final link similarity by linearly combining them. The link similarity between two personal websites is:

$$S_{link} = \alpha_{site} S_{inlink} + (1-\alpha_{site}) S_{outlink}$$

where S_{inlink} is the inlink similarity for the two websites, $S_{outlink}$ is outlink similarity, and α_{site} is a parameter to adjust the weights of S_{inlink} and $S_{outlink}$. α_{site} determines the degrees of importance of the two kinds of similarities in this integration, and its value ranges from 0 to 1.

Inspired by the representation method of the textual content of a document or website, which uses a bag of words, similarly, a bag of inlinks (or outlinks) is used to represent a website's inlink (or outlink) information. The inlink or outlink similarity calculation between two websites now becomes the similarity calculation between the two inlink (or outlink) bags representing the two websites.

This bag of links (inlinks or outlinks) contains all the inlinks (or outlinks). The bag of links is as follow:

$$B = \{(L_1, W_1), \ldots, (L_k, W_k)\}$$

where B is the bag of inlinks (or outlinks), $L_i (1 \leq i \leq k)$ is an inlink (or outlink) of this website, and W_i is the weight for link L_i. The inlink (or outlink) similarity calculation between two websites now becomes the similarity calculation between the two inlink (or outlink) bags.

In the link similarity calculation, the *navigational* links are not considered. These are links that solely serve the purpose of navigating within the same website, and they do

not convey an endorsement (by other people) for the content of the target page. Some examples of *navigational* links are: in the main page, the links pointing to the sub pages of this website, and in the sub pages, the links pointing to the main page of this website. For example, the link "Home" in a sub page usually points to the main page of this website. This "Home" link is for users to go back to the main page, and it is a *navigational* link.

3.4.3.2 Similarity Measure. In this study, the similarity measure used for link similarity calculation is also cosine similarity. Two other possible options are Jaccard coefficient measure and Euclidean distance measure. Previous studies have used Jaccard coefficient to measure the link similarity between two web pages (Menczer, 2004; Dean and Henzinger, 1999). Euclidean distance measure has been used in some precious studies to calculate content similarity between two documents, though it is less popular than the cosine similarity method. None of the previous studies has used Euclidean distance measure in link similarity calculation. To justify that the cosine measure is the most suitable measure, a test was conducted to compare these three measures in calculating link similarity to find if the cosine similarity is the best. The test method is described in Section 4.2.1 and the results are presented in Section 5.3.1. The formula of cosine similarity for inlinks is:

$$S_{inlink}(B_1, B_2) = \frac{\sum_{k=1}^{N}(w_{1k} \times w_{2k})}{\sqrt{\sum_{k=1}^{N}(w_{1k})^2} \times \sqrt{\sum_{k=1}^{N}(w_{2k})^2}}$$

S_{inlink} – the inlink similarity between two personal websites.

B_1, B_2 – bag of inlinks for the two websites

w_{1k}– the weight of inlink k in bag B_1

w_{2k}– the weight of inlink k in bag B_2

N – total number of the unique inlinks in a person's website.

k – from 1 to N.

The value of S_{inlink} rages from 0 to 1. The higher S_{inlink} is, the more similar the two websites are, in terms of their inlink similarity. The outlink similarity uses the same equation as inlink's. The only difference is that two bags of outlinks, instead of inlinks, are used in the calculation.

3.4.3.3 Link Weight. Previous studies (Menczer, 2004; Dean and Henzinger, 1999) use Jaccard coefficient to measure the commonality of links between two pages. They use only the presence and absence of the links, without any weight. Their method is likely to lose the information about the degree of importance of each link (e.g., a rare outlink may be more important than a very common outlink, such as a link to google.com). In this study, the link similarity method exploits the degree of importance of each link. Similar to how weights are assigned to terms in the content similarity calculation, weights are also assigned to links. A modified TF.IDF measure, called LF.IWF, is used to assign weight to links. LF refers to link frequency; it is the absolute frequency of a link (inlink or outlink) for a personal website. For example, if there is an outside page that points to five pages of a personal website, then the LF of this inlink for this website is 5. IWF is the link's inverse website frequency. The method to derive IWF is similar to the way of obtaining IDF. More details on calculating IDF are in Section 2.3.1. The weight of a link can be expressed by the following formula:

$$W = LF.IWF$$

where *W* is the weight of this link, *LF* is the absolute frequency of this link for this website, and *IWF* is the value of this link's inverse website frequency. This formula shows that the importance of a link for a website increases when the frequency of this link in the website increases, and decreases when the number of websites containing this link increases.

One other factor that may affect the importance of a link is also considered: the links for the main page, which is explained in the following paragraph.

3.4.3.4 Links of the Main Page. When calculating the weight of a link, the difference between a main page and the sub pages is also considered. For two different links having the same frequency, if one link is related to only the main page (meaning it is one of the main page's outlinks or inlinks) and the other one is related to only sub pages, it is very possible that the first link may be more important than the second one in link similarity calculation. To reflect this potential difference, a parameter called p_{ml} is used to adjust the weight of a link for the main page by adjusting its frequency. If a link is related to only the main page, then its adjusted frequency is $LF = p_{ml} * f_{mainl}$, where f_{mainl} is the frequency of the link related to the main page. If a link is related to both the main page and the sub pages (e.g., the main page and a sub page both have an identical outlink), then its adjusted frequency will be $LF = p_{ml} * f_{mainl} + f_{subl}$, where f_{subl} is its frequency related to the sub pages. p_{ml} ranges from 1 to 10. If it is 1, that means the links that are related to the main page and the links that are related to sub pages have no difference in terms of their importance in the link similarity calculation.

The algorithm presented in this section is the main algorithm under the proposed people search framework. It is hypothesized that it will outperform the other 13 algorithms introduced in the next section, Section 3.5.

3.5 Algorithms for Comparison

In this section, the other 13 algorithms are briefly described. The main focus is their differences from the main algorithm – the People-Search algorithm.

3.5.1 Site_Content_Inlink

The difference between this algorithm and the People-Search algorithm is that this algorithm integrates content information and inlink information of an entire website. This algorithm does not include any outlink information. In this algorithm, the similarity between two personal websites is:

$$S=\beta_{site\text{-}inlink}S_{content}+(1-\beta_{site\text{-}inlink})S_{inlink}$$

where $\beta_{site\text{-}inlink}$ is a parameter to adjust the weights of $S_{content}$ and S_{inlink}.

3.5.2 Site_Content_Outlink

The difference between this algorithm and the People-Search algorithm is that this algorithm combines content information and outlink information of an entire website. It does not consider any inlink information. The similarity between two personal websites is:

$$S=\beta_{site\text{-}outlink}S_{content}+(1-\beta_{site\text{-}outlink})S_{outlink}$$

where $\beta_{site\text{-}outlink}$ is a parameter to adjust the weights of $S_{content}$ and $S_{outlink}$.

3.5.3 Site_Content

This algorithm uses only the textual content of all the web pages of a personal website to calculate the similarity between two sites. In this algorithm, the link information is totally ignored, so the similarity between two personal websites is:

$$S = S_{content}$$

3.5.4 Site_Link

This algorithm uses only the link information of the web pages of a personal website. The method to calculate the link similarity between websites is exactly the same as that in the People-Search algorithm. The link similarity, S_{link}, is a linear combination of inlink similarity and outlink similarity. The similarity between two personal websites is:

$$S = S_{link} = \alpha_{site} S_{inlink} + (1-\alpha_{site}) S_{outlink}$$

where α_{site} is a parameter to adjust the weights of S_{inlink} and $S_{outlink}$.

3.5.5 Site_Inlink

This algorithm uses only the inlink information of a personal website. It ignores the content information and outlink information. The similarity between two personal websites is:

$$S = S_{inlink}$$

3.5.6 Site_Outlink

This algorithm uses only the outlink information of a personal website. It does not consider any content or inlink information in it similarity calculation. The similarity formula between two personal websites is:

$$S = S_{outlink}$$

3.5.7 MainPage_Content_Link

The difference between this algorithm and the People-Search algorithm is: this algorithm uses only the main page of a person's website to find the similarity between two sites, while the People-Search algorithm uses all the pages of a person's website, including the main page and all sub pages. Both of them integrate the content and link information, but this algorithm exploits the content and link information from only the main page, not the sub pages. The formula of the similarity between two personal websites is:

$$S = \beta_{page\text{-}link} S_{content} + (1-\beta_{page\text{-}link})\, S_{link}$$

where $S_{content}$ is the content similarity between the main pages of two personal websites, and S_{link} is the link similarity between the two main pages. $\beta_{page\text{-}link}$ is a parameter to adjust the weights of $S_{content}$ and S_{link}.

The formula to calculate link similarity between the two main pages is:

$$S_{link} = \alpha_{page} S_{inlink} + (1-\alpha_{page})\, S_{outlink}$$

where S_{inlink} is inlink similarity between two main pages, $S_{outlink}$ is outlink similarity between two main pages, and α_{page} is a parameter to adjust their weights. The differences between this algorithm and the People-Search algorithm are emphasized below.

The calculation of $S_{content}$: All the procedures are the same as that of the People-Search algorithm, except the followings:

1. In this algorithm, only the content-bearing words from the main page are used, while in the main algorithm, the content-bearing words from all pages are used.
2. This algorithm uses only the main page, so the parameter p_m does not exist in this algorithm. In the main algorithm, this parameter is used to adjust the weight of a term appearing in the main page by adjusting its frequency.

The calculation of S_{inlink} and $S_{outlink}$: All the procedures are the same as that in the main algorithm, except the followings:

1. In this algorithm, only the links (inlinks and outlinks) of the main page are used, while in the main algorithm the links of all the pages in a website are used.
2. This algorithm uses only the main page, so the parameter p_{ml} does not exist in this algorithm. In the main algorithm, this parameter is used to adjust the weight of a link related to the main page.

3.5.8 MainPage_Content_Inlink

This algorithm combines content information and inlink information of the main page of a personal website in calculating similarity between two websites; it ignores the outlink information. The formula is:

$$S=\beta_{page\text{-}inlink}S_{content} + (1-\beta_{page\text{-}inlink})\, S_{inlink}$$

where $\beta_{page\text{-}inlink}$ is a parameter to adjust the weights of $S_{content}$ and S_{inlink}

3.5.9 MainPage_Content_Outlink

This algorithm combines content information and outlink information of the main page; it does not consider any inlink information. The similarity formula is:

$$S=\beta_{page\text{-}outlink}S_{content} + (1-\beta_{page\text{-}outlink}) S_{outlink}$$

where $\beta_{page\text{-}outlink}$ is a parameter to adjust the weights of $S_{content}$ and $S_{outlink}$

3.5.10 MainPage_Content

This algorithm uses only the textual content of the main page of a personal website. Both inlink and outlink information are ignored. The similarity formula is:

$$S = S_{content}$$

3.5.11 MainPage_Link

This algorithm uses only the link information of the main page. The method to calculate the link similarity between two websites is exactly the same as that in algorithm MainPage_Content_Link. The link similarity, S_{link}, is a linear combination of inlink similarity and outlink similarity. In this algorithm, the similarity between two personal websites is:

$$S = S_{link} = \alpha_{page}S_{inlink} + (1-\alpha_{page}) S_{outlink}$$

where α_{page} is a parameter to adjust the weights of S_{inlink} and $S_{outlink}$.

3.5.12 MainPage_Inlink

This algorithm uses only the inlink information of the main page. Both content information and outlink information are ignored in this algorithm. The similarity formula is:

$$S = S_{inlink}$$

3.5.13 MainPage_Outlink

This algorithm uses only the outlink information of the main page. It does not consider content or inlink information. The formula of the similarity between two sites is:

$$S = S_{outlink}$$

3.6 Turning the Algorithm Parameters

In Sections 3.4 and 3.5, several parameters, which are used to integrate and calculate the content similarity and link similarity for the 14 algorithms, have been introduced. To determine the best values for those parameters, a genetic algorithm is used to optimize them. The genetic algorithm was inspired by biological evolution (Holland, 1975; Whitely, 1989; Goldberg, 1989). It works with a set of bit strings, called population of individuals. The initial population is usually randomly generated within the value ranges of the parameters. New individuals (parameter value sets) are generated in parallel by mutation and crossover. Each individual is assigned a score based on pre-defined measures of quality. The best few of these solutions are chosen and replicated, and the poorer solutions are discarded. After the replication, new breeding population is created. From the created breeding population, new individuals are generated. The breeding

operation is fulfilled by an exchange of some of the characteristics of the chosen individuals in a crossover operation. It is analogous to the biological interchange of genes between two chromosomes. Among the new individuals, some of them may be better than their parents, while some others may be worse. A new iteration will start for all the existing individuals and the new ones. During iterations, a mutation process may happen. It will randomly choose some individuals and exchange some of their characteristics. As iterations for the genetic algorithm progress, gradually, the fitter individuals will have more children than less fit individuals, and the new individuals tend to be increasingly fit, until reaching the optimized state and the optimal values for the parameters are found.

Genetic algorithms have been used in many applications, such as relevance feedback for IR systems (Yang & Korfhage, 1992) and tuning parameters for keyphrase extraction program (Turney, 2000). A genetic algorithm is a supervised machine learning method, which means a set of training data and a performance measure are needed to help the learning system adapt. Based on the training data and the given measures, the system can obtain a performance value for each set of parameter values and finally obtain the set of optimal parameter values. In this study, the personal websites obtained from ODP are used as the training data, and a statistic measure, Kruskal-Goodman Γ measure (Goodman & Kruskal, 1954; Haveliwala et al. 2002), is used as the performance measure. Using Kruskal-Goodman Γ value as the performance measure, the genetic algorithm was applied on the ODP training dataset to find the optimal parameters for the explored algorithms (More details about Γ measure are discussed in Chapter 4). After the genetic learning process, a set of optimized parameters for each algorithm were obtained. The results are reported in Chapter 5.

3.7 People Search System Architecture

In this section, the high-level system architecture of the people search system, the main system components, the evaluation systems and the system database structure are described.

3.7.1 System Architecture and the Main Components

Figure 3.2 shows the high level architecture of the people search system and associated prototype systems and programs used for the evaluation. The people search system includes the following components: a personal website crawler, a web page processing and indexing module, a website similarity calculation module, a people search query processing module, an algorithm parameter training program and two types of data repositories – the system database and data files. Figure 3.2 also contains a dash-line box, which represents the people search prototype systems and programs used to evaluate the algorithms explored in this study. These prototype systems and programs are independent of the people search system, and are not further used once the evaluation task was done. The algorithm parameter training program is used to find the optimal values for the algorithm parameters by applying a genetic algorithm. System data are stored in a system database and also in the system data files. The system database and data files are shared by all the components and evaluation systems. The regular arrows in Figure 3.2 represent the data flows between different components. Different modules of the people search system do not directly interact with each other; they share and transfer data through the system database and data files. The dash-line arrows in Figure 3.2 do not represent any data flow or control signal; they only show the sequence of executions of

the four main components. All the components appearing in Figure 3.2 are discussed in more details below.

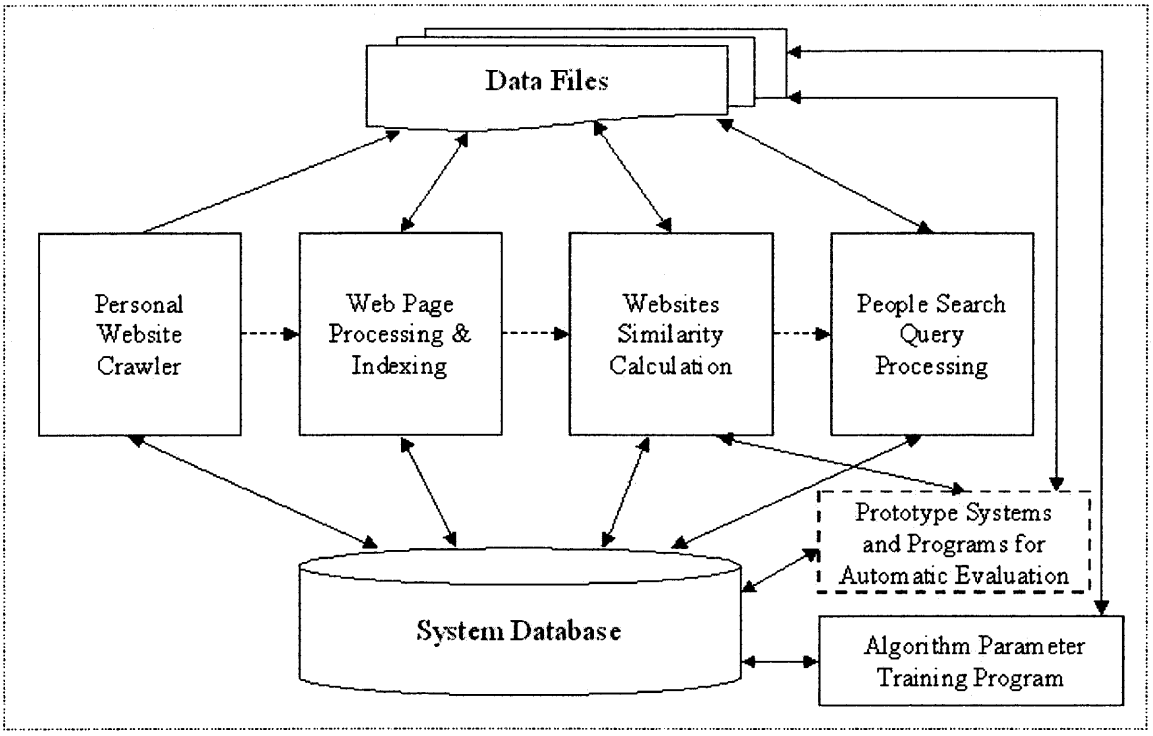

Figure 3.2 People search system architecture.

3.7.1.1 Personal Website Crawler. A special crawler was developed for the task of crawling personal websites. Given the URL of a personal website, this crawler can automatically download all web pages belonging to this site. It first downloads the main page of this site. Then it extracts the root directory of this website, and based on the root directory, all subdirectories and pages under the root directory are crawled. The crawler gives a unique page id to each crawled page and saves this page locally in its original format. The corresponding metadata, such as page URL and the last updated date, are also stored in the system database. Only the pages with textual information, such as html files and plain text files, are saved. Image, .PDF, Word and other kinds of files with special format are crawled but not saved locally; however, their metadata, such as URL,

are stored in the system database. The crawler can crawl 10 personal websites in parallel at the same time.

3.7.1.2 Web Page Processing and Indexing. After a personal website is crawled, all its pages are processed by this component. First, outlinks in each page are extracted and stored in the system database. Second, for each web page, except the tags used to identify the unusual terms, such as terms in bold, all other HTML tags, JavaScript code, non-alphabetic characters, and other non-content related symbols, are removed. Then all terms are extracted from the page. After all terms are extracted, a stop words removal process is applied. All the terms appearing on the stop words list will be removed. Then each content-bearing word is processed by a word stemming program, developed specifically for this study. The related information for some special words, such as words in meta-description or bold, are also recorded. Third, each page's inlinks are obtained from Yahoo and Google, and then are combined together.

After all the words, inlinks and outlinks are processed, their IWF values over all the collected websites are calculated (see Sections 3.4.2 and 3.4.3 for details about IWF). Finally, all terms, inlinks and outlinks are indexed for similarity calculation and user search.

3.7.1.3 Website Similarity Calculation. This component is to calculate various kinds of similarities between two personal websites, such as inlink similarity and content similarity. The integration of different kinds of similarities is also done by this component. To calculate similarity between two websites, this component will access inlink, outlink and content information stored in the system database and data files. All the similarity values are stored in the system database after they are calculated.

3.7.1.4 People Search Query Processing. This is the retrieval part of the people search system. It includes a people search user interface and associated functions, which retrieve the relevant websites from the system database, rank them and present them to users.

3.7.1.5 Prototype Systems and Programs for Evaluation. To automatically evaluate the proposed People-Search algorithm and the other 13 algorithms, fourteen prototype systems based on these algorithms were developed. Several programs used to calculate the performance values for different measures, such as Kruskal-Goodman Γ, were also developed. The prototype systems automatically execute a set of queries and send the search results to the evaluation programs mentioned above. These programs then calculate the values for each performance measure and store them in the system database. The main purposes of these programs are:

- Calculate precision, recall, and F values for the seven algorithms using information from an entire website
- Calculate precision, recall, and F values for the seven algorithms using information from only the main page of a personal website
- Calculate Kruskal-Goodman Γ values for all these algorithms
- Compare the three kinds of link similarity measures (cosine, Euclidean distance and Jaccard coefficient).

3.7.1.6 Algorithm Parameter Training Program. This is the genetic algorithm used for training the algorithms to obtain the optimal values of the algorithm parameters. This program accepts a set of queries and finds a set of optimal values for the algorithm parameters, based on these queries and the performance measure, i.e., Kruskal-Goodman Γ measure.

3.7.1.7 System Database. The system database is shared by all system modules and programs. It stores the metadata of each crawled website, the similarity values between websites and the data about inlinks, outlinks and terms. A database diagram showing the main tables of the system database is shown in Figure 3.3. Details about the system database are discussed in Section 3.7.2

3.7.1.8 System Data Files. The data files are used to store system data that are not suitable for storing in the database due to some limitations, such as access speed. The main data files include the followings: the WordNet lexicon data files, the original web page files, plain text files generated from crawled web pages, term files for each website as well as for each individual page, and inverted index files. For each term listed in a term file, there is a list of items attached to it. Examples of such items are: this term's frequency in the main page, its frequency in sub pages, and its original form before stemming.

3.7.2 System Database Structure

The system database contains website metadata, similarity values between websites, and data about links and terms. Figure 3.3 shows a simplified system database diagram. Because of the large number of tables in the database, only the important tables are included in this diagram.

The T_website table contains metadata about each website, such as the internal web id, website title, description, root URL, etc. It is the center of all other tables. The T_ODP table contains the ODP catogery information for each website that was crawled from ODP directory. Table T_site_inlink and T_site_outlink store inlinks and outlinks for

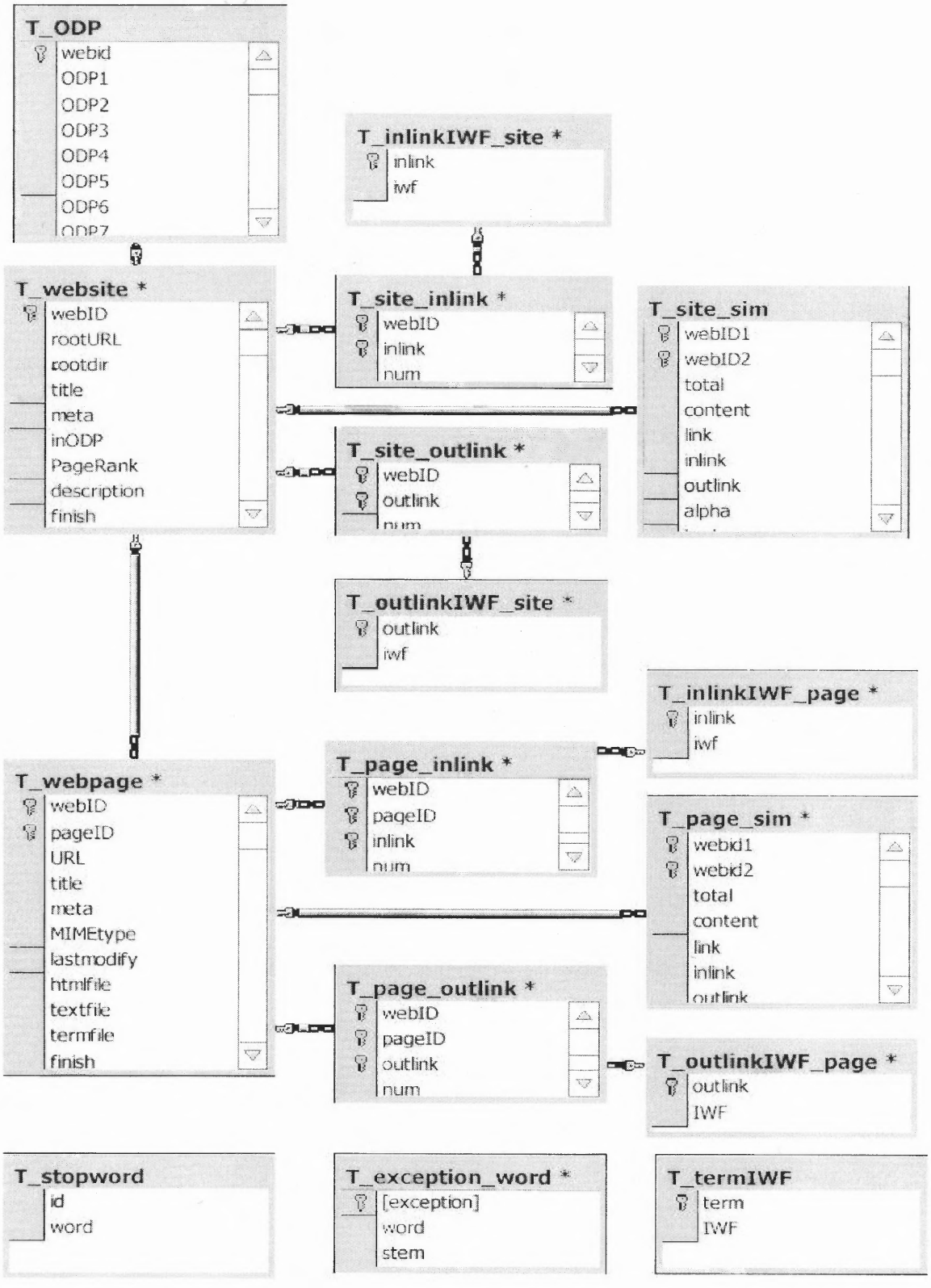

Figure 3.3 System database diagram.

each website and are linked to table T_website. They are also liked to table T_inlinkIWF_site and T_outlinkIWF_site, which contain the link IWF information. Table T_site_sim contains different kinds of similarity values between two websites. When the system receives a user query, it will look up the T_website table to find the query's internal web id, and then from the similarity table it will search the relevant websites based on their similarity values. Each website may have one or more pages. The T_webpage table stores information about these pages. Several tables are connected to this table, such as T_page_inlink. T_stopword, T_exception_word and T_termIWF are used for term processing and content similarity calculation.

3.8 Summary

This chapter has presented the research methodology of this study. The people search framework was first introduced, and then the 14 algorithms based on this framework were described, with emphasis on the People-Search algorithm, which integrates both the content and the link information of all the web pages of a personal website. The genetic algorithm, which is used to tune the algorithm parameters, was also introduced. Finally, the people search system architecture and its main components were described. In the next chapter, the evaluation method for the proposed people search solution will be described.

CHAPTER 4

EVALUATION METHODOLOGY

Chapter 3 has described the goals of this research and presented four research questions which are investigated in this study. The first two questions are how to represent a person on the Web for people search and what kinds of methods/algorithms can be designed based on the proposed person representation method. Section 3.3 has presented the people search framework and the 14 algorithms based on the proposed framework. The proposed framework and the designed algorithms can answer the research question 1 and 2. This chapter describes the evaluation methods, which try to answer research question 3 and 4:

Research question 3:

Among the possible methods/algorithms, which one performs the best?

Research question 4:

How effective is the best algorithm on ranking the returned search results?

In this study, there are two kinds of evaluations: an automatic evaluation without subject involvement and a human evaluation to collect the subjective ratings of the prototype system. In the automatic evaluation, the 14 algorithms were compared to each other to test the hypothesis, which is that the People-Search algorithm outperforms the other 13 algorithms. User studies are usually time-consuming and costly, but they reflect how the real users feel about a system. Therefore, human subjects were also recruited to evaluate the algorithm which had the best performance in the automatic evaluation and

two other important ones. Several prototype systems were developed for the automatic evaluation and human evaluation.

The experimental dataset used in this study is introduced in Section 4.1. In Section 4.2, the automatic evaluation method is described. The human evaluation method is presented in Section 4.3

4.1 Experimental Dataset

To automatically evaluate these algorithms, there should be a dataset of personal websites which are already labeled. For each personal website in this dataset, the information about what other personal websites in this dataset are relevant should be available. Open Directory Project (ODP), also called DMOZ, *http://www.dmoz.org*, is the largest, most widely distributed, and most comprehensive human-edited directory of the Web. Millions of web pages are classified into this hierarchical structure. It is constructed and maintained by a vast, global community of volunteer editors. ODP powers the core directory services for the Web's largest and most popular search engines and portals, including Netscape Search, AOL Search, Google, Lycos, HotBot, and many others. Many previous studies have used ODP directory for document classification and other kinds of tasks related to web pages (Plu et al. 2003; Menczer, 2004; Fogaras and Racz, 2004; Haveliwala, 2002). In addition to the general web pages, ODP directory also contains personal websites. The ODP personal website directory is also hierarchically arranged by subject - from broad to specific. Its personal website directory is maintained by community editors who evaluate sites for inclusion in the directory. The editors are experts, and all submissions are subject to editor evaluation. The personal websites listed

in the ODP directory were used as the dataset in the evaluation. In ODP's personal website directory, similar personal websites are placed into the same sub-category. Figure 4.1 shows an example of the ODP personal websites structure. This figure shows that 43 personal websites are placed into the *Science:Math:Algebra* sub-category.

For the evaluation, 20,000 personal websites were crawled. First the home addresses of these personal websites were obtained from ODP. Then these addresses were sent to the web crawler that was implemented specifically for this study. For each personal home address, the crawler crawled all the web pages belonging to this personal website, including the main page and all sub pages. These web pages were then processed to obtain the content-bearing words, inlinks and outlinks. Overall, these 20,000 websites have 740,230 pages, and on average a person's website contains about 37 web pages. Table 4.1 shows the distribution of the crawled personal websites among the 11 ODP top categories. More detailed analysis of the dataset will be presented in Chapter 5.

Table 4.1 Distribution of Crawled Personal Websites

Top Category	Number of Personal Websites
Arts	3,255
Business	113
Computers	3,200
Games	246
Health	578
Home	1,495
Kids_Teens	1,086
Recreation	4,257
Science	878
Society	3,653
Sports	1,239

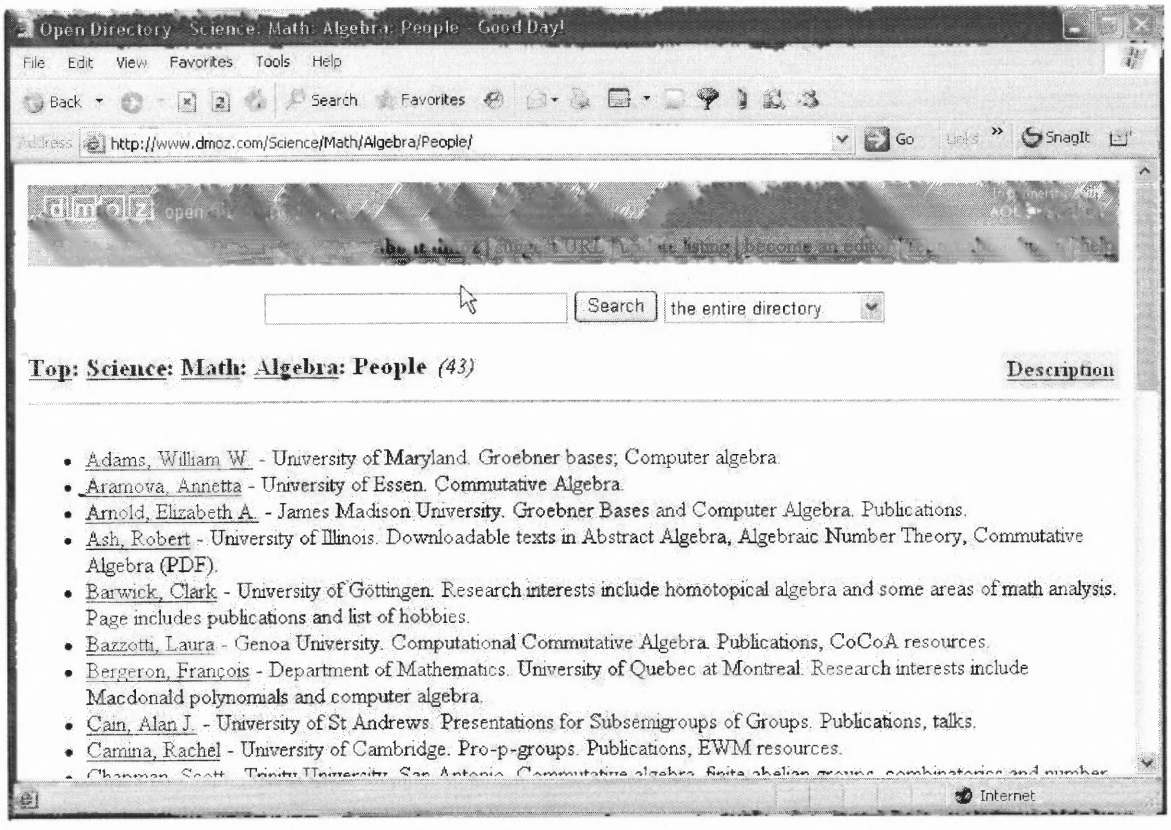

Figure 4.1 Personal websites in the ODP directory.

4.2 The Automatic Evaluation

This section describes the methodology of the automatic evaluation. In Section 4.2.1, how to compare the three kinds of link similarity measures, which have been briefly discussed in Section 3.4.3, is described. Section 4.2.2 presents how the traditional information retrieval measures (precision, recall and F measure) are used to evaluate these 14 algorithms. Section 4.2.3 describes the drawbacks of using the traditional IR measures in this study and presents one another measure, Kruskal-Goodman Γ, which is more suitable for comparing these algorithms with this dataset.

4.2.1 Comparing Three Link Similarity Measures

As described in Chapter 3, in this study, cosine similarity is used in both content similarity calculation and link similarity calculation. Many previous studies have proved that cosine similarity measure is better than other similarity measures in calculating content similarity, and many researchers have used it in their studies. Therefore, this measure is used in this study to calculate content similarity. However, none of the previous studies has used this measure in the link similarity calculation. To ensure that the right measure has been chosen for the link similarity, cosine measure is compared to two other measures, Jaccard coefficient and Euclidean distance, to see which would perform the best in calculating link similarity. The reason these two are chosen for comparison is: Jaccard coefficient similarity measure has been used by previous studies in measuring link similarity (Menczer, 2004; Dean and Henzinger, 1999), and Euclidean distance measure is also a popular measure in calculating content similarity and has been used by precious studies in content similarity calculation, though it is not as popular as cosine similarity measure.

These three measures were tested in both inlink similarity calculation and outlink similarity calculation. For cosine similarity and Euclidenn similarity method, each link (inlink or outlink) has a weight. The calculation of a link's weight is discussed in Section 3.4.3. The formulas for calculating cosine similarity and distance similarity have been introduced in Chapter 2 and Chapter 3. Jaccard coefficient only considers the presence and absence of a link; therefore, a link has no weight in Jaccard coefficient measure. Jaccard coefficient is defined as the size of the intersection divided by the size of the union of the two sets of links from the two personal websites under consideration:

$$S(p,q) = |U_p \cap U_q| / |U_p \cup U_q|$$

where p and q are two websites and U_p is the set containing p's inlinks or outlinks. U_q has a similar meaning as U_p.

The dataset used for this test has been described in Section 4.1. Kruskal-Goodman Γ was used as the performance measure, which is described in Section 4.2.3. Five hundred personal websites were randomly selected from this dataset as the queries. Below is the procedure of testing these three measures in inlink similarity calculation. The test procedure for outlink similarity calculation is similar. The procedure for comparing the three similarity measures in inlink similarity calculation is:

1. For each of the 500 queries, based on cosine similarity, the inlink similarities between this query and all other websites in the dataset were calculated.
2. For each query, all other websites were ranked in descending order according to their similarity values computed in the previous step.
3. Γ value was calculated for each query.
4. An average Γ value was calculated for all the 500 queries.
5. Repeat step 1 to 4 using Euclidean distance and Jaccard coefficient.
6. The three similarity measures were compared to each other based on their average Γ values. The one having the highest Γ value is the best for calculating inlink similarity. Paired t-test was used to test the significance of the result.

4.2.2 Evaluating the 14 Algorithms Using Precision, Recall and F Measure

This section explains how to use the traditional IR effectiveness measures to evaluate the 14 algorithms introduced in Chapter 3. Section 4.2.2.1 presents the definitions of precision, recall and F in the context of this study. Section 4.2.2.2 describes how to use domain-independent queries to evaluate the 14 algorithms. Based on the evaluation

results using domain-independent queries, the top five algorithms were evaluated again in three individual domains (categories) to see if they would perform differently across domains. This test is described in Section 4.2.2.3. Section 4.2.2.4 describes how to evaluate the People-Search algorithm's effectiveness on ranking returned results.

4.2.2.1 Definitions of Precision, Recall and F. Measuring precision, recall and F measure is easy to carry out, and allows more precise comparison between different systems or algorithms. In this automatic evaluation, these three traditional IR measures were used to compare the performance of the 14 algorithms.

As mentioned before, in this study, a query is a person (this person's website) and the returned search results are a list of similar people (a list of personal websites). In the context of this study, the definitions for precision, recall and F measure are defined as follows. Precision is the proportion of returned personal websites that are relevant to the query website.

$$\text{Precision} = \frac{R}{N}$$

R – number of returned personal websites that are relevant to the query.

N – total number of returned personal websites.

Recall is the proportion of relevant personal websites that are returned.

$$\text{Recall} = \frac{R}{M}$$

R – number of returned personal websites that are relevant to the query

M – total number of relevant personal websites in the dataset.

There is usually a trade-off between precision and recall, and either of them alone does not paint a complete picture of system effectiveness. Therefore, the F measure was

invented to show the combined results (Rijsbergen, 1979). Below is the most common formula for F, which was used in this evaluation:

$$F = \frac{2 * \Pr ecision * \mathrm{Re} call}{\Pr ecision + \mathrm{Re} call}$$

4.2.2.2 Evaluating the 14 Algorithms Using Queries from All ODP Categories. In the ODP directory, semantically similar personal websites are placed in the same sub-category. In the automatic evaluation using precision, recall and F measure, the websites in the same leaf sub-category are considered relevant to each other, and the websites outside this sub-category are considered not relevant to the websites in this sub-category. Let us use Figure 4.1 as an example. In this case, all these 43 personal websites are relevant to each other, and all other websites not belonging to this sub-category are not relevant to the websites in this sub-category.

Fourteen prototype systems were developed for the 14 algorithms, each of which was implemented based on one of the 14 algorithms. Two thousand personal websites were randomly selected from the dataset as queries. The procedure for comparing the 14 algorithms using precision, recall and F measure is:

1. The 2,000 queries were sent to each of the 14 prototype systems. Each system executed the same 2,000 queries and returned a list of personal websites for each query.

2. For each system (algorithm), precision, recall and F were calculated for each query at five measure points: when the number of returned websites was 10, 20, 30, 40 and 50. Then for each measure, the average of the 2,000 queries was calculated at each of the five cases.

3. For each measure, the 14 algorithms were compared to each other at these five measure points. Significance tests were conducted using pair-t test to see if the performance differences between these systems were statistically significant. The results would show whether the People-Search algorithm outperformed all other algorithms.

4.2.2.3 Evaluating the Top Five Algorithms Using Queries from Three Individual Domains. The results of the experiment described in the last section would tell us the performance of the algorithms across all ODP categories (domains). To see if these algorithms would perform differently in different domains, the top five algorithms were also tested in three individual domains. Based on the cross-domain test results (described in Chapter 5), the top five algorithms are Site_Content_Link (the People-Search algorithm), Site_Content_Inlink, Site_Content_Outlink, Site_Content, and MainPage_Content_Link. The three chosen domains are Arts, Sports and Computers. The test results in these three domains will show if the People-Search algorithm was still the best algorithm in a specific domain. The performance measures used were still precision, recall and F. For each domain, the procedure for comparing these five algorithms is:

1. Five hundred websites were randomly chosen from this domain as queries. The 500 queries were sent to each of the five prototype systems. Each system executed the same 500 queries and returned a list of personal websites for each query.

2. For each algorithm, precision, recall and F were calculated for each query at five measure points: when the number of returned websites was 10, 20, 30, 40 and 50. Then for each measure, the average of the 500 queries was calculated at each of the five measure points.

3. For each measure, the five algorithms were compared to each other at the five measure points. Significance tests were conducted using pair-t test to see if the performance differences between these algorithms were significant.

4.2.2.4 Evaluating People-Search Algorithm's Effectiveness on Ranking Returned Results. This test is to answer research question 4 – how effective the best algorithm is on ranking the search results. Based on the experimental results, the algorithm having the best performance was the People-Search algorithm (see Chapter 5). Therefore, this test was to test the People-Search algorithm's effectiveness on ranking the

search results. A good IR algorithm should rank the returned results in descending order of their relevance. In other words, the most relevant hits should be presented first, and the least relevant ones should be at the bottom of the returned list. To evaluate the People-Search algorithm's ranking effectiveness, for each query, the top 50 returned hits were divided into five groups: top 1 to top 10 as group 1, top 11 to 20 as group 2, top 21 to 30 as group 3, top 31 to 40 as group 4, and top 41 to 50 as group 5. Precision was first calculated for each group, and then they were compared to each other. If group 1's precision was higher than group 2's, and group2's was higher than group3's, and so on, then that means this algorithms was effective in ranking the search results. The procedure is as follows:

1. The same 2,000 queries used in Section 4.2.2.2 were used as the experimental queries. These 2,000 queries were executed by the prototype system implemented based on the People-Search algorithm.

2. First, for each query, the precision for each group was calculated. Then, for each group, the average precision over all the 2,000 queries was calculated.

3. The five groups were compared to each other based on their average precisions. Pair t-test was used to test if the precision difference between two groups was statistically significant.

4.2.3 Evaluating the 14 Algorithms Using Kruskal-Goodman Γ Measure

Last section describes how to use the traditional IR measures to evaluate the 14 algorithms. These measures require that the query and returned hits to have a clear relationship, either relevant or not. In the last section, the websites in the same leaf ODP sub-category are considered relevant to each other; all other websites outside the sub-category are not relevant to the websites in the sub-category mentioned above. However,

in the ODP directory, the personal websites are organized into a hierarchical structure. It may not be appropriate to categorize the relationship of two websites into just one of the two types, relevant or irrelevant. Most of the time, it is in between. Actually, there is a great deal of implicit ordering information in ODP hierarchical Web directory (Haveliwala et al. 2002). For example, a personal website in the *computer/Internet/search engine* sub-category is more similar to other websites in the same sub-category than those outside of this sub-category. Furthermore, that website is likely to be more similar to other websites in other *computer/Internet* sub categories than those entirely outside of the computer category, such as *sports/soccer*. The most similar websites to a given website (or source website) are the other websites in the same sub-category, followed by those in sibling sub-categories, and so on.

Based on the rationale described above, it is more appropriate to use other measures, which fit and exploit the hierarchical nature of the ODP directory, to evaluate the fourteen algorithms. In this section, the Kruskal-Goodman Γ measure (Goodman & Kruskal, 1954; Haveliwala et al. 2002) is introduced to compare the 14 algorithms. This method uses the structure of the ODP directory as the ground truth, and compares it with the returned results of a prototype system to evaluate the system's effectiveness. Kruskal-Goodman Γ has been used in previous studies where ODP directory is used as the experimental dataset (Haveliwala et al. 2002; Fogaras & Racz, 2005). Before introducing Kruskal-Goodman Γ, the nature of the ODP hierarchical structure is described in detail.

4.2.3.1 ODP Hierarchical Structure and the Kruskal-Goodman Γ Measure.

To formalize the notion of distance from one personal website to another in the hierarchy, the *familial distance* $D_{distance}(x,y)$ from a website x to another website y in a class hierarchy is defined as the distance from x's class to the most specific class (category) dominating both x and y (Haveliwala et al. 2002). The website x is on average more similar to a same-class website than to a sibling-class website, and is on average more similar to a sibling-class website than a cousin-class website, and so on (see Figure 4.2). This induces a partial ordering of the websites which is referred to as the familial ordering with respect to x. It shows the relationship between partial ordering and the directory hierarchy.

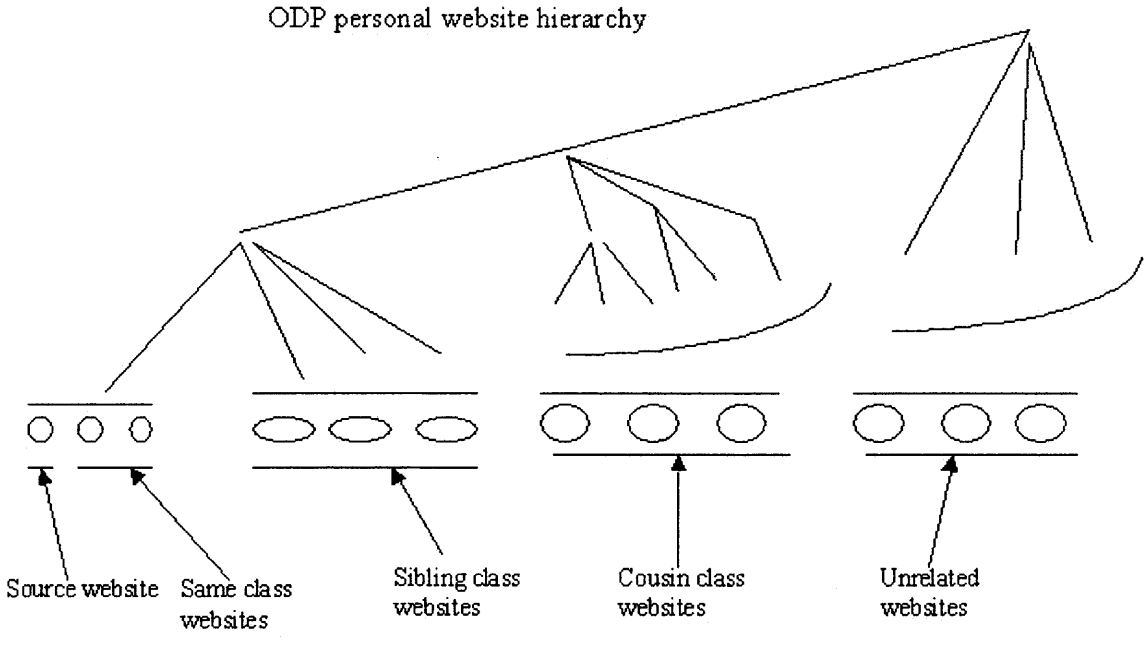

Figure 4.2 Given a website, mapping an ODP hierarchy onto a partial ordering.

Given a query website, its familial distances to other websites can be used to construct a partial similarity ordering over those websites. The general principle is: *on average, the true similarity of websites to a query website decreases monotonically with the familial distance from that website.* Given this principle and the definition of familial distance, for any query personal website in the hierarchical directory, a partial ordering of all other websites in the directory can be derived. Then this partial ordering can be used to evaluate the correctness of the ordering (the rank of the returned websites) produced by one of the prototype systems. The two orderings (the partial ordering induced from ODP hierarchy and the website ordering produced by one of the prototype systems) can then be compared. The Γ value is defined as following (Goodman & Kruskal, 1954). For orderings A and B:

$$\Gamma(A, B) = 2 \times Pr[A, B \text{ agree on } (x, y) \mid A, B \text{ order } (x, y)] - 1$$

A and B agree on the pair (x, y) means the order between x and y is the same in both A and B. For example, if x has a higher rank than y in A and B, then A and B agree on the pair of (x, y). However, if x's rank is higher than y's in A, but lower than y's in B, then A and B do not agree on the pair of (x, y).

Γ value ranges from –1 to +1, where 0 is the expected value of a random ordering. If Γ value is 1, that means A and B agree on all the website pairs. On the other hand, if Γ value is –1, then that means A and B disagree on the entire website pairs. If $\Gamma(A, B) = 0.7$, it means the two orderings agree on 85% of the pairs (percentage of agreements $Pr = (\Gamma + 1)/2$).

When comparing two orderings, considering all the pairs will give a more complete view on the difference between these two orderings. Therefore, when using Γ

measure to compare different algorithms, all the returned results were included, not just the top N returned hits. An average Γ value was obtained for each of these prototype systems. The higher a system's Γ value is, the better the system is, since having a higher Γ value means the ordering of the returned websites produced by this system is closer to the ground truth ordering of the ODP hierarchy.

4.2.3.2 Compare the 14 Algorithms Using Γ Measure and Queries from All ODP Categories. This test is to compare the 14 algorithms across all domains (categories). The same 2,000 queries used for the traditional IR measures were used as the queries of this test. They were randomly selected from the whole experimental dataset. The procedure for this test is:

1. The 2,000 queries were sent to each of the 14 prototype systems. Each system executed the same 2,000 queries and returned a list of personal websites for each query.

2. For each system (algorithm), Γ value was calculated for each query. Then the average Γ value for all the 2,000 queries was calculated for each algorithm.

3. Finally, these 14 algorithms were compared to each other based on their Γ values. Significance tests were conducted using pair-t test to see if differences between the Γ values were statistically significant. The algorithm with the highest Γ value would be the best, since it was the closest to the ground truth, i.e., the ordering induced from ODP directory.

4.2.3.2 Comparing the Top Five Algorithms Using Γ Measure and Queries from Three Individual Categories. The test in last section would tell us the performance of the algorithms across all categories (domains). Next was to investigate whether these algorithms would perform differently in different domains and if the People-Search algorithm was still the best in each individual domain. Therefore, the top five algorithms, obtained based on their cross-domain performance, were also tested in

three individual domains using Γ measure. Based on the cross-domain test results obtained by using Γ measure (described in Chapter 5), the top five algorithms were Site_Content_Link (the People-Search algorithm), Site_Content_Inlink, Site_Content_Outlink, Site_Content, and MainPage_Content_Link. The same three domains (Arts, Sports and Computers) were used. For each domain, the procedure for comparing these five algorithms was:

1. Five hundred websites were randomly chosen from this domain as queries. These queries were the same as the ones used in evaluating the top five algorithms using the traditional IR measures, described in Section 4.2.2.3. These queries were sent to each of the five prototype systems.

2. For each algorithm, the Γ value was calculated for each query. Then the average Γ value for all the 500 queries was calculated for each algorithm.

3. Finally these five algorithms were compared to each other based on their Γ values. Pair-t test was used to test the significance of the results.

4.3 The Human Evaluation

User studies reflect how the real users feel about a system. Therefore, human subjects were recruited to evaluate the important algorithms. Considering the amount of efforts required for human evaluation, and the fact that the 14 algorithms had been evaluated in the automatic evaluation, in the human evaluation only three algorithms were evaluated. These three algorithms were: the People-Search algorithm, Site_Content and MainPage_Content_Link. The reasons only these three algorithms were chosen for the human evaluation were: first, they had better performance in the automatic evaluation; and second, they represent three directions of exploiting information of a personal website to represent a person, which are quite different. The People-Search algorithm

uses both content and link information of a website; Site_Content algorithm uses only content information from an entire site; and MainPage_Content_Link considers link and content information but only from the main page. Although Site_Content_Inlink and Site_Content_outlink also had good performance in the automatic evaluation, they are similar to the People-Search algorithm, in terms of the information they use – both content and link (inlink or outlink) information. The Site_Link algorithm uses only link information, but its performance in the automatic evaluation was far behind the algorithms mentioned above, so it was not evaluated in the human evaluation.

An online prototype system was developed for the human evaluation. Subjects did the entire experiment online. The main function of the prototype system is the people search function, giving a personal website as the query the system will return personal websites relevant to the query site. For the sake of the human evaluation, this system implemented all the three algorithms to be evaluated. For each algorithm, the top 20 returned results were evaluated by subjects. For each query, the system combined the search hits of all these three algorithms and returned them to users together. Therefore, theoretically there would be 60 returned hits for each query. However, since there were overlaps among the returned hits of the three algorithms, the actual number of returned hits for each query was much less than 60, which was 30.6 (see Section 5.4 for details). To make a fair comparison, their returned hits were mixed together, without any special order.

In the following subsections, the experimental dataset, queries and subjects are discussed first; then the experimental procedure is described; and finally what experimental results and data analyses to be presented in Chapter 5 are listed.

4.3.1 Dataset, Queries and Subjects

The dataset was the same as the one used in the automatic evaluation, which was described in Section 4.1. One reason of using the same dataset was to find if there was any correlation between the results of the automatic evaluation and the human evaluation. Fourteen queries were chosen from the dataset and presented to subjects. Subjects were required to choose four of them, with which they would feel comfortable, for the evaluation. If they preferred to use their own queries, they were asked to send their queries to the experiment coordinator to pre-process them. More details about the queries and dataset are discussed in Chapter 5.

Forty subjects were recruited to participate in the experiment. Most of the subjects had IT background and experiences of using search engines. Detailed demographic data of the subjects are presented in Chapter 5.

4.3.2 Experimental Procedure

The subjects first logged in the experimental prototype system. Then they would go through the experimental introduction and instructions. A screenshot of the system interface and experimental steps is shown in Figure 4.3. After going through the introduction and instructions, subjects were asked to read and sign a consent form, and fill out a pre-evaluation questionnaire which was used to collect their demographic data and background information related to search engines. The next step was to execute queries and evaluate the search results. For each query, subjects evaluated the returned websites in terms of how relevant they were to the query, based on the 7-point Likert scale. Before doing a search, subjects would browse the query website (a person's

website) to understand the query site. Figure 4.4 shows a query site. Figure 4.5 shows an example of the search results of a query. Each returned hit was a personal website. It contained a title, a URL and a short description of the personal website. Subjects might judge the relevance of a search hit by reading its description or by browsing this person's website, and the latter was encouraged. Subjects were also asked to give their confidence level on understanding the query website and each of the returned websites evaluated. The confidence level was based on the 7-point Likert scale. If the confidence level a query site received was lower than 4, the middle point of the 7-point Likert scale, the evaluation result for this query would not be used in the final data analysis. Similarly, if the confidence level a returned hit received was lower than 4, the evaluation results for this hit would not be used in the data analysis, either. After finishing the experiment, subjects were asked to answer a post-questionnaire.

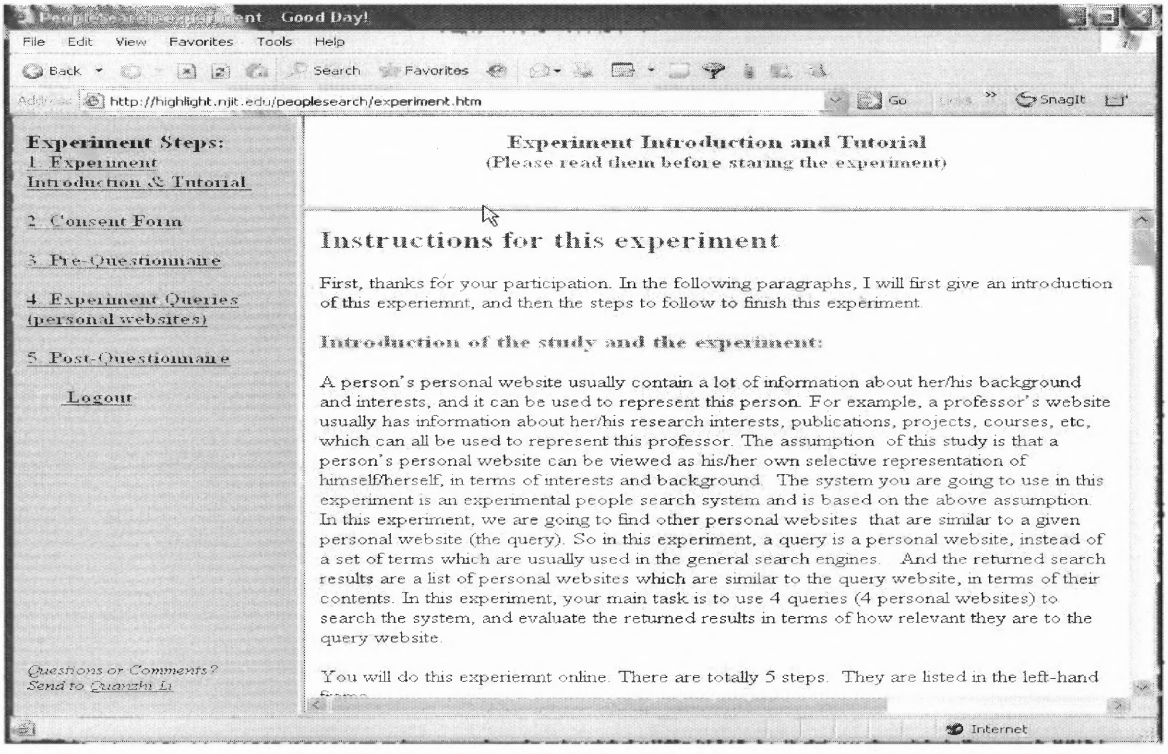

Figure 4.3 A screenshot of the experimental site.

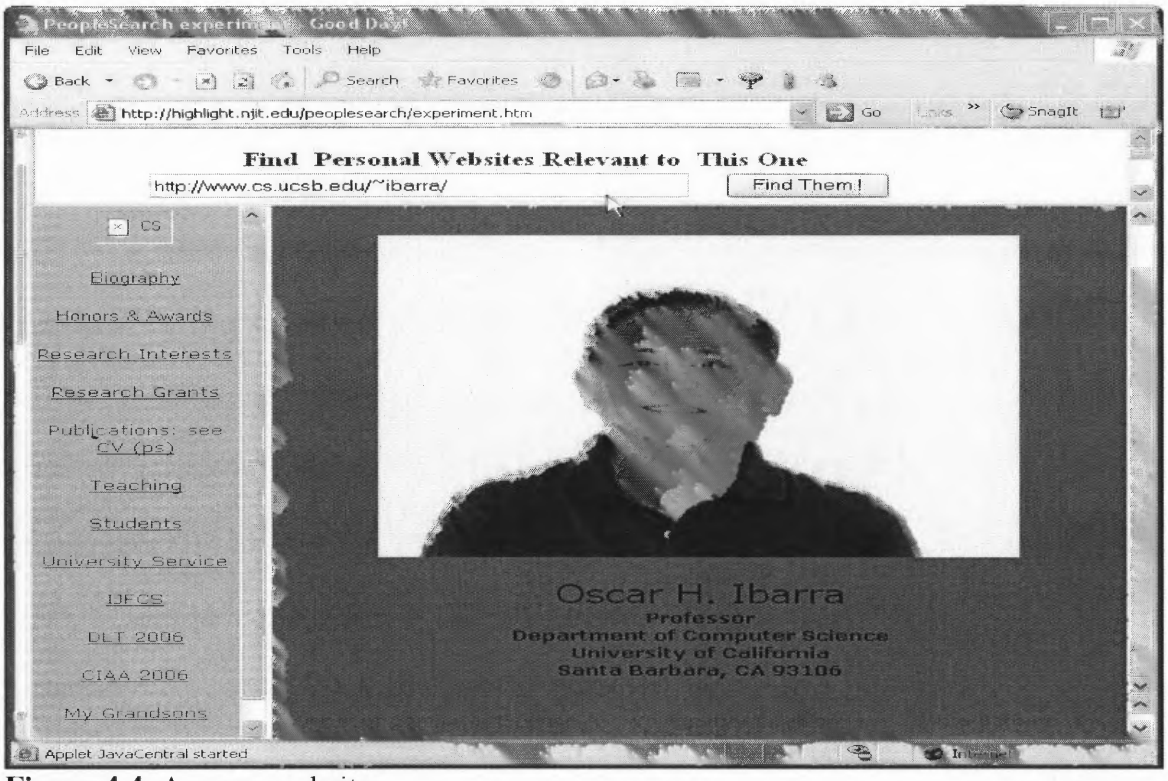

Figure 4.4 A query website.

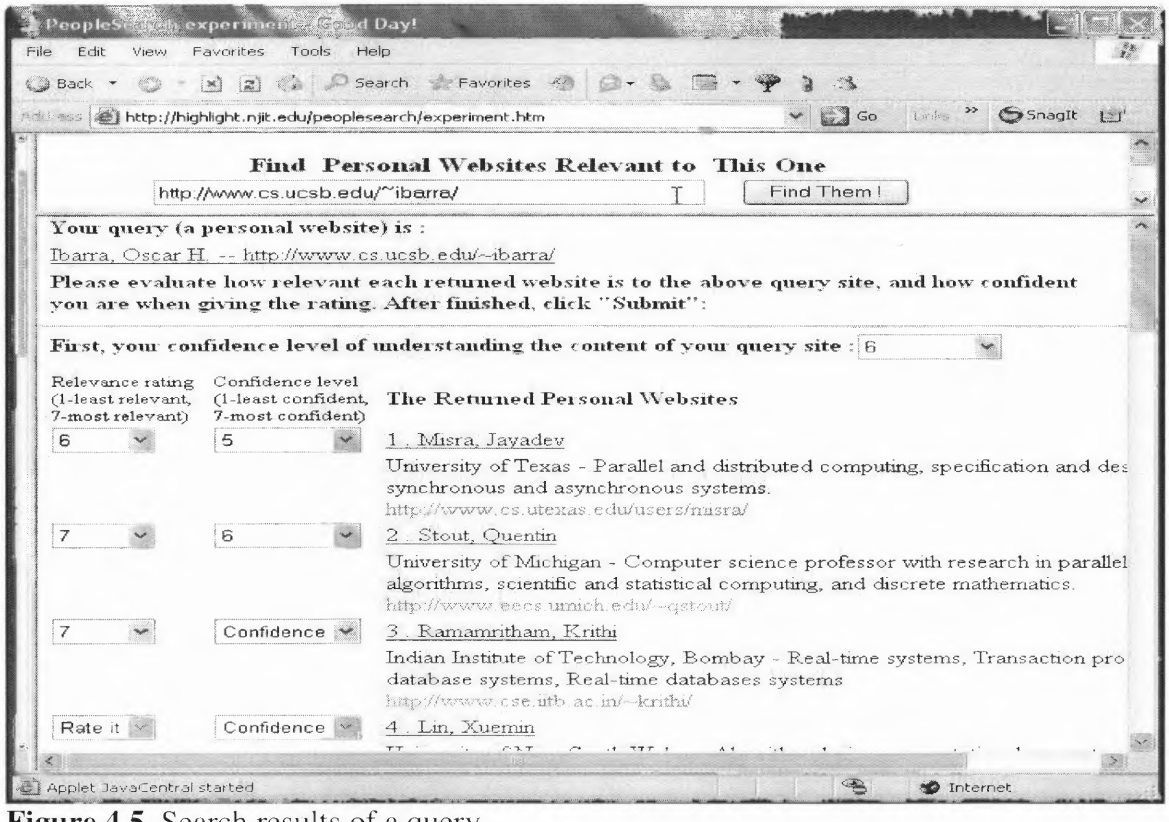

Figure 4.5 Search results of a query.

4.3.3 Data Analysis

This section summarizes the experimental results to be presented and the types of analyses to be performed. The results and analyses are presented in Chapter 5.

Pre-questionnaire Result: The demographic data of the participants and their background information regarding their experiences of using search engine were analyzed.

Queries and Related Statistics: some statistical data related to the queries and subjects' searching behavior during the experiment are reported and analyzed in Chapter 5, such as the number of returned websites opened by the subjects, and the average time spent on each returned hit.

Subject Confidence on Understanding Queries and Returned Results: the data about the subject confidence on understanding the queries and search results will be reported.

Inter-subject Agreement: for human evaluation, inter-rater agreement is one important factor to be considered. The inter-subject agreement on the ratings of the search results was analyzed. The agreement value for each query was first calculated. Then the average agreement value over all the queries was calculated. The Kendall Coefficient of Concordance (W) method was used to measure the inter-subject agreement (Siegel and Castellan, 1988).

Comparison of the Three Algorithms: The three algorithms were compared based on subject ratings of the search results. They were compared in four situations: when top 5, top 10, top 15 and top 20 search hits were considered. Paired t-test was used to test the significance of the results.

Correlation between Results of the Human Evaluation and the Automatic Evaluation: In the automatic evaluation, each algorithm's precision was calculated when the number of search results was 10, 20 30, 40 and 50. In the human evaluation, only the top 20 search results were rated by subjects. Therefore, the correlation between these two kinds of evaluation results was calculated only when the number of returned hits was 10 and 20. The correlation calculation was based on the results of the 14 queries in the human evaluation, as well as in the automatic evaluation. The Pearson's correlation coefficient was used as the correlation measure.

People-Search Algorithm's Effectiveness on Ranking Returned Results: based on subjects' ratings, the People-Search algorithm's effectiveness on ranking search results was calculated. The method used was similar to the one used in evaluating People-Search algorithm's ranking effectiveness in the automatic evaluation. First the top 20 returned results of a query were divided into four groups: top 1 to 5 as group 1, top 6 to 10 as group 2, top 11 to 15 as group 3 and top 16 to 20 as group 4. Then the average human rating for each group was computed, and the four groups were compared to each other based on their average ratings. Paired t-test was used to test the significance of the results.

Post-Questionnaire Result: subjects' opinions about the prototype system and the experiment will also be presented.

4.4 Summary

In this chapter, the evaluation methods used in this study were introduced. The evaluation includes two parts, the automatic evaluation and the human evaluation. In the automatic evaluation, the 14 algorithms were evaluated, using precision, recall, F measure and Kruskal-Goodman Γ measure. In the human evaluation, three algorithms were evaluated. In Chapter 5, the experimental results and data analyses are presented.

CHAPTER 5

EXPERIMENTAL RESULTS AND DATA ANALYSIS

In this chapter, the experimental results and data analyses for the both automatic evaluation and the human evaluation are presented.

5.1 Experimental Dataset Analysis

Section 4.1 has briefly introduced the dataset, i.e., 20,000 personal websites crawled from the Web. The distribution of the crawled personal websites among the 11 ODP (DMOZ) top categories is shown in Table 4.1 of Chapter 4. The descriptive data of this dataset are presented below.

Table 5.1 shows the related statistics of the crawled websites. This table shows that on average a personal website contains about 37 pages, including the main page and all sub pages. Among the 20,000 crawled sites, about 10.2% (2034/20000=10.2%) of them only have one page, i.e., the main page.

Table 5.1 Information about the Crawled Websites

Total number of websites	Total number of web pages	Average number of pages per site	Std	Lowest number of pages a site has	Highest number of pages a site has	Number of sites having only the main page
20,000	740,230	36.6	40.7	1	301	2,034

After the websites were crawled, each page's inlinks were obtained from Yahoo and Google, and then were combined together. DMOZ directory has existed for nearly

ten years, and it powers the core directory services for some of the most popular search engines and web portals. Because of its high quality and popularity, there are many mirror sites of the entire or partial DMOZ directory on the Web. This is also true for the DMOZ personal website category. Due to this reason, the personal home pages listed in the same DMOZ category page have many co-inlinks. This will increase the link similarity between any two personal websites listed in the same DMOZ category page. To avoid the bias caused by this fact, and to make the experimental data cleaner, all the collected inlinks were examined and the inlinks which were the mirror sites of DMOZ pages were removed. The same problem also existed for outlinks, though it was not as serious as that of inlinks - two persons' websites might have the same outlink pointing to the same DMOZ page where they were listed in. Therefore, all the outlinks were also examined and such kind of outlinks were also removed.

Table 5.2 presents related statistics about inlinks. The data were obtained after the inlink removal process mentioned above. Some algorithms (Site_Content_Link, Site_Content_Inlink, Site_Link and Site_Inlink) need inlinks information for the entire website, some other algorithms (MainPage_Content_Link, Mainpage_Content_Inlink, MainPage_Link and MainPage_Inlink) need inlink information for only the main page. Therefore, this table has two parts: one for an entire website, and the other part for only the main page of a website. This table shows that on average a personal website has about 30 unique inlinks. These inlinks include the inlinks for the main page as well as all the sub pages. Among the 20,000 websites, about 3.7% (747/20000 =3.7%) of them do not have any inlink. The average number of unique inlinks for the main pages is about 24.

These two average numbers, 30.3 and 24.2, show that, on average, 80% (24.2/30.3=80%) of the inlinks of a personal website point to the main page.

Table 5.2 Inlink Distribution

Entire Website					Main Page				
Average number of unique inlinks a website has	Std	Lowest number of unique inlinks a website has	Highest number of unique inlinks a website has	Number of websites that have no inlink	Average number of unique inlinks a main page has	Std	Lowest number of unique inlinks a main page has	Highest number of unique inlinks a main page has	Number of main pages that have no inlink
30.3	49.8	0	1,171	747	24.2	38.7	0	734	1,024

Outlink information is used in eight of the 14 algorithms. Table 5.3 shows some statistics about outlinks. It also has two parts, one for the entire website, the other one for only the main page. This table shows that the average number of outlinks for a personal website is 70.6, while this number is 7.6 for the main page. This shows that only 10.8% of the outlinks are from the main page. In contrast, as mentioned above, 80% of a personal website's inlinks point to the main page. 11.5% (2283/20000=11.5%) of the sites do not have any outlink. 31.3% of the main pages do not have any outlink.

Table 5.3 Outlink Distribution

Entire Website					Main Page				
Average number of unique outlinks a website has	Std	Lowest number of unique outlinks a website has	Highest number of unique outlinks a website has	Number of websites that have no outlink	Average number of unique outlinks a main page has	Std	Lowest number of unique outlinks a main page has	Highest number of unique outlinks a main page has	Number of main pages that have no outlink
70.6	246.2	0	8,357	2,283	7.6	20.5	0	282	6,259

For each website, the content-bearing words were extracted from each of its pages. Table 5.4 shows how many terms a website and main page contains. The terms referred in Table 5.4 are all content-bearing words after word stemming. For example, "work," worked" and "working" are treated as one term "work." This table shows that on average a personal website has 581 terms, while the main page has 73 terms.

Table 5.4 Number of Terms in a Website and Main Page

Entire Website				Main Page			
Average number of unique terms a website has	Std	Lowest number of unique terms a website has	Highest number of unique terms a website has	Average number of unique terms a main page has	Std	Lowest number of unique terms a main page has	Highest number of unique terms a main page has
581	2,768	12	17,827	73	221	3	2,012

Table 5.5 shows term distribution among all the crawled websites. Table 5.6 shows the most frequent 15 terms among the 20,000 websites. These terms are all very common terms in personal home pages, as well as in the ordinary web pages. Because their frequencies are very high, based on the term weight formula (TF.IWF) introduced in Chapter 4, their weights are very low in the calculation of content similarity between two websites.

Table 5.5 Term Distribution among All Websites

Total number of unique terms for the 20000 websites	Average number of websites a term appears in	Std	Lowest number of websites a term appears in	Highest number of websites a term appears in
355,474	11	51	1	3,033

Table 5.6 The 15 Most Frequent Terms

Top 15 terms	Number of websites the term appears in
Work	3,033
Site	2,894
Page	2,745
Home	2,715
Time	2,639
Make	2,513
Link	2,498
Show	2,400
Image	2,255
Information	2,228
View	2,159
Life	2,147
Interest	2,138
Web	2,121
Design	2,094

5.2 Algorithm Parameter Values

In Chapter 3, several algorithm parameters have been introduced. To determine the optimal values for those parameters, a genetic algorithm was used to optimize them. The genetic algorithm and how it tuned the algorithm parameters have been explained in Chapter 4. The results are reported in this section. The training dataset was gathered from the same master dataset described in Section 5.1. Five hundred personal websites were randomly selected as the queries. They were different from the queries used for automatic or human evaluation. The parameters and their optimal values are listed in Table 5.7.

Table 5.7 Algorithm Parameter Values

Parameter	Usage	Value
$\beta_{site\text{-}link}$	Adjusting $S_{content}$ and S_{link} for Site_Content_Link algorithm	0.7
$\beta_{site\text{-}inink}$	Adjusting $S_{content}$ and S_{inlink} for Site_Content_Inlink algorithm	0.77
$\beta_{site\text{-}outlink}$	Adjusting $S_{content}$ and $S_{outlink}$ for Site_Content_Outlink algorithm	0.84
α_{site}	Adjusting S_{inlink} and $S_{outlink}$ for Site_Link algorithm	0.62
$\beta_{page\text{-}link}$	Adjusting $S_{content}$ and S_{link} for MainPage_Content_Link algorithm	0.75
$\beta_{page\text{-}inlink}$	Adjusting $S_{content}$ and S_{inlink} for MainPage_Content_Inlink algorithm	0.78
$\beta_{page\text{-}outlink}$	Adjusting $S_{content}$ and $S_{outlink}$ for MainPage_Content_Outlink algorithm	0.92
α_{page}	Adjusting S_{inlink} and $S_{outlink}$ for MainPage_Link algorithm	0.84
p_m	Adjusting a word's weight if it appears in the main page	1.08
p_c	Adjusting a word's weight if it is capitalized	1.04
p_b	Adjusting a word's weight if it is in bold	1.02
p_l	Adjusting a word's weight if it is in larger font	1
p_t	Adjusting a word's weight if it is in title or meta-description	1.22
p_{ml}	Adjusting a link's weight if it is for the main page	1.1

α and β are the two most important parameters. α is used to combine inlink similarity and outlink similarity. β is used to combine content similarity and link (link, inlink or outlink) similarity. There are two kinds of α parameters: α_{site} and α_{page}. Table 5.7 shows that both α_{site} and α_{page} are greater than 0.5, which means inlinks are more important than outlinks. α_{page} is greater than α_{site} (0.84 vs. 0.62). This is reasonable, since

the average number of outlinks for a main page is much less than that for an entire website (70.6 vs. 7.6), while that difference between their inlinks is much smaller (30.3 vs. 24.2). There are six types of β parameters. Their meanings and usages have been described in Chapter 3 and are also briefly explained in Table 5.7. Their values range from 0.7 to 0.92. $\beta_{page\text{-}outlink}$ is used to adjust the weights of a main page's content similarity and outlink similarity, and it has the highest β value, 0.92. Part of the reason that $\beta_{page\text{-}outlink}$ has a high value is that usually a main page has very few outlinks, as mentioned above. Therefore, the outlink similarity based on the main pages is not a good similarity predictor, and it accounts for only a small portion of the combined similarity value in algorithm MainPage_Content_Outlink.

Several *p* parameters are used to adjust the weights of words or links when they are specifically emphasized. The p_t parameter has the highest p value, 1.22, which indicates that words appearing in the title or meta-description of a web page should be given a higher weight than the regular words appearing only in the body of the page. Three p parameters (p_l, p_b and p_c) have a value of 1 or very close to 1. This means whether or not a word is capitalized, in bold or in a larger font, has no or very small effect in calculating website similarity. Sometimes people use bold, larger font or capitalization to emphasize important concepts. But on the other hand, in web pages, very often they are also used to highlight the terms that are used to attract viewers' attention to the paragraph or sentences following them. In the latter case, the highlighted terms are too common and have no specific meaning to the theme of that page. For example, "My Research Interests" and "My Favorite Links" are usually in a larger font, in bold or capitalized, but they are very common terms and actually their weights should be

decreased. Because of the fact just mentioned, the values of these three parameters are close or equal to zero.

5.3 Automatic Evaluation Results and Analysis

In this study, two kinds of automatic evaluations were conducted to evaluate the 14 algorithms. One used the traditional IR measures, which are precision, recall and F measure; the other one used Kruskal-Goodman Γ measure. Before reporting the results in Section 5.3.2 and 5.3.3, the result of comparing the three link similarity measures is first presented.

5.3.1 Comparison Results of the Three Link Similarity Measures

The procedure to compare the three similarity measures (cosine, Euclidean distance and Jaccard coefficient) has been discussed in Section 4.2.1. Table 5.8 shows the comparison results when they were tested in calculating inlink similarity. Table 5.9 displays the results for outlink similarity calculation. The results were based on Kruskal-Goodman Γ measure. Paired t-test was used as the significance test. In both cases, the cosine similarity measure outperformed the other two, and it was statistically significant at the level of $p < 0.01$. The results support the choice – using cosine similarity measure to calculate the link similarity between two personal websites.

These two tables also show that, in inlink similarity calculation, although Jaccard coefficient's performance was lower than the other two's, the difference was not big (0.289 vs. 0.301 and 0.314). However, in the outlink similarity calculation, the difference was much bigger (0.228 vs. 0.261 and 0.272). This is mainly caused by two facts: inlinks

and outlinks of personal websites have different characteristics, and Jaccard coefficient uses different link weighting method from the other two methods. Personal websites usually have a lot of common outlinks, such as google.com and cnn.com; and these kinds of outlinks have no or very little relationship with the personal websites they belong to, in terms of their contents. The cosine measure and distance measure have taken care of this problem by using the LF.IWF weighting method (see Section 3.4.3 for details), which will decrease the importance of this kind of outlinks in the similarity calculation by adjusting their weights. Because Jaccard coefficient only considers the absence and presence of a link, it treats this kind of common but barely useful outlinks as important as other outlinks. These two facts together caused the low performance of Jaccard coefficient in outlink similarity calculation.

Table 5.8 Comparison Results of the Three Measures in Inlink Similarity Calculation

Similarity measure	Average Γ value	Significance test on the mean difference between a measure and the next best one
Cosine	0.314	$p < 0.01$
Euclidean distance	0.301	$p < 0.01$
Jaccard coefficient	0.289	N/A

Table 5.9 Comparison Results of the Three Measures in Outlink Similarity Calculation

Similarity measure	Average Γ value	Significance test on the mean difference between a measure and the next best one
Cosine	0.272	$p < 0.01$
Euclidean distance	0.261	$p < 0.01$
Jaccard coefficient	0.228	N/A

5.3.2 Experimental Results Using Precision, Recall and F Measure

In this section, first the performance of the 14 algorithms obtained by using domain-independent queries is reported. Then the comparison results of the top five algorithms in three individual domains are presented. And finally, how effective the People-Search algorithm was on ranking search results is presented.

5.3.2.1 Results for Queries from All Categories. As described in Chapter 4, the 2,000 queries were randomly selected from all top ODP categories in the dataset. Therefore, the experimental results show the performance of the 14 algorithms over all domains. Due to the large amount of data, the results are organized into two tables: Table 5.10 presents the results for the seven algorithms that use information from an entire personal website; and Table 5.11 presents the results for the other seven algorithms which use information only from the main page. In the two tables, N is the number of returned search results for each query. P refers to precision, R means recall and F is the F measure explained in Chapter 4.

In Table 5.10, from left to right, the algorithms are listed in the descending order of their performance. The results show that the People-Search algorithm outperformed all the other 6 algorithms. The Site_Outlink algorithm performed the worst. The paired t-test was used to test the significance of the results. T-test (Snedecor & Cochran, 1989) is a type of significance test which is used to determine if two populations' means are significantly different. The data may either be paired or not paired. For paired t-test, the number of points in each dataset must be the same, and the data must be organized in pairs, in which there is a definite relationship between each pair of data points. In this experiment, there were 2,000 queries, so the data size was 2,000. When comparing two

Table 5.10 Results of the seven Algorithms Using Information from an Entire Website

N	Measure	Site_Content_Link (People-Search algorithm)	Site_Content_Inlink	Site_Content_Outlink	Site_Content	Site_Link	Site_Inlink	Site_Outlink
10	**P**	**0.732**	**0.707**	**0.672**	**0.630**	**0.491**	**0.361**	**0.270**
	R	0.078	0.073	0.069	0.066	0.055	0.050	0.033
	F	0.112	0.105	0.097	0.090	0.073	0.065	0.041
20	**P**	**0.671**	**0.651**	**0.626**	**0.600**	**0.376**	**0.228**	**0.201**
	R	0.114	0.109	0.099	0.095	0.07	0.061	0.038
	F	0.135	0.131	0.125	0.120	0.099	0.073	0.049
30	**P**	**0.629**	**0.617**	**0.598**	**0.578**	**0.272**	**0.152**	**0.144**
	R	0.135	0.129	0.123	0.118	0.074	0.063	0.040
	F	0.152	0.145	0.137	0.131	0.105	0.066	0.053
40	**P**	**0.600**	**0.593**	**0.574**	**0.560**	**0.215**	**0.117**	**0.112**
	R	0.152	0.145	0.132	0.123	0.075	0.063	0.042
	F	0.163	0.157	0.146	0.137	0.1	0.059	0.053
50	**P**	**0.578**	**0.572**	**0.552**	**0.549**	**0.171**	**0.094**	**0.089**
	R	0.165	0.161	0.140	0.137	0.075	0.063	0.042
	F	0.176	0.171	0.152	0.149	0.095	0.054	0.053

systems, for each measure, there were 2,000 pairs of data points (one pair for each query). Fifteen paired t-tests were conducted when comparing two systems; each test corresponded to one measure point of a performance measure (e.g., the precision when the number of returned documents was 20). There are seven systems, so totally there were 21 system-system comparisons. Therefore, overall, 315 paired t-tests were conducted for these seven algorithms.

Based on the results of the paired t-tests, the performance difference between any two systems was statistically significant for all three measures, except in the following situations: (1) when N=50, the difference between Site_Content_Link and Site_Content_Inlink, and the difference between Site_Content_Outlink and Site_Content were not significant at $p < 0.01$ level for any of the three measures; and (2) when N was 40 and 50, the precision difference between Site_Inlink and Site_Outlink was not statistically significant.

Paired t-test is a significance test and as such is made up of two components, the effect size and the size of the study. In this experiment, the effect size referred to the degree of performance difference between two systems. Table 5.10 shows that the performance difference between some algorithms, such as Site_Content_Outlink and Site_Content, was not large, but the results were still statistically significant. This was due to the large size of the study, which was 2,000 queries in this experiment.

In the three measures, precision is the most important one, since it directly affects users' feeling and evaluation about an IR system. In order to make the comparison of these algorithms clearer, the precisions of the seven algorithms are plotted in Figure 5.1. Figure 5.1 clearly shows that the four algorithms (Site_Content_Link,

Site_Content_Inlink, Site_Content_Outlink and Site_Content) performed much better than the other three algorithms, which mainly rely on link information. It also shows that the precision of the algorithms relying only on link information decreased very fast, while the precision of the algorithms using content information decreased at a slower pace. Part of the reason is that a personal website generally only shares few co-inlinks or co-outlinks with other websites, therefore, when the number of returned hits increased, only very few more relevant websites would be returned. This figure also shows that when N increased, the performance difference between the top four algorithms decreased. This is because all these four algorithms use content information of a personal website, but they differ in how or whether or not to use the link information. When N increased, as mentioned above, the effect of link information decreased, therefore the content information dominated in these four algorithms. Since these four algorithms share the same content information, their performance was closer to each other when N increased.

Table 5.11 shows the results of the three traditional IR measures for the seven algorithms that exploit information from only the main page of a personal website. From left to right, the algorithms are listed in the descending order of their performance. Based on the results, the MainPage_Content_Link algorithm was the best algorithm, and the Site_Outlink algorithm was the worst. The paired t-test was used to test the significance of the results. Overall, 315 paired t-tests were conducted for these seven algorithms. Based on the results of the paired t-tests, the performance difference between any two systems was statistically significant at the level of $p < 0.01$, for all three measures except in the following three situations: (1) when N was 30, 40 and 50, the performance difference between MainPage_Content_Link and Mainpage_Content_Inlink was not

significant for any of the three measures; (2) when N was 40 or 50, the difference between MainPage_Inlink and MainPage_Outlink was not significant for any of the three measures; and (3) when N was 30, MainPage_Inlink performed better than MainPage_Outlink only at $p < 0.05$ level for all the three IR measures.

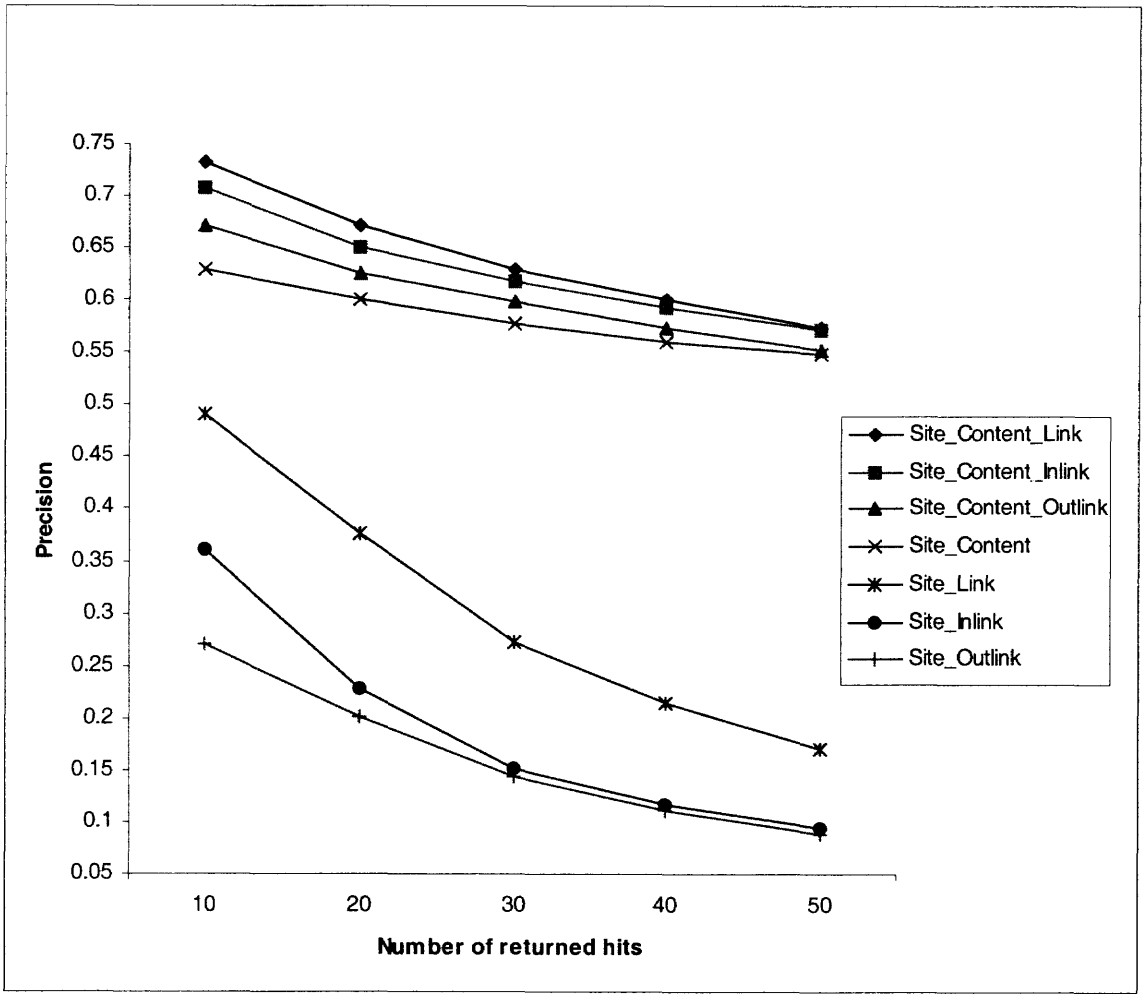

Figure 5.1 Precision for algorithms using information from an entire website.

Table 5.11 Results of the seven Algorithms Using Information from only the Main Page

N	Measure	MainPage_Content_Link	MainPage_Content_Inlink	MainPage_Content_Outlink	MainPage_Content	MainPage_Link	MainPage_Inlink	MainPage_Outlink
10	**P**	**0.68**	**0.662**	**0.568**	**0.548**	**0.382**	**0.345**	**0.073**
	R	0.071	0.068	0.061	0.056	0.051	0.048	0.005
	F	0.1	0.095	0.078	0.074	0.067	0.062	0.008
20	**P**	**0.603**	**0.596**	**0.531**	**0.523**	**0.235**	**0.208**	**0.042**
	R	0.094	0.091	0.082	0.079	0.063	0.059	0.005
	F	0.121	0.119	0.109	0.108	0.075	0.069	0.008
30	**P**	**0.559**	**0.556**	**0.505**	**0.502**	**0.162**	**0.142**	**0.028**
	R	0.106	0.104	0.096	0.094	0.064	0.06	0.005
	F	0.125	0.124	0.118	0.116	0.066	0.061	0.007
40	**P**	**0.529**	**0.528**	**0.488**	**0.487**	**0.12**	**0.106**	**0.021**
	R	0.119	0.117	0.112	0.112	0.064	0.06	0.005
	F	0.133	0.131	0.127	0.126	0.059	0.054	0.007
50	**P**	**0.508**	**0.508**	**0.474**	**0.473**	**0.097**	**0.085**	**0.017**
	R	0.127	0.127	0.122	0.122	0.064	0.06	0.005
	F	0.138	0.137	0.136	0.136	0.053	0.048	0.006

The precisions of the seven algorithms exploiting information from only the main page are also plotted in Figure 5.2. This figure shows that the MainPage_Outlink algorithm performed the worst. Its precision was only 0.07 when N was 10. This algorithm uses only the outlink information of the main page in a personal website. On average, a main page has only 7.6 outlinks, compared to 24.2 inlinks a main page has and 70.6 outlinks an entire website has. The much less amount of outlinks a main page has is part of the reason why MainPage_Outlink performed the worst. Another reason is that outlink is a bad indicator of similarity between two websites, compared to inlink. This can be seen from the experimental results for the algorithms exploiting information from the entire website (see Table 5.10 and Figure 5.1). The Site_Inlink algorithm performed better than Site_Outlink, though the average number of inlinks for an entire personal website is less than that of outlinks. Table 5.2 and Table 5.3 show that the average number of inlinks a personal website has is 30.3. In contrast, the average number of outlinks is 70.6 for an entire website, which is much higher than the number of inlinks. This shows inlink is a better indicator than outlink in finding similar personal websites. For a person's website, its inlinks usually either are relevant to it in terms of the content, or have some kind of organizational relationship with it (e.g., www.is.njit.edu points to an NJIT Information System professor's website). In contrast, lots of outlinks of a person's website point to some sites that are not relevant to it in terms of content relevance or organizational structure. For example, many people's websites have outlinks pointing to google.com, ebay.com, and cnn.com.

Figure 5.2 also shows that the performance difference between MainPage_Content_Link and MainPage_Content_Inlink was very small (same for

MainPage_Content_Outlink and MainPage_Content). The difference was not statistically significant at $p < 0.01$ level. The reason is that the only difference between these two algorithms is whether or not to include the outlink information, and the outlink information contributes very little in algorithms using only main page information, as described before.

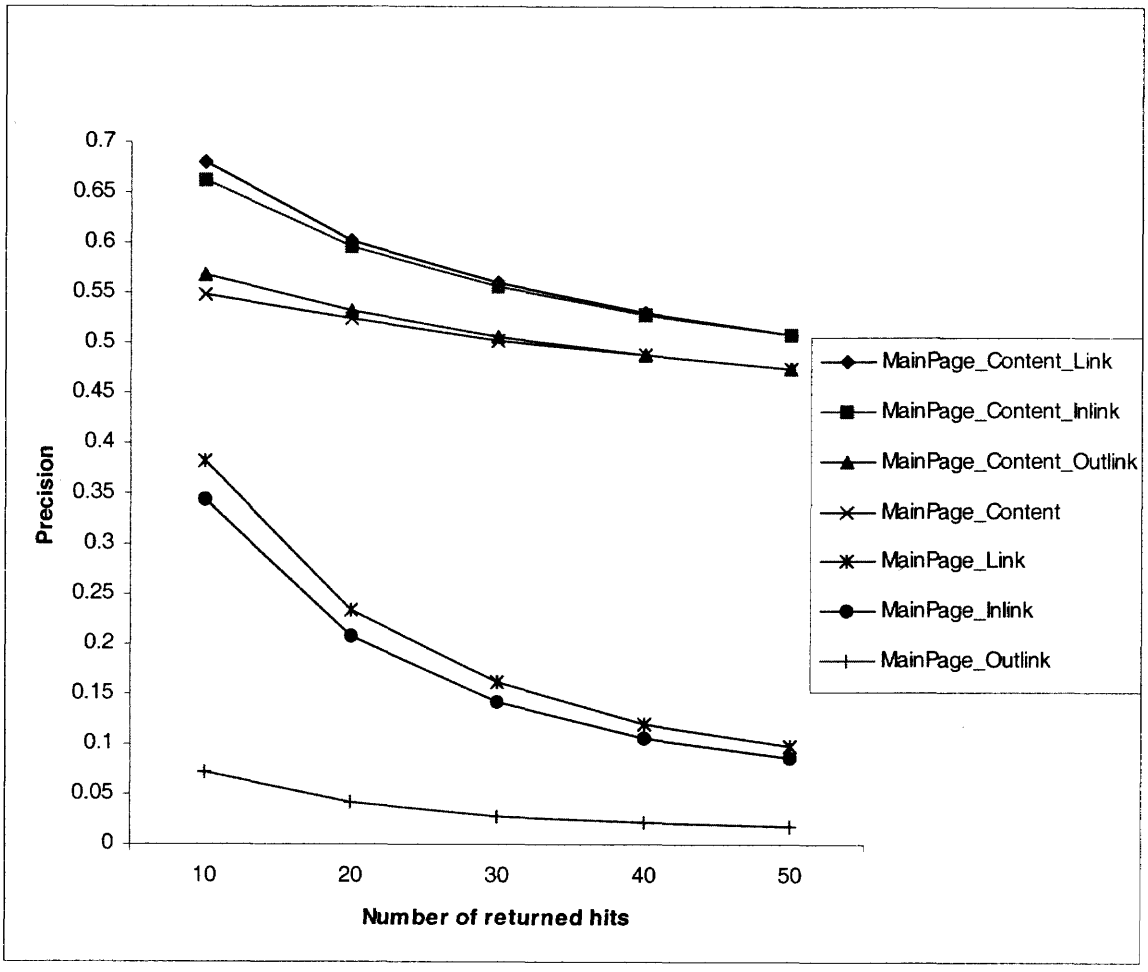

Figure 5.2 Precision for algorithms using information from only the main page.

The 14 algorithms are divided into two groups and their results are reported in two tables and figures. To better understand which algorithm performed the best, the precisions of the top algorithms from the two groups are plotted in one figure. Figure 5.3 shows precisions of the top six algorithms. Four of them are from the first group, in

which the algorithms use information of an entire website, and two of them are from group 2, whose algorithms use information from only the main page. They are chosen because they obviously outperformed the other algorithms in their groups (see Figure 5.1 and 5.2).

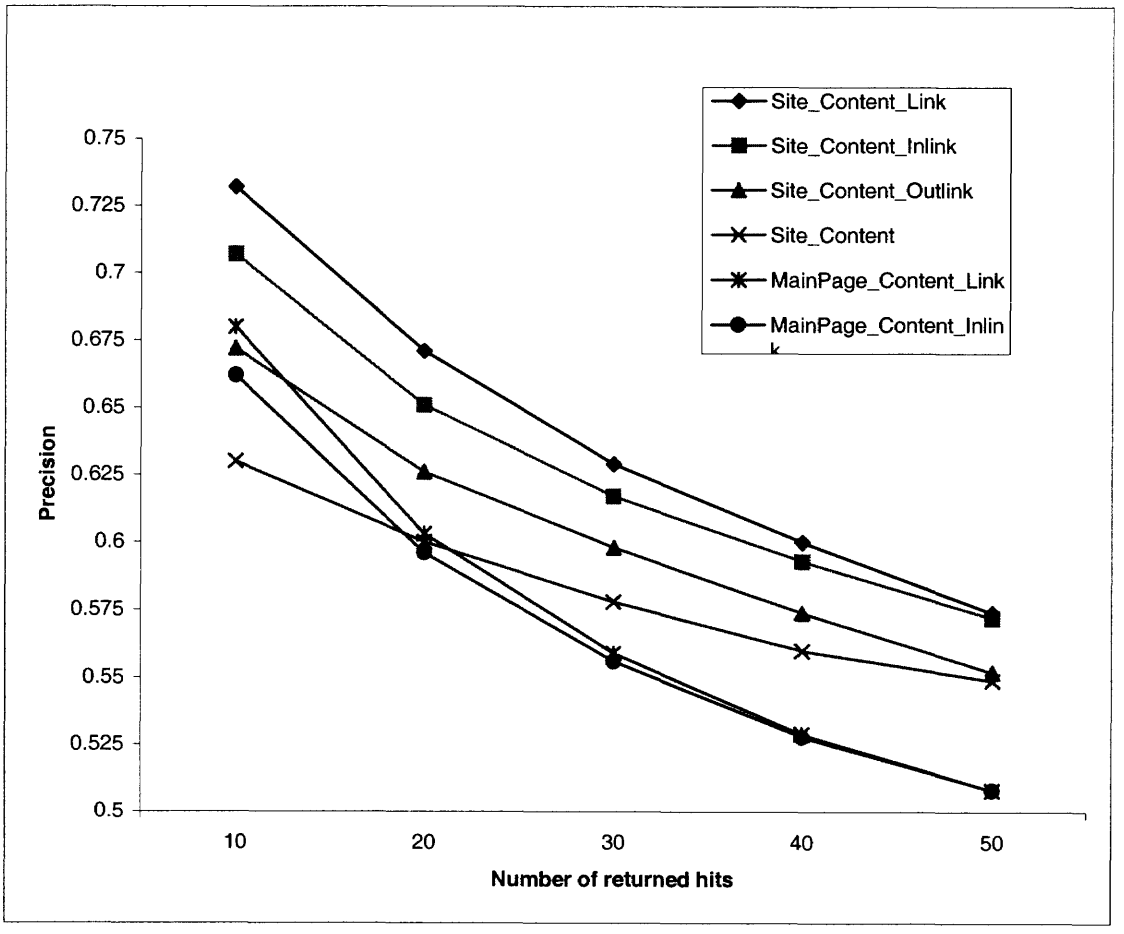

Figure 5.3 Precision for the top six algorithms.

Figure 5.3 shows that the People-Search algorithm was the best. The People-Search algorithm and Site_Content_Inlink outperformed all other algorithms, and the results were statistically significant at $p < 0.01$, based on paired t-tests. For the other four algorithms, when N was small (10 or 20), their precisions were close to each other; when N was large, Site_Content_Outlink and Site_Content outperformed MianPage_Content_Link and MainPage_Content_Inlink.

The experimental results based on cross-domain queries show that the People-Search algorithm was the best.

5.3.2.2 Results for Queries from Three Individual Domains. The experimental results reported in the last section are based on queries drawn from all domains. To see if the People-Search algorithm still outperforms the rest in each individual domain, the top five algorithms, which were chosen based on the cross-domain experimental results, were also tested in three individual domains, Arts, Sports and Computers. These five algorithms are: the People-Search algorithm, Site_Content_Inlink, Site_Content_Outlink, Site_Content, and MainPage_Content_Link. The MainPage_Content_Inlink algorithm was not included because MainPage_Content_Link and MainPage_Content_Inlink had similar performance. The only difference between these two algorithms is whether or not to include the main page's outlink information. As mentioned in Section 5.3.2.1, the main page's outlink information contributes little to the website similarity calculation.

Five hundred queries were randomly selected from each of the three domains and executed by these five algorithms. Precision, recall and F were calculated when the number of search results was 10, 30 and 50. Table 5.12 shows results for the Arts domain. Table 5.13 shows results for the Sports domain. And Table 5.14 is for the Computers domain. To illustrate their performance clearer, the precisions at N = 10, 30 and 50 for these three domains are also plotted in Figure 5.3, 5.4 and 5.5, respectively.

Table 5.12 Results of the Five Algorithms in Arts Domain

N	Measure	Site_Content_Link	Site_Content_Inlink	Site_Content_Outlink	Site_Content	MainPage_Content_Link
10	**P**	**0.731**	**0.721**	**0.712**	**0.698**	**0.647**
	R	0.064	0.062	0.06	0.057	0.051
	F	0.101	0.097	0.094	0.090	0.078
30	**P**	**0.643**	**0.638**	**0.629**	**0.624**	**0.564**
	R	0.142	0.140	0.138	0.136	0.121
	F	0.202	0.200	0.197	0.196	0.174
50	**P**	**0.584**	**0.583**	**0.577**	**0.574**	**0.483**
	R	0.196	0.196	0.192	0.191	0.162
	F	0.272	0.271	0.266	0.265	0.218

Table 5.13 Results of the Five Algorithms in Sports Domain

N	Measure	Site_Content_Link	Site_Content_Inlink	Site_Content_Outlink	Site_Content	MainPage_Content_Link
10	**P**	**0.657**	**0.644**	**0.632**	**0.613**	**0.607**
	R	0.160	0.153	0.146	0.139	0.138
	F	0.252	0.240	0.228	0.219	0.216
30	**P**	**0.509**	**0.5**	**0.494**	**0.482**	**0.43**
	R	0.34 7	0.341	0.37	0.33	0.302
	F	0.405	0.397	0.389	0.382	0.344
50	**P**	**0.426**	**0.418**	**0.409**	**0.403**	**0.37**
	R	0.472	0.465	0.461	0.452	0.441
	F	0.45	0.441	0.429	0.417	0.394

Table 5.14 Results of the Five Algorithms in Computers Domain

N	Measure	Site_Content_Link	Site_Content_Inlink	Site_Content_Outlink	Site_Content	MainPage_Content_Link
10	**P**	**0.746**	**0.717**	**0.683**	**0.644**	**0.692**
	R	0.109	0.106	0.101	0.096	0.103
	F	0.180	0.175	0.165	0.159	0.172
30	**P**	**0.631**	**0.613**	**0.591**	**0.573**	**0.554**
	R	0.260	0.255	0.247	0.239	0.228
	F	0.357	0.348	0.332	0.320	0.307
50	**P**	**0.568**	**0.557**	**0.549**	**0.535**	**0.502**
	R	0.393	0.388	0.382	0.371	0.358
	F	0.442	0.436	0.427	0.415	0.395

The results in Table 5.12, 5.13 and 5.14 show that the People-Search algorithm outperformed the other four algorithms in all three domains. Paired t-test was used to test the significance of the results. The significance test results show that, in the Sports and Computers domains, the People-Search algorithm performed better than the other four algorithms at the significance level of $p < 0.01$, for all three measures at all three comparison points (N= 10, 30 and 50). In the Arts domain, it performed better than others, except the Site_Content_Inlink algorithm, at $p < 0.01$ level for all measures. For the comparison with the Site_Content_Inlink algorithm, when N=10, the People-Search performed better at the significance level of $p < 0.01$. When N=30, its precision was higher than that of Site_Content_Inlink at $p < 0.05$ level, but their recall and F values had no statistically significant difference. When N=50, there was no statistically significant difference between the two algorithms for all three measures. This can also be seen from Figure 5.4. When N was 30 or 50, the precisions of the two algorithms were very close to each other.

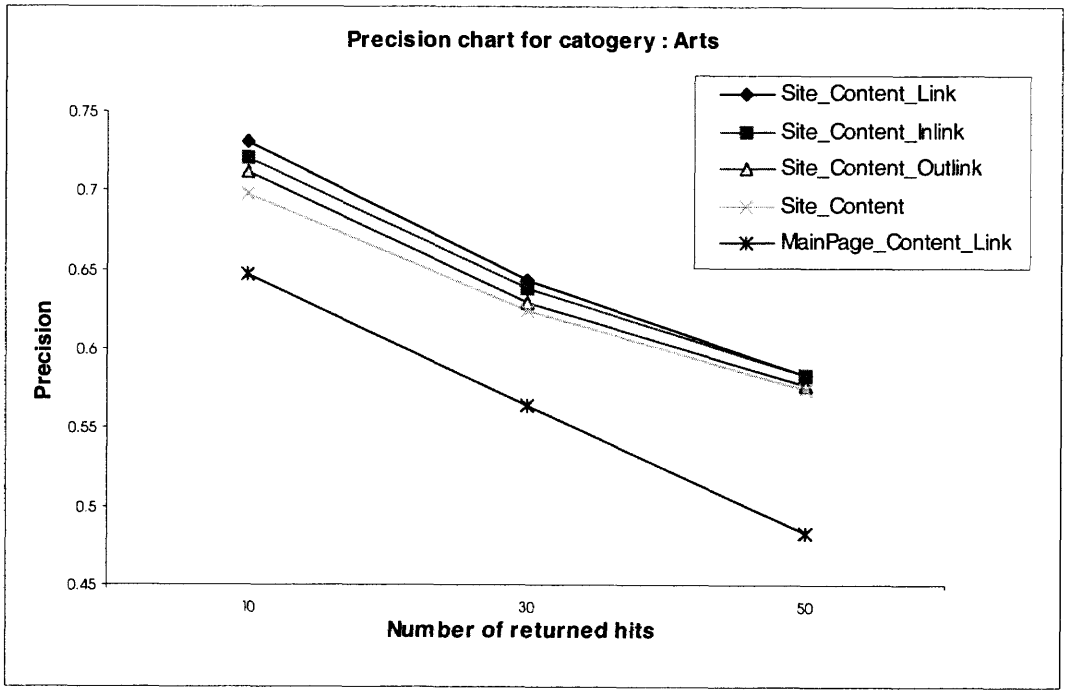

Figure 5.4 Precision for the five algorithms in Arts domain.

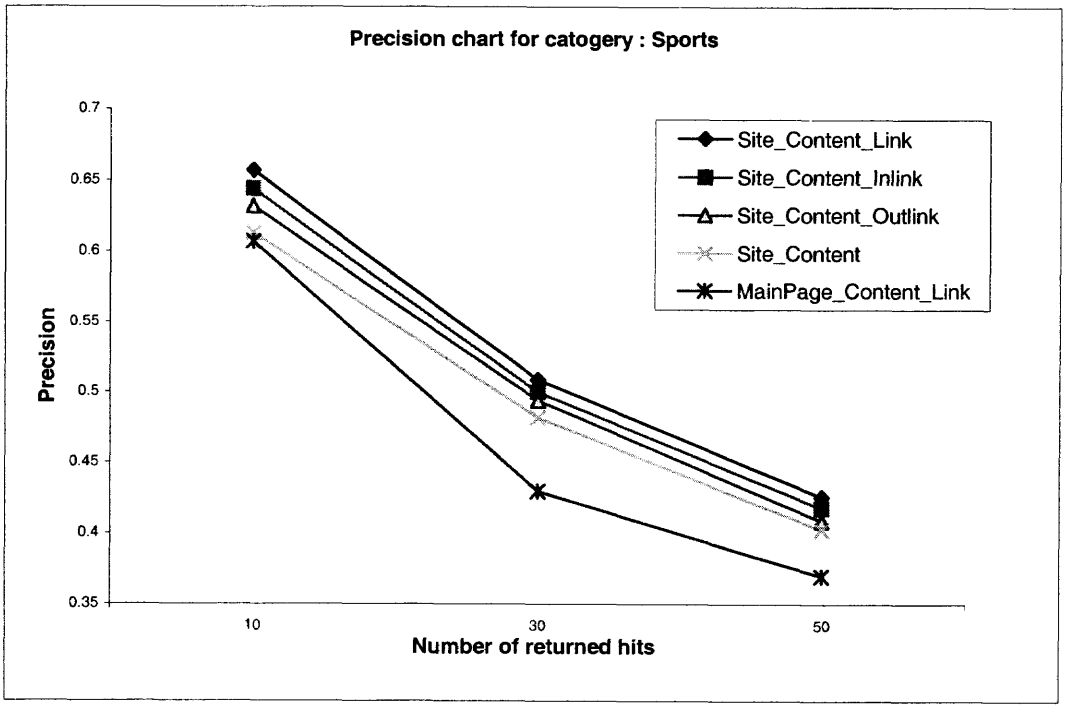

Figure 5.5 Precision for the five algorithms in Sports domain.

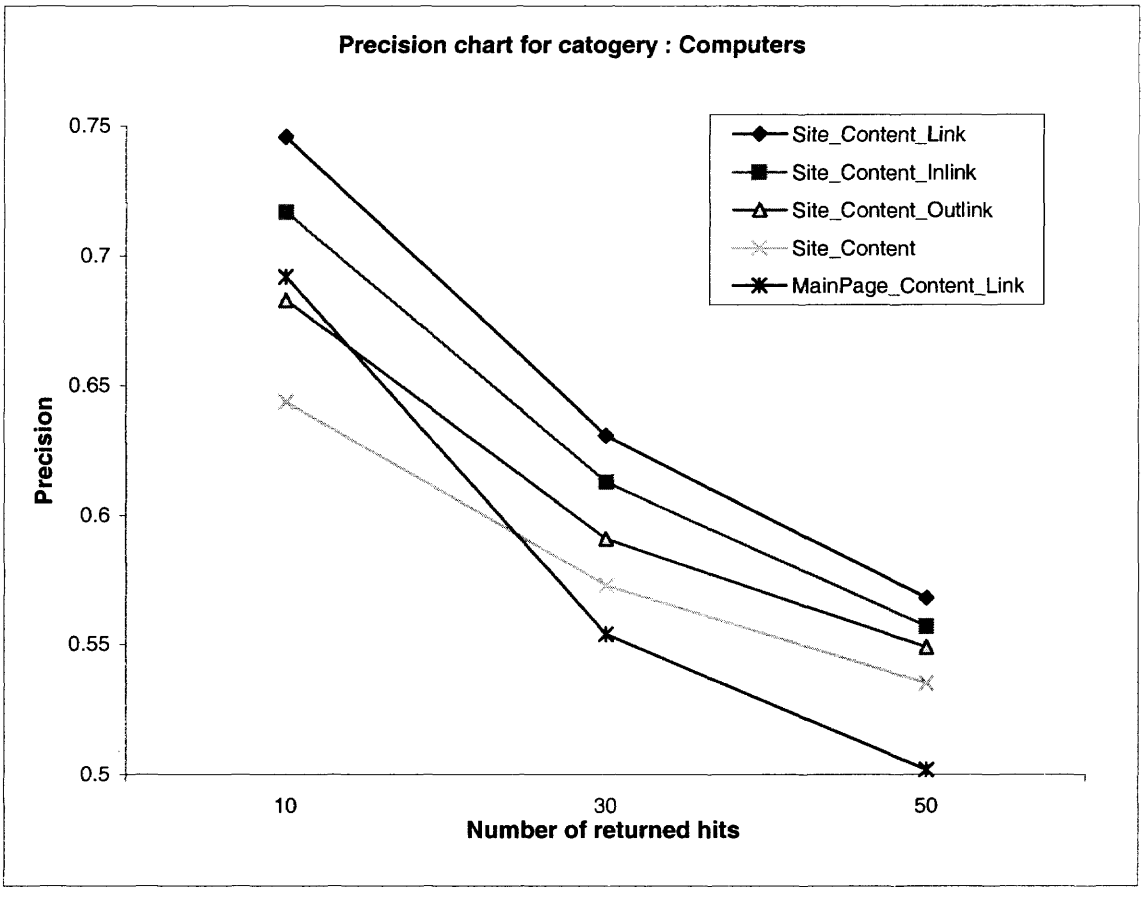

Figure 5.6 Precision for the five algorithms in Computers domain.

Table 5.15 Link Information for the Three Domains

Domain	Entire Website		Main Page	
	Average number of inlinks	Average number of outlinks	Average number of inlinks	Average number of outlinks
Arts	19.5	35.2	14.9	3.5
Sports	25.7	57.0	13.4	7.1
Computers	50.3	69.2	20.5	12.4
All 11 Domains	30.2	70.6	18.2	7.6

Figure 5.4, 5.5 and 5.6 also show some interesting findings. These three figures show that the difference in precision performance between the People-Search algorithm

and other algorithms was bigger in the Computers domain than that in the Arts or Sports domain. One reason for this is that the personal websites in the Computers domain have more link information. Table 5.15 shows the average numbers of inlink and outlinks each website or main page has in the three domains. Table 5.15 shows that the average number of inlinks or outlink a personal website or the main page in the Computers domain has is much greater than that in the Arts or Sports domain. For example, the average number of inlinks for a personal website in the Computers domain is 50.3. In contrast, that number is 19.5 in the Arts domain. For outlinks, that number is 69.2 and 35.2, respectively. This fact also explains why the performance of People-Search and Site_Content_Inlink algorithms was so close to each other in the Arts domain.

The results from the three individual domains show that the People-Search algorithm is the best one not only in domain-independent (cross domain) dataset but also in domain-dependent dataset.

5.3.2.3 The Effectiveness of the People-Search Algorithm on Ranking Returned Search Results. To test People Search algorithm's effectiveness on ranking the returned results, the top 50 returned hits of each query were divided into five groups. Precision was calculated for each group, and then they were compared to each other. If group 1's precision was higher than group 2's, and group2's was higher than group3's, and so on, then this algorithms was effective on ranking the search results.

Table 5.16 shows the experimental results of this test, which is also shown in Figure 5.7. The results are based on the same 2,000 queries used in testing the cross-domain performance of the 14 algorithms. The results show that group 1's precision was better than group 2's, group 2's was greater than group 3's and so on. Paired t-test was

conducted to test if the difference between precisions of two groups was statistically significant. The results show that their differences were statistically significant at the level of $p < 0.01$. These results illustrate that the People-Search algorithm was effective in ranking the search results.

Table 5.16 Results of the People-Search Algorithm's Ranking Effectiveness

Group	Group 1 (Top 1 to 10)	Group 2 (Top 11 to 20)	Group 3 (Top 21 to 30)	Group 4 (Top 31 to 40)	Group 5 (Top 41 to 50)
Average precisio n	0.732	0.610	0.545	0.513	0.490
Std	0.341	0.344	0.310	0.302	0.307

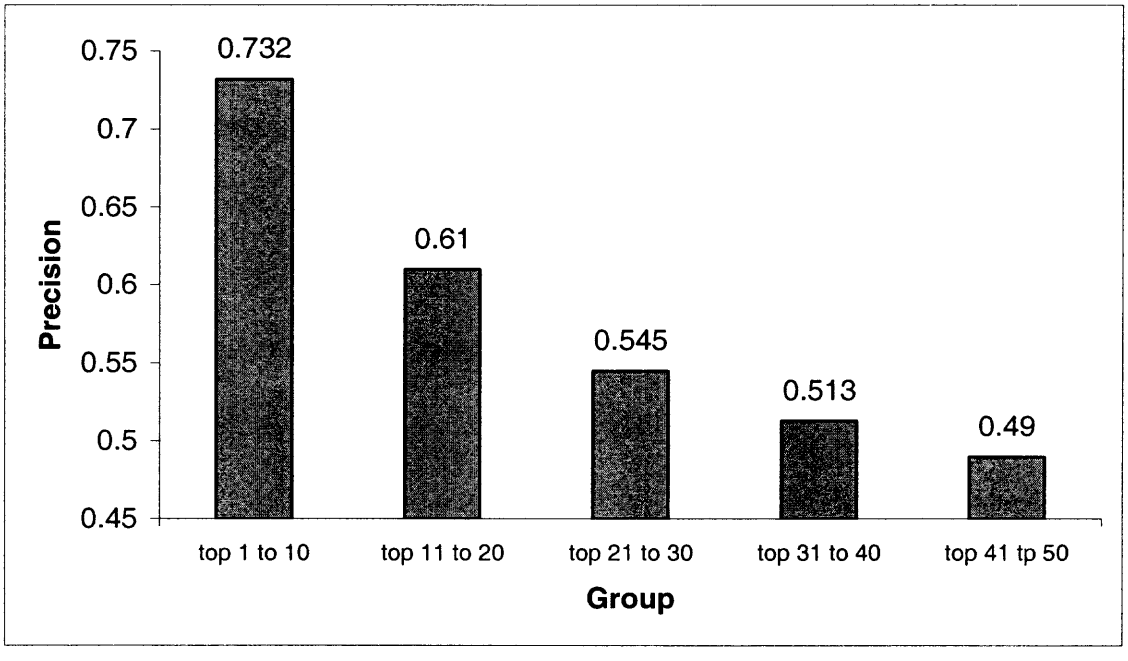

Figure 5.7 Precision for the five groups of search results.

5.3.3 Experimental Results Using Kruskal-Goodman Γ Measure

5.3.3.1 Results for Queries from All ODP Categories. As discussed in Chapter 4, in the context of our experimental dataset, Γ measure is more suitable for comparing the 14 algorithms than the traditional IR measures. Γ measure considers all the returned websites when comparing two orderings (one is produced by one of the 14 algorithms, and the other one is the ground truth ordering, which is induced from the ODP structure), rather than just the top N returned results. The higher a system's Γ value is, the better the system's performance is. Having a higher Γ value means the ordering of the returned websites produced by this algorithm is closer to the ground truth ordering of the ODP hierarchy. As what was done in last section, the 14 algorithms are classified into two groups: one includes the algorithms using information from an entire website, and the other one includes algorithms using information from only the main page. To better understand how each algorithm performed in its group, the Γ values are reported in two tables. Table 5.17 presents Γ values for algorithms in group 1, and Table 5.18 presents Γ values for algorithms in group 2. These results were obtained based on the 2,000 queries using micro-averaging. For each algorithm, Γ value was calculated for each query first, and then the average Γ value for all these 2,000 queries was calculated. The queries were the same as the ones used in the evaluation using the traditional IR measures.

Table 5.17 and 5.18 shows that, for both groups, the algorithm integrating both the content and link information performed the best, and the algorithm using only outlink information preformed the worst. All the 14 algorithms are compared together in Figure 5.8 and also in Table 5.19. In Figure 5.8, the algorithms belonging to group 1 are shown on the left side, and algorithms in group 2 are shown on the right side. In Table 5.19,

according to their Γ values, all these algorithms are ranked in the descending order of their performance from top to bottom. Figure 5.8 and Table 5.19 show that the People-Search algorithm performed the best, followed by Site_Content_Inlink, and the MainPage_Outlink algorithm performed the worst. This result conforms to the result from the evaluation using precision, recall and F measure. Section 5.3.2 has explained why MainPage_Outlink was the worst.

Table 5.17 Γ Values for Algorithms Using Information from an Entire Website

Algorithm	Average Γ value	Std
Site_Content_Link (People-Search)	0.589	0.272
Site_Content_Inlink	0.576	0.264
Site_Content_Outlink	0.548	0.269
Site_Content	0.531	0.241
Site_Link	0.380	0.344
Site_Inlink	0.297	0.312
Site_Outlink	0.262	0.338

Table 5.18 Γ Values for Algorithms Using Information from only the Main Page

Algorithm	Average Γ value	Std
MainPage_Content_Link	0.512	0.264
MainPage_Content_Inlink	0.508	0.261
MainPage_Content_Outlink	0.481	0.256
MainPage_Content	0.477	0.233
MainPage_Link	0.301	0.322
MainPage_Inlink	0.289	0.305
MainPage_Outlink	0.082	0.138

Paired t-test was used to test the significance of the Γ results. The test results are listed in Table 5.19. The results show that, for an algorithm listed in the table, all the algorithms listed above it performed better than it, and it performed better than all the algorithms listed below it, at the significance level of $p < 0.01$, except the following two cases: when comparing MainPage_Content_Outlink and MainPage_Content, and when comparing MainPage_Link and Site_Inlink. MainPage_Content_Outlink outperformed MainPage_Content at $p < 0.05$ level. This conforms to the results obtained by using the traditional IR measures, reported in Section 5.3.2. The performance difference between MainPage_Link and Site_Inlinkt was not statistically significant; its p value was greater than 0.05.

As mentioned in Chapter 4, the agreement percentage between two ordering is: $Pr = ((\Gamma + 1) / 2)$. For the People-Search algorithm, $Pr= (0.589+1)/2 =80\%$. This means the ordering produced by the People-Search algorithm agrees with the ground truth ordering on 80% of the website pairs.

This cross-domain test shows that the People-Search algorithm was the best one among all these 14 algorithms.

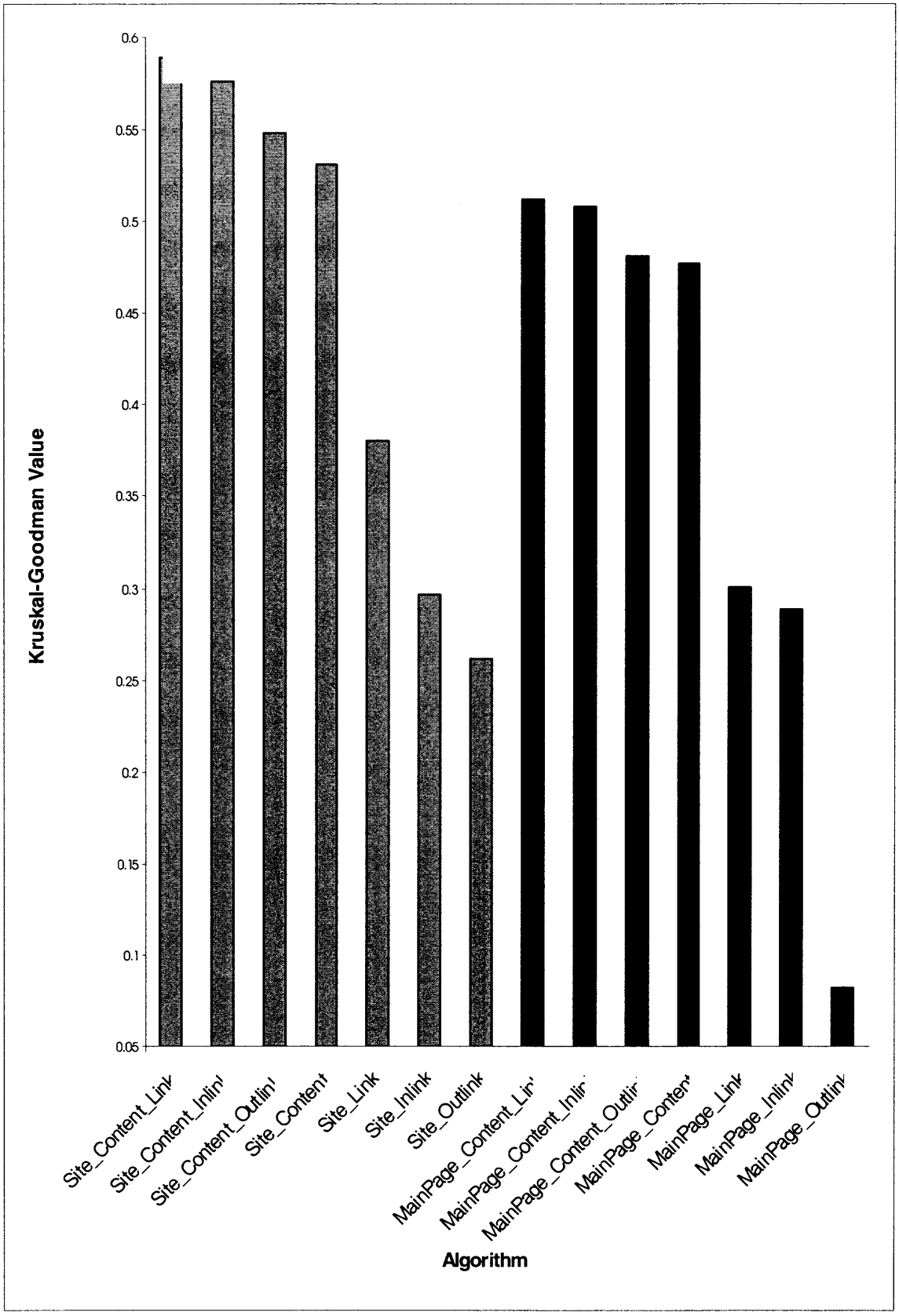

Figure 5.8 Γ values of the 14 algorithms.

Table 5.19 Ranks of the 14 Algorithms

Algorithm	Average Γ value	Significance test on the mean difference between an algorithm and the one listed just below it
Site_Content_Link (People-Search)	0.589	$p < 0.01$
Site_Content_Inlink	0.576	$p < 0.01$
Site_Content_Outlink	0.548	$p < 0.01$
Site_Content	0.531	$p < 0.01$
MainPage_Content_Link	0.512	$p < 0.01$
MainPage_Content_Inlink	0.508	$p < 0.01$
MainPage_Content_Outlink	0.481	$p < 0.05$
MainPage_Content	0.477	$p < 0.01$
Site_Link	0.380	$p < 0.01$
MainPage_Link	0.301	$p > 0.05$
Site_Inlink	0.297	$p < 0.01$
MainPage_Inlink	0.289	$p < 0.01$
Site_Outlink	0.262	$p < 0.01$
MainPage_Outlink	0.082	N/A

5.3.3.2 Results for Queries from Three Individual Categories. Based on the cross-domain test results obtained by using Γ measure, the top five algorithms are the People-Search algorithm, Site_Content_Inlink, Site_Content_Outlink, Site_Content, and MainPage_Content_Link. The three domains were the same ones used in the test using the traditional IR measures, which were Arts, Sports and Computers. Five hundred websites were randomly selected from each of the three domains as queries. They were the same ones used in evaluating the top five algorithms using the traditional IR measures, described in Section 5.3.2.2.

Table 5.20 Results of Γ Measure for Arts, Sports and Computers Domains

Algorithm	Domain					
	Arts		Sports		Computers	
	Γ measure	Std	Γ measure	Std	Γ measure	Std
Site_Content_Link	0.590	0.233	0.532	0.205	0.593	0.258
Site_Content_Inlink	0.585	0.229	0.522	0.197	0.576	0.237
Site_Content_Outlink	0.575	0.218	0.511	0.199	0.558	0.236
Site_Content	0.571	0.210	0.497	0.187	0.533	0.201
MainPage_Content_Link	0.507	0.234	0.483	0.217	0.527	0.244

Table 5.20 shows the Γ measure results for the three domains. The results for each domain are also shown in Figure 5.9, 5.10 and 5.11, respectively. The algorithms are ordered from top to bottom in descending order of their Γ values. Paired t-tests were conducted to test the significance of the results. The results show that the People-Search algorithm outperformed all other four algorithms in all the three domains, and the results were statistically significant. The results also show that for any given algorithm in Table 5.20, the algorithms listed above it performed better than it, and it performed better than all the algorithms listed below it. This was true for all three domains. The results were statistically significant at $p < 0.01$ level except the following: for the Arts domain, Site_Content_Outlink was better than Site_Content, but the difference was not statistically significant. This conforms to the results obtained using the traditional IR measures, reported in Section 5.3.2.2. Section 5.3.2 has explained why the difference between these two algorithms was not big in the Arts domain – the personal websites in the Arts domain do not have much outlink information, compared to the websites in the

Sports or Computers domain. Personal websites in the Arts domain do not have much inlink information, either (see Table 5.15, in Section 5.3.2). This is why the Γ values of the four algorithms, which use the same website content information, were close to each other, and far better than the MainPage_Content_Link algorithm (see Figure 5.9). In contrast, personal websites in the Computers domain have relatively richer inlink and outlink information to exploit. Therefore, in the Computers domain, the differences between the top four algorithms (People-Search, Site_Content_Inlink, Site_Content_Outlink and Site_Content) were relatively larger, compared to that in the Arts or Sports domain. The reason is that the only difference between these four algorithms is includiing link (inlink, outlink or both) information or not, and the Computers domain has richer link information. This observation conforms to that from the results reported in Section 5.3.2.2, which were obtained by using traditional IR measures.

Based on the Γ measure results from the three individual domains, the conclusion is that the People-Search algorithm is the best not only in cross-domain dataset but also in each individual domain. The same conclusion was obtained from the results based on precision, recall and F measure.

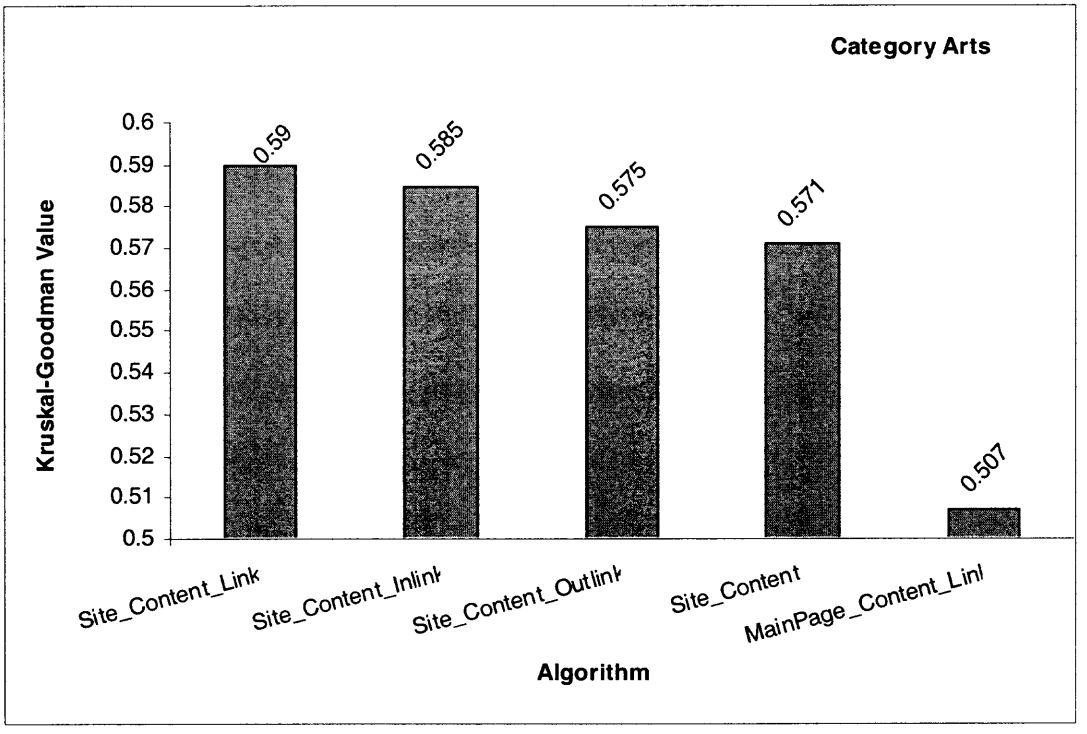

Figure 5.9 Γ measure results for the Arts domain.

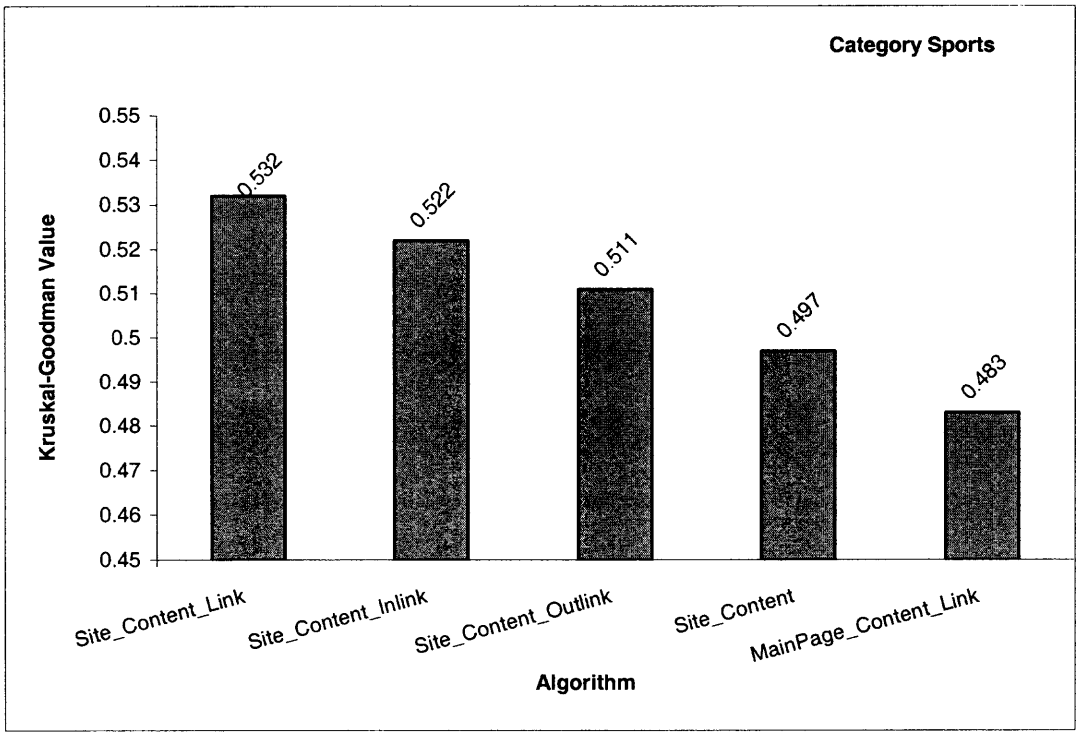

Figure 5.10 Γ measure results for the Sports domain.

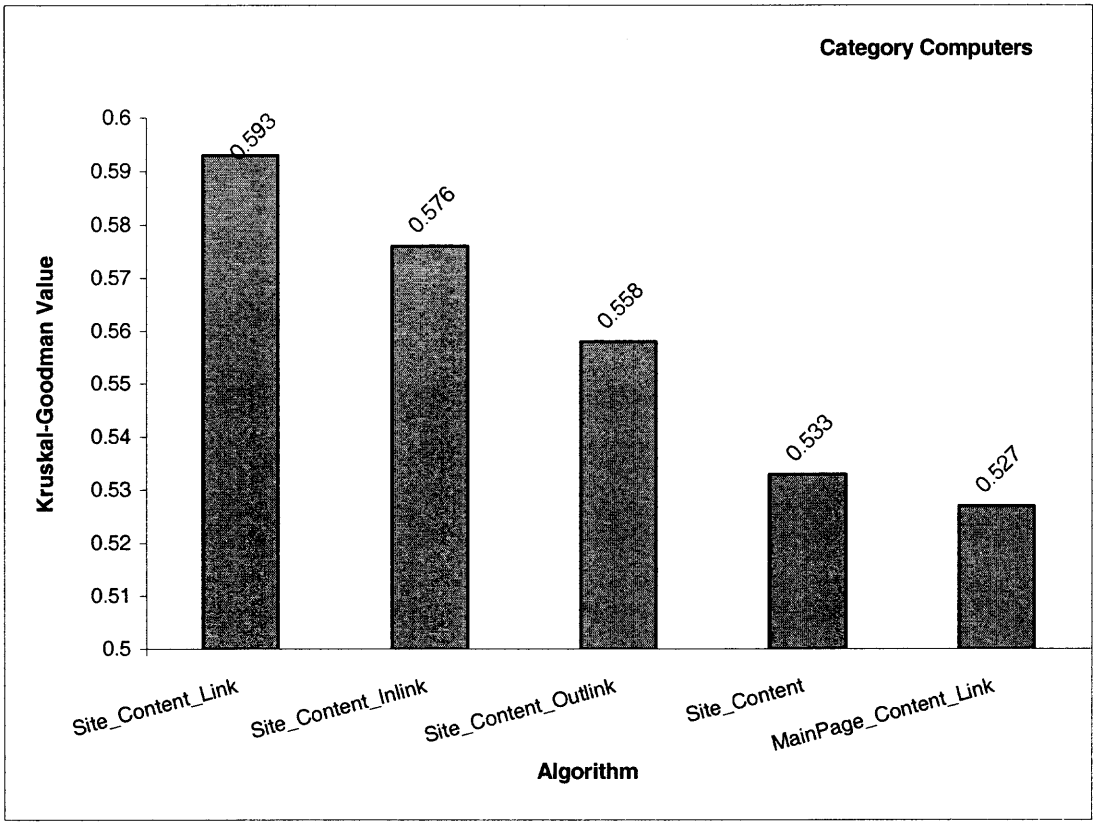

Figure 5.11 Γ measure results for the Computers domain.

This section reports the automatic evaluation results based on two kinds of measures, the traditional IR measures and the Γ measure. Results show that People-Search algorithm outperformed all the other algorithms.

5.4 Human Evaluation Results and Analysis

5.4.1 Demographic Background of the Subjects

The demographic information of subjects was collected from the pre-evaluation questionnaire (see Appendix C). The questions were designed to investigate subjects' background information regarding their experiences with Internet and search engines. Initially, forty three subjects were invited to participate in this experiment, and finally 40

of them completely finished the experiment. Only the results from these 40 subjects are included in the data analysis. The subject background information is presented in Table 5.21 and 5.22. Table 5.21 shows results for questions whose answers can be categorized into different categories. Table 5.22 shows results for questions that can be answered by giving a numeric value. In Table 5.21, the answers for each question are not exclusive, so the sum of percentages of answers for a question may be larger than 100%. It is noteworthy that the contents in the "Others" category were provided by subjects.

Table 5.21 shows that the majority of the participants were in IT related areas, and their purposes of using search engines varied, from searching information for work/study to entertainments, news and shopping. An encouraging result is that 95% of the subjects considered searching people online was one of their purposes of using search engines. Eighty percent of the participants thought one purpose of searching people online was to find people in their interest areas. Regarding the methods used for searching people online, all of them had used search engines, and about 40% of them had used other approaches, such as online community and paper citations. One subject also specified "from people's website" as one of the methods used. When being asked to clarify, this subject said sometimes, from the links in a person's website he was browsing, he could find other people similar to the current one.

Table 5.22 shows some information about subjects' experiences of using Internet and search engines, and the time spent on them. On average, a subject had about six years of experience using search engines and about four hours were spent on Internet each day. Based on the 7-point Likert scale, with 1 meaning novice and 7 meaning expert, the average experience of using search engines was 5.6, which means subjects considered

them quite skillful in using search engines. This is not a surprising result, since most of the participants were in IT areas.

Table 5.21 Subjects' Demographic Information

Subject Characteristics	Category	Number of Responses	Percentage
Work/study area	IT related (CS, IS, etc.)	35	87.5%
	Chemistry, Business, Bioinformatics	5	12.5%
Purposes of using search engines	My research or my work	36	90%
	My study	38	95%
	Entertainment (search music, movie, etc.)	37	92.5%
	News	32	80%
	Knowledge acquisition (history, politics, etc.)	40	100%
	Search people	38	95%
	Shopping	12	30%
	Others - images, English grammar, for fun, etc.	7	17.5%
Purposes of searching people online	Find other people in my interest areas.	32	80%
	Find experts in certain areas.	17	42.5%
	Find a person I am interested in.	22	55%
	Search celebrities	15	37.5%
	Others		
Approaches to searching people	Use search engines	40	100%
	Use online community	18	45%
	Use online directory (e.g., yahoo directory)	17	42.5%
	From paper citations	15	37.5%
	Others - from people's websites	1	2.5%

Table 5.22 Information about Subjects' Experiences of Using Internet and Search Engines

Characteristics	Average	Std
Years of experience using computer	11.5	3.2
Years of experience using Internet	7.8	1.4
Years of experience using search engine	5.9	1.4
Hours per day spent on Internet	3.8	1.6
Experience on using search engine (novice 1 – 7 expert)	5.6	1.3

5.4.2 Queries and Related Statistics

For the human evaluation, 14 queries were chosen from the dataset and presented to the subjects. Each subject was required to choose four, which they would feel comfortable with, from the 14 queries for the evaluation. If they would prefer to use their own queries, they might send their queries to the experiment coordinator to preprocess them. In this experiment, all subjects have used the given queries to do the experiment. One query might be executed multiple times by different subjects. There were 40 subjects, and each subject was asked to execute four queries. Therefore, the total number of executed queries should be 160. Actually, because one subject executed six queries, the final number of executed queries was 162. Table 5.23 shows some information for each query: the topic area, the number of retuned websites and the number of times being executed by the subjects.

Table 5.24 shows some other statistics related to the queries. The table shows that the total number of times the 14 queries were executed was 162. On average, the number

of times a query was executed was 11.6. As described in Chapter 4, in this evaluation, for each query, the top 20 returned hits for each of the three algorithms (People-Search algorithm, Site_Content, and MainPage_Content_Link) were evaluated. For each query, these 60 returned hits were mixed together and presented to the subjects. Due to the overlaps between the returned hits, the final number of returned hits for each query was less than 60. The average number of returned websites per query presented to subjects was 30.6. To judge whether a returned personal website was relevant or not, the subjects could open the returned site to check its content, or make a judgment just based on the metadata provided with the hit, such as the snippet. Table 5.24 shows that on average for each query 18 returned sites were actually clicked by the subjects, which was about 60% of all the returned hits. For all other returned hits, they judged their relevance by examining their metadata. The metadata of a returned hit included title, description (snippet) and URL. The title and description were extracted from ODP directory and they were provided by the owners of the websites. Therefore, they are a good indicator of the website content.

Table 5.25 shows information about the amount of time spent on different steps of the experiment. The average time spent on browsing a query site was about 1.83 minutes, and the average time spent on reading an opened returned website was 0.88 minute. On average, a subject spent about 75 minutes to finish the experiment. This number did not include the time spent on reading the experimental instructions. How these numbers were calculated is briefly described below. The time spent on each clicked website is the difference between the time of opening the site and the time of rating this site. The time spent on browsing a query is the time difference between when the query site was opened

and when the subject gave the first rating to or first clicking on any of the returned hits. It is not known if the subjects did anything else during the experiment, so the time reported might include the time spent on other things during the experiment, such as answering phone calls. This is obviously one limitation that affects the accuracy of the time data.

Table 5.23 Query Information

Query No.	Query topic	Number of returned hits	Number of times executed by subjects
1	Physics	25	8
2	biology, brain, nerve system	29	8
3	cigar, smoking	22	16
4	computer programming	33	12
5	computer science, java	33	9
6	arts, painting	36	13
7	planet, solar system	37	15
8	bridge card	25	8
9	home camping	34	16
10	database, programming	34	15
11	Basketball	32	20
12	health, fitness	32	5
13	algorithm, computation complexity	26	5
14	star, planet	30	12

Table 5.24 Query Related Statistics

Total number of unique queries	Average number of times a query was executed $\pm$ std	Average number of returned websites per query $\pm$ std	Average number of returned websites opened by subjects per query $\pm$ std	Percentage of returned sites opened by subjects
14	11.6 ± 4.5	30.6 ± 5.0	18.1 ± 5.1	60%

Table 5.25 Time Spent on the Experiment (minutes)

Type	Time on browsing all opened returned websites of a query	Time on browsing one opened search hit	Time on browsing a query site	Average time on the evaluation per subject
Average	15.90	0.88	1.83	74.50
Std	9.73	0.52	0.94	26.41

5.4.3 Subject Confidence on Understanding Queries and Returned Results

To ensure the reliability of the experimental results, the analysis and the conclusion, participants' confidence levels on understanding the query sites and the returned sites were also collected. Subject confidence ratings were collected based on the 7-point Likert scale. If a query received a confidence score less than 4, then this query and all its search results were not included in the final data analysis. If a search result received a confidence score less than 4, it would also be excluded from the final data analysis.

Table 5.26 shows the results about subject confidence on queries and search results. In this experiment, 162 queries were executed (14 unique queries were executed 162 times). The total number of returned hits for all these 162 queries was 5,034. Four queries received a confidence value less than 4. The number of hits for these four queries was 109. These four queries and their returned hits were excluded from the final data analysis. After removing unqualified data, the total number of returned hits for the 158 queries was 4,925. Among the 4,925 returned hits, 288 of them received a confidence level less than 4. After excluding these 288 hits, overall, 4,637 returned websites were

included in the final data analysis. The average confidence score for these 4,637 hits was 5.95.

Table 5.26 Subject Confidence Information on Understanding Queries and Search Results

Total number of queries executed	Number of queries with received confidence >= 4	Average confidence value for the 158 queries ± Std	Total number of returned hits for the 158 queries	Total number of returned hits with confidence < 4	Total number of returned hits included in final data analysis	Average confidence ± Std (for all returned hits included in the final data analysis)
162	158	6.17 ± 0.73	4925	288	4637	5.95 ± 0.83

5.4.4 Inter-subject Agreement

Kendall Coefficient of Concordance (W) is a measure of agreement between ratings given by human subjects (Siegel and Castellan, 1988). It was used in this test to see if there was agreement between subjects on rating the search results. Based on the W value and the related significance test, it could be determined whether or not there was a significant agreement between the participants and whether or not the hypothesis that the agreement was just observed by chance could be rejected. W value ranges from 0 (complete disagreement) to 1 (complete agreement). A high value of W is interpreted to mean that the subjects applied the same overall standard on rating the observations under study -- in the case of this study, rating the search results.

Table 5.27 shows the W value for each query and also the average W value for all queries. The average W value was 0.62, which implies a high agreement among the subjects. This can be interpreted as that the subjects agreed on 81% ((1+0.62)/2=0.81) of rankings, which was a strong agreement. Z test, a kind of significance test, was used to

test the significance of the W results. Based on the Z test results, the W values shown in Table 5.27 were all statistically significant at level of $p < 0.01$. Therefore, the hypothesis that the high inter-subject agreement on the ratings occurred merely by chance could be rejected, and there was a strong and significant level of agreement among the subjects when rating the search results.

Table 5.27 Inter-subject Agreement

Measure	Query No.														Average	Std
	1	2	3	4	5	6	7	8	9	10	11	12	13	14		
W	0.55	0.37	0.78	0.52	0.46	0.75	0.79	0.7	0.56	0.48	0.84	0.46	0.57	0.79	0.62	0.16

5.4.5 Comparison of the Three Algorithms

In the human evaluation, the three algorithms being evaluated by subjects were the People-Search algorithm, Site_Content and MainPage_Content_Link. Section 4.3 has explained why these three algorithms were chosen, instead of others. Table 5.28 presents the subject evaluation results of the three algorithms. As mentioned before, when presenting search results to subjects, the search results from the three algorithms were mixed together to avoid bias. After all the data were collected, for each query, the original order of the search results was restored. Then for each algorithm, the average score for the top 5, top 10, top 15 and top 20 returned search results was calculated. The results are also shown in Figure 5.12. The results show that the People-Search algorithm outperformed the other two algorithms: its score was higher than the others for any of the

four N values. Paired t-test was conducted to test the significance of the results. Overall, the 14 queries were executed 158 times, so the size for the paired t-test was 158. Based on the t-test results, the People-Search algorithm was better than the other two algorithms at the significance level of $p < 0.01$. This conforms to the results of the automatic evaluation. In the automatic evaluation, precision, recall and F were calculated when N was 10, 20, 30, 40 and 50. Due to the limitation of human evaluation, subjects only evaluated the top 20 search results for each algorithm. Figure 5.12 also shows that MainPage_Content_Link performed better than or close to Site_Content algorithm when N was small. But when N was larger, Site_Content performed better than MainPage_Content_Link. This also conforms to the findings in the automatic evaluation (see Figure 5.3). Section 5.3.2.1 has explained the reason.

Table 5.28 Human Evaluation Results of the Three Algorithms

Algorithm	Measure	N=5	N=10	N=15	N=20
People-Search Algorithm (Site_Content_Link)	Mean	6.38	6.22	5.98	5.82
	Std	1.24	1.25	1.46	1.56
Site_Content	Mean	6.18	6.01	5.81	5.65
	Std	1.29	1.30	1.55	1.62
MainPage_Content_Link	Mean	6.20	5.97	5.62	5.49
	Std	1.53	1.65	1.77	1.94

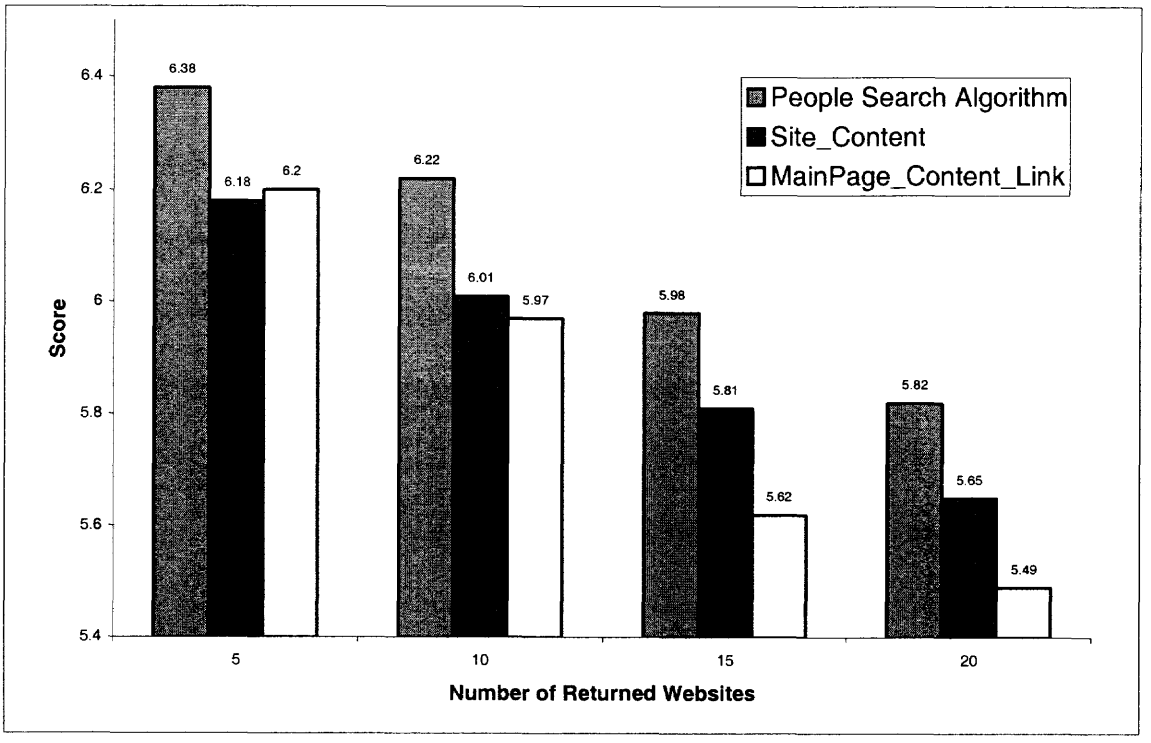

Figure 5.12 Comparisons of human evaluation results of the three algorithms.

5.4.6 Correlations between the Results of Human Evaluation and Automatic Evaluation

The correlations were calculated based on the results of the 14 queries. Given N = 10 or 20, for each of the three algorithms, the precisions of the 14 queries obtained in the automatic evaluation were compared to the human ratings to find the relationship between these two variables (two sets of results).

Pearson's correlation coefficient r was used as the correlation measure. Pearson r's value is between –1 and +1. A value near the upper limit, 1, indicates a strong positive relationship, while an r close to the lower limit, -1, suggests a strong negative relationship. A value near 0 means there is no or very weak relationship between the two variables. Usually, a value between –0.5 and +0.5 should be considered as a weak relationship (Devore and Peck, 1997). Table 5.29 shows the correlation results for the

three algorithms. T test was used to test the significance of these correlations. The t test results are also shown in this table. All of the correlations shown in this table were larger than 0.5, in other words, they all show a strong relationship. When N =10, for any of the three algorithms, the correlation between the two kinds of evaluation results was significant at the level of $p < 0.05$. When N =20, the correlations were significant at level of $p < 0.01$.

Table 5.29 Correlations between Results of Human Evaluation and Automatic Evaluation

Algorithm	N=10		N=20	
	R	p value	R	p value
People-Search Algorithm	0.69	<0.05	0.86	<0.01
Site_Content	0.66	<0.05	0.75	<0.01
MainPage_content_Link	0.52	<0.05	0.69	<0.01

5.4.7 People-Search Algorithm's Effectiveness on Ranking the Search Results

In the human evaluation, the People-Search algorithm's effectiveness on ranking the search results was also measured. Similar to the method used in the automatic evaluation, the retuned results were also divided into groups. In this experiment, the top 20 results were split into four groups. Their average ratings were compared to each other.

Table 5.30 shows the experimental results for this test, which is also shown in Figure 5.13. The results are based on the 158 executed queries. The results show that group 1's precision was better than group 2's, group 2's was greater than group 3's and so on. Paired t-test was conducted to test if the difference between any two groups was

statistically significant. The result shows that their differences were statistically significant at the level of $p < 0.01$. These results show that the People-Search algorithm was effective on ranking the search results. The results conform to the results obtained by using the precision measure in the automatic evaluation. Both tests show that the People-Search algorithm was effective in ranking search results.

Table 5.30 People-Search Algorithm's Ranking Effectiveness on Search Results

Group	Group 1 (Top 1 to 5)	Group 2 (Top 6 to 10)	Group 3 (Top 11 to 15)	Group 4 (Top 16 to 20)
Average score	6.38	5.99	5.46	5.21
Std	1.21	1.28	1.73	1.86

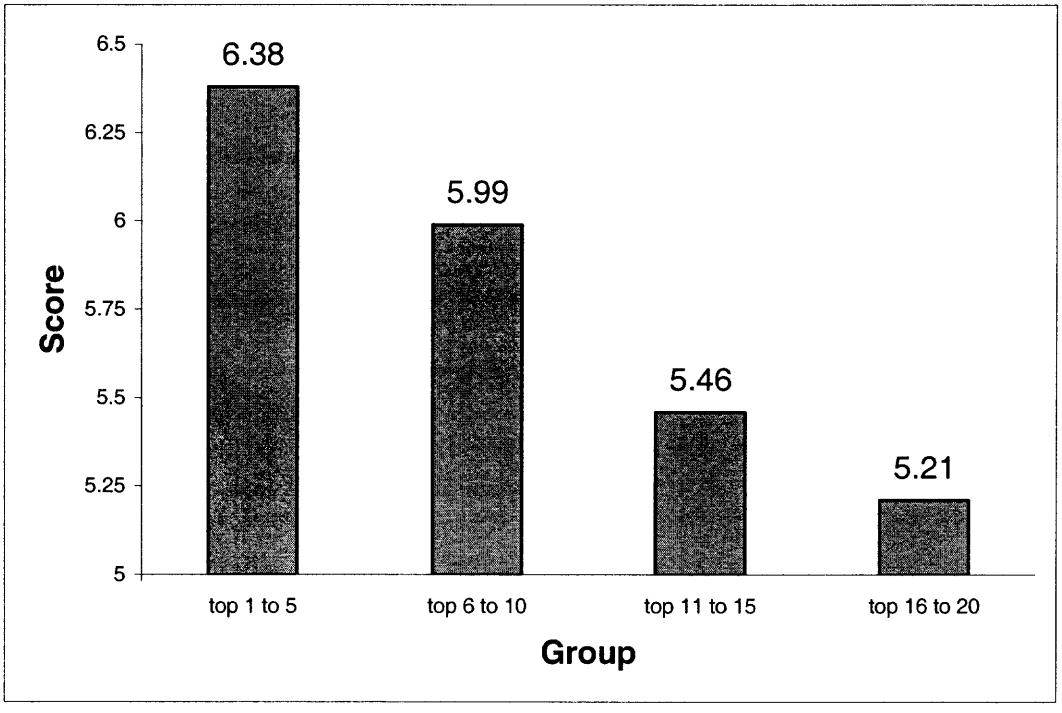

Figure 5.13 Ranking effectiveness of People-Search algorithm.

5.4.8 Post-Questionnaire Results

After completing the experiment, the subjects were asked to answer a short post-questionnaire (see Appendix D for the questionnaire). Table 5.31 presents the results of the closed questions, which were based on the 7-point Likert scale. The first closed question asked if the subjects would like to use a people search systems similar to the one used in this experiment. The average score for this question was 5.7, which is very positive. About 92% of the responses were greater than or equal to 4, the mid point of the 7-point scale measure. The second question asked the subjects if they preferred the method used in this experiment over other ones in searching similar people online. The average response score for this question was 4.8, greater than the mid point, but not as positive as the responses to the first question. Fifteen percent of the responses were negative. Eighty five percent of the responses were greater than or equal to 4. 50% (20 out of 40) of the responses were greater than 4. 35% of the responses were 4, which is a relative large portion. This shows that many subjects might not be certain about their preference. One possible reason might be the lack of direct comparisons between the people search approach proposed in this study and other methods. A direct comparison between this method and other methods is difficult, because other approaches are not specialized for the people search task and their dataset formats are different from the one used in this study. Therefore, a direct comparison with other approaches might not be fair or appropriate, but this might be a future research topic.

The post-questionnaire also includes an open-ended question asking subjects' opinions about this people search method and this experiment. Some of the responses are listed below, followed by our comments.

Table 5.31 Post-questionnaire Result

Question	Response (Strongly disagree 1 - 7 Strongly disagree)			
	Average	Std	Responses >= 4	
			Number	Percentage among all responses
I would like to use a people search system similar to this one in the future	5.7	1.1	37	92.5%
To search similar people on the Web, I prefer the method used in this experiment over other ones.	4.8	1.2	34	85%

Subject Comment 1:

Some of the links provided did not open. I don't know it is temporally unavailable or they have been moved. I marked those as 1 (irrelevant) else the program would not let me complete the submission.

In the experimental prototype system, when a returned website is clicked, the system will open the real personal website (not the mirror one stored in the system). It is possible that this website might not be available due to some reasons, such as network traffic or temporary unavailability of its server.

Subject Comment 2:

I was not sure what you mean by "relevant". So I kind of changed my standards of "relevancy" while I was rating more of the retrieved web pages.

Though it has been explained in the experimental introduction and instructions, subjects would still have their own interpretation of what "relevant" means. Given the same query, a query like a personal website in this study or just a regular query, different users may have different understandings about the query, and about what kinds of returned

results are "relevant." This is a common problem for all IR experiments involving human evaluation, not just for this study.

Subject Comment 3:

> *Some websites have many pages. They are talking about different things, but I may be interested in only one of them. If the system can also let me search other people based on a single page, that'll be great.*

This is true. Some personal websites are homogenous – all the pages are talking about one main topic. Some other sites may contain heterogeneous pages about totally different topics, such as computers and fishing. When using a whole website as the query, if this site is heterogeneous, the search results may also be heterogeneous. Usually a single web page is homogenous, meaning its content is mainly about one main theme. Search-by-page would allow users to specify more specific needs through the query page, and the search results would also be more relevant. Actually, this will be one of the future research topics.

Subject Comment 4:

> *Generally speaking, it is an amazing system. It would have helped me a lot if I had such a system to use when applying for graduate schools. I think it would be very helpful for my research, because I can easily find those people with same interests*

This is encouraging.

5.5 Summary

This chapter has presented the experimental results and data analysis. The characteristics of the experimental dataset were discussed first. Then the optimal values of the algorithm parameters were reported. Next the results and analysis of the automatic evaluation were presented. Finally, results from the human evaluation were reported. The results and analysis show that, based on the proposed person representation method, the People-Search algorithm outperformed others in finding similar people from the Web, and it is effective on ranking the returned search results. The results of the two kinds of evaluations answered the research question 3 and 4.

CHAPTER 6

SUMMARY, LIMITATIONS AND FUTURE RESEARCH

This chapter summarizes the major findings of this study, discusses the limitations of this study, outlines the main contributions, and presents possible future research directions.

6.1 Summary

6.1.1 Research Goals and Research Questions

This study tries to provide a solution for people search: specifying characteristics of a person and finding other persons who share the similar characteristics with the given person. It aims at finding a people search solution that requires no manual involvement for building searchable people profiles and is able to search people from all available domains on the Web. Two major research issues in this study are: how to represent a person and how to match persons. The first problem is how to profile users - what type of information does a system use to represent its users, and how does it acquire this information? The second problem is how to compute matches - what is the system's model of a good match? and how does the system compute matches? In short, this study, has answered the following main research questions:

1. On the Web, how to represent a person? This representation method should reflect this person's characteristics, and can be used for the process of matching persons.

2. Given the person representation method, how to find similar people from the Web for a given person? What kinds of methods/algorithms can we design?

3. Among the possible methods/algorithms, which one performs the best?

4. How effective is the best algorithm on ranking the returned search results?

6.1.2 People Search Framework

To solve the people search problem, this study first defines a framework to specify what kind of information may be used in person representation and the process of matching persons. The following attributes together define the proposed people search framework:

- A person's personal website can be used to represent this person, in terms of his/her interests and background.
- If persons can be represented by their own websites, then the search query can be represented by a personal website as well. Therefore, in the people search process, a query is a personal website, and the returned results are a list of personal websites.
- All web pages belonging to a person's website may be used to compare two persons.
- Both the content and the link information of the web pages of a person's website could be used for representing this person.

This framework defines the way of representing a person on the Web, and acts as the guidelines for designing algorithms for people search. This framework answered research question 1.

6.1.3 Algorithms

Under the proposed framework and person representation method, a main algorithm was proposed and 13 other algorithms were also explored and compared to the main algorithm. The main algorithm, called People-Search algorithm, integrated content and link information of all the pages belonging to a personal website to represent the person and match persons. The other 13 algorithms were based on different combinations of content, inlink, and outlink information of an entire personal website or only the main page. It was hypothesized that the People-Search algorithm would outperform the other

13 algorithms. These 14 algorithms and the types of information they use are listed in Table 6.1. These designed algorithms answered research question 2.

Table 6.1 The 14 Algorithms and the Information Used in the Similarity Calculation

Algorithm Name	Information used	Information source	
		From the entire website	From only the main page
Site_Content_Link (The People-Search algorithm)	Content & link (both inlink & outlink)	√	
Site_Content_Inlink	Content and inlink	√	
Site_Content_Outlink	Content and outlink	√	
Site_Content	Content only	√	
Site_Link (Inlink &outlink)	Link only (inlink & outlink)	√	
Site_Inlink	Inlink only	√	
Site_Outlink	Outlink only	√	
MainPage_Content_Link	Content & link (both inlink & outlink)		√
MainPage_Content_Inlink	Content and inlink		√
MainPage_Content_Outlink	Content and outlink		√
MainPage_Content	Content only		√
MainPage_Link (Inlink& Outlink)	Link only (inlink & outlink)		√
MainPage_Inlink	Inlink only		√
MainPage_Outlink	Outlink only		√

6.1.4 Evaluation Results and Findings

Both automatic evaluation and human evaluation were conducted to test the performance of the 14 algorithms. The main experimental tasks, the results and findings are listed in Table 6.2. These evaluation results answered research question 3 and 4.

Table 6.2 Evaluations and Results

Evaluation method	Task	Measure	Results & findings
Automatic evaluation	Compare the three kinds of link similarity measures: cosine, Euclidean distance and Jaccard coefficient	Γ measure	Cosine similarity measure was the best.
	Compare the 14 algorithms' cross-domain performance	Precision, recall and F	The People-Search algorithm outperformed all other algorithms; in finding website similarity, content information was better than link information, and inlink information was better than outlink information
	Compare the top five algorithms in three individual domains	Precision, recall and F	The People-Search algorithm was the best in all of the three chosen domains: Arts, Sports and Computers.
	Evaluate People-Search algorithm's effectiveness in ranking search results	Precision	The People-Search algorithm was effective in ranking search results
	Compare the 14 algorithms' cross-domain performance	Γ measure	The People-Search algorithm outperformed all other algorithms; in finding website similarity, content information was better than link information, and inlink was better than outlink information.
	Compare the top five algorithms in three individual domains	Γ measure	The People-Search algorithm was the best in all of the three domains
Human evaluation	Compare the People-Search algorithm with two other important ones: the one using only site content information and the one using content & link information from only the main page	Human ratings	The People-Search algorithm outperformed the other two algorithms.
	People-Search algorithm's effectiveness in ranking returned search results	Human ratings	The People-Search algorithm was effective in ranking search results

6.1.5 Implications of this Study

This study attempts to find a general purpose people search solution, which means it is intended to be used in all domains or subject areas. In both the automatic evaluation and the human evaluation, the People-Search algorithm performed the best. The results were statistically significant. The experimental results show that, using the proposed person representation method, to find similar people to a given one from the Web, the content information and link information of all web pages of a person's website should be integrated together.

The experimental results show that the content information was better than the link information, and the inlink information was better than the outlink information in calculating the similarity between two personal websites. The results were statistically significant. These conclusions might help other studies which need to exploit different types of information in website similarity calculation.

In this study, a new link weighting method is also presented, which is called LF.IWF and is adapted from the TF.IDF method. Although TF.IDF is a popular term weighting method in content similarity calculation, this study is the first to adapt and apply it in link weighting calculation. This work is also the first study to apply cosine similarity measure in the link similarity calculation. One main difference between this method and other link similarity methods, such as the Jaccard coefficient, is that this method considers a link's weight, instead of just presence and absence. In the experiment, the cosine measure with LF.IWF link weighting method was compared to two other measures, the Jaccard coefficient and the Euclidean distance measure. Results show that the cosine similarity measure significantly outperformed the other two measures in link

similarity calculation. The new link similarity measure and link weighting method not only can be used in comparing two personal websites, but, potentially, they may also be used in calculating link similarity in other web search areas, such as link similarity between two general web pages.

Nowadays, besides the regular personal home pages, people also use other kinds of online services to expose their interests and background online, such as web blogs and the personal web services provided by Myspace.com. In Myspace.com, users can build their own online spaces and incorporate their other virtual representations, such as blogs, videos and articles, in their own spaces. All the information stored in a user's space can be used to represent this user. In this study, a person's personal website is used to represent a person, and the proposed People-Search algorithm is based on this representation method. A user's personal space in Myspace.com is very similar to a regular personal homepage. They both basically have three types of information: content, inlink and outlink. And they both have a quite clear boundary, e.g., in Myspace.com, it is very clear which page belongs to which user. Therefore, it is highly possible that the person representation method and the People-Search algorithm proposed in this study can be adapted and applied to the user spaces of Myspace.com to find similar users, too. They might also be applied to other similar kinds of online personal spaces.

Although the framework and algorithms proposed in this study are for personal websites and people search, one advantage of the proposed people search approach is that it can also be used to index and search objects other than people. For example, it can be modified to search companies having similar services and products from the Web.

6.2 Limitations and Discussions

This section presents the main limitations of this study.

Websites Having Insufficient Information to Represent a Person: In this study, it is proposed to use a person's website to represent this person's web appearance, in terms of his/her interests or background. However, in reality, some people's websites may not have enough information to represent their owners on the Web, which is one limitation of the study. For example, a person's website has only one main page (home page), and this main page contains only contact information. In this case, this person's website does not have enough information to describe and represent this person. Therefore, the chance that this website is retrieved as a relevant returned hit to a query is very small.

Availability and Diversity of Personal Websites: Nowadays, more and more people have built their own home pages. However, compared to the entire population of online users, the number of people owning personal websites is relatively low. Because the proposed people search method is based on people's home pages, this is one limitation of this study – the method can only search people who already have their home pages. At present, perhaps the majority of the home page owners are from academia, such as researchers, professors and college students. If this is the case, then it is also one of the limitations of this study: because the search space will be mainly limited to people in academia and most of search results will be from academia. However, in the dataset used in this experiment, academic and non-academic personal websites are roughly balanced. Among the 20,000 crawled websites, half of them belong to the following five top categories: Recreation, Sports, Home, Games and Arts. Most of the personal home pages

belonging to these categories are non-academic. As more and more companies provide online spaces and services for people to build personal websites, more and more non-academic people are able to build their home pages. It is reasonable to believe that the percentage of non-academic home page owners will gradually increase in the future.

Distinguishing Personal Websites from Non-personal Websites: The algorithms proposed in this study require that the system to be able to access the personal websites on the Web and index them. In the prototype systems, personal websites are collected from the ODP directory. But for a practical commercial system, this is not sufficient. It is better for a practical system to be able to automatically, continuously crawl the Web to index personal websites. How to design a web crawler, which can intelligently distinguish personal websites from non-personal websites (e.g., company web sites) and automatically crawl them, remains as an unsolved issue and will be one of the future research topics.

Dataset Cleaning: With ODP directory's high quality and popularity, on one hand, it is good for the evaluation, but on the other hand, this also raises a new issue. As discussed in Section 5.1, ODP pages are linked by many other web pages, and there are many mirror sites of the entire ODP personal website directory or part of it. Therefore, the personal home pages listed in ODP directory's web link structure might be skewed. Fortunately, all the inlinks and outlinks used in the experiment have been examined, and the ones caused by the fact that a personal home page appears in the ODP directory have been removed. However, part of the examination was done manually and their accuracy might be a potential problem. Therefore, this might be a potential limitation of the experiment.

Incomplete Inlinks: To calculate the link similarity between personal websites, inlink and outlink information of personal websites needs to be obtained. Obtaining outlinks is straightforward; it is done by extracting the outlinks from source html codes. In contrast, obtaining inlinks is a difficult task. To get all the inlinks of a page, it requires crawling and processing all the pages on the Web, which is impossible. Therefore, most studies use search engines to obtain page inlinks. In the evaluation, each page's inlinks were obtained from Yahoo and Google, and then were combined together. However, even the most popular search engines cannot guarantee to obtain all the inlinks of a web page, because they are only capable of crawling part of the entire Web. Therefore, in this study, it is very possible that for some pages only part of their inlinks were obtained. Actually, it is impossible to determine whether all the inlinks of a page have been collected, since people do not know how many inlinks there are for this page on the Web. Any study involving web page inlinks will also encounter this problem.

Multimedia Information: In the People-Search algorithm, the content similarity, inlink similarity and out link similarity are combined together for finding the similarity between two sites, but not multimedia similarity. However, some pages, such as an image page, do not have any outlink information or textual content information (but they may have inlink information). For this type of web pages, only the inlink information is exploited. If two websites contain the same images, the People-Search algorithm will not be aware of this, unless the URLs of the two images are the same. The People-Search algorithm cannot compare two images based on their content, such as color or pixels. Similar problem exists to audio files.

Direct Link: Sometimes two persons' websites may have direct links between them, which means one site may have one or more outlinks directly pointing to the other person's website. This kind of links is called direct link. For example, if we want to calculate the link similarity between two personal websites A and B, and there is a link in website A directly pointing to a page in website B, then this link is a direct link between A and B. This link is also one of A's outlinks and one of B's inlinks, but it is a special one, and is different from other regular outlinks or inlinks. Direct links are more important than other regular outlinks or inlinks, since they usually indicate a stronger relationship between two websites. Since there are very few direct links in the experimental dataset, in this study, direct links are not considered in the link similarity calculation. This is one of the limitations of this work.

Unavailability and Obsolescence of Personal Web Pages: Some people update their web pages very often. This raises a maintenance problem for the people search system. It needs to keep track of the changes of people's websites to ensure that the information stored in the system database is up-to-date and reflects the current interests/background of their owners. When crawling people's websites, the people search crawler also records the last update time for each crawled page. The system may use this data to check if the corresponding page has been updated by its owner since it was crawled last time. If yes, then the crawler may re-crawl that page. Another problem with web pages is the page obsolescence problem; some pages are not available anymore or their owners have not updated them for a very long time and therefore their contents may not reflect their owners' current interests or background anymore. To solve the page unavailability problem, the crawler may check each page's availability after a period of

time. If a page does not exist anymore, then the system may delete that page from the system database and update related data. For the second case – pages have not been updated for a long time and do not reflect the owners current interests anymore - there is no good solutions for it, because there is no way for the people search system to know if a page still reflects its author's current interests/background. One approach to reduce the effect of this problem is to decrease the ranks of their websites in the search results.

Noun Phrases: The evidence from language learning of children (Snow and Ferguson, 1997) and discourse analysis theories (Kamp, 1981) shows that the primary concepts in text are carried by noun phrases. In the proposed people search method, single words, instead of noun phrases, are used in the content similarity calculation. The reasons why single words are used in this study are explained below. Extracting single words from web pages is simpler and easier than extracting noun phrases. Most of the online search systems have used single words, instead of phrases, in their search operations. Although some systems are able to handle phrase queries, it does not mean these systems have extracted phrases from web pages; usually these systems record the locations of each word in a web page, and use this information to handle phrase queries (e.g., if the query is a phrase, the system will try to find documents which contain all the words in this phrase, and in which the locations of these words are adjacent to each other in the exact sequence). Identifying noun phrases is a time-consuming and difficult task, which involves part-of-speech tagging and other NLP techniques. The pages on the Web are heterogeneous, and most of them have various problems, such as spelling errors and grammatical errors. This makes it more challenging to identify noun phrases from web pages. Because of these reasons, single words, instead of noun phrases, are used in this

study. However, the proposed people search framework and algorithms might benefit from using noun phrases.

6.3 Contributions

This study contributes to web search areas, especially to the people/person search field. This study provides a new people search solution, which tries to address the problems the current methods have. The current people search approaches, such as regular search engines and online dating systems, either are not specialized for this task, require much user effort to build searchable people profiles, or are limited to certain domains. The proposed people search approach requires no user involvement in building searchable people profiles. A system based on the proposed method is able to search people from various domains, and has access to a large, diverse body of people. This is the first people search solution that can be applied to the entire Web.

The people search framework proposed in this study defines a person representation method, which uses a person's website to represent this person on the Web, and the types of information in a personal website that can be used to represent a person. To the author's knowledge, this study is the first attempt proposing to use a person's personal website to represent this person in people search.

Under the proposed person representation method and people search framework, in this study, all the possible algorithms have been explored for calculating similarity between two personal websites. This study also illustrates how to integrate different kinds of similarities methods together to get a unified similarity calculation method which could give the best performance in finding similar websites. Using a genetic

algorithm to fine tune these integrations is also presented in this work. The proposed algorithm and similarity integration methods will be useful in both the field of people search and also in other web search areas.

In this study, the ODP personal website directory was used as the dataset to evaluate the 14 algorithms. The way how these different performance measures (precision, recall, F and Γ measure) were used with this dataset to evaluate the algorithms and the problems observed, such as the ODP mirror sites problem described in Section 5.1, may shed some lights on other future studies involving the ODP directory.

The evaluation results show that the People-Search algorithm outperformed all other algorithms in calculating similarity between two personal website. This means one should exploit both the content and link information from all web pages of a personal website when calculating website similarity. Although this conclusion is obtained based on the experiments using personal websites, it might also be true for other kinds of websites. Other future studies involving website similarity calculation may consider this conclusion as a reference. The experimental results also show that, among the seven algorithms using only the main page of a personal website, the MainPage_Content_Link algorithm performed the best. This algorithm also integrates both the content and link information, but they are from only one page, the main page. Therefore, this result might also shed some light to other studies involving similarity calculation for regular web pages or other kinds of web pages. Another noticeable conclusion is that, for similarity calculation between two personal websites or between two main pages, content information was a better indicator than link information, and inlink information was better than outlink information. This conclusion might also help other future studies.

The framework and algorithms proposed in this study are based on the context of people search and personal websites. However, as mentioned before, potentially, they might also be applied to other applications involving website similarity calculation, such as searching similar organizations. An example is to apply the framework and algorithms in searching similar companies. A company usually has a website describing its products, technologies, etc. They together can represent this company online. Given one company's website, by applying the proposed algorithm, one may find other companies similar to this one. This would be very useful for users who are looking for companies selling similar products they are interested in. This would also be useful for a startup company to search its competitors.

6.4 Future Research

This study can be extended in the following directions:

The framework and algorithms proposed in this study require the people search system to be able to access the personal websites on the Web and index them. One issue raised by this is how to automatically index these personal websites. In the prototype systems, the personal websites are collected from the ODP personal website directory. But for a practical commercial system, this is not sufficient. It is better for a practical system to be able to automatically, continuously crawl the Web to index personal websites. How a web crawler intelligently distinguishes personal websites from non-personal websites (e.g., university web sites, company web sites, etc.) is one of the future research topics. One possible approach is described below. Given the home address of a personal website, the crawler developed for this study is able to crawl that entire personal

website. Therefore, if the crawler can distinguish the main page of a personal website (the URL of the main page of a personal website is this website's home address) from other kinds of pages, then it will be able to automatically crawl personal websites. The crawler will work as follows: first it crawls the Web just as a regular crawler does. When a web page is crawled, the crawler will determine whether or not this page is the main page of a personal website. If it is, then the crawler will crawl this whole personal website; if not, this page is discarded. The key step of this process is how the crawler determines if a page is the main page of a personal web site. One solution is to use a binary classifier, which has two classes – one for main pages of personal websites and the other one for all other kinds of pages. This classifier will classify each crawled page into one of the two classes. To build such kind of a classifier, the difficult part is to find a set of appropriate features for each class, which can be used by the classifier to distinguish the two types of pages. For a general classifier, usually the important terms from the training pages are used as the class features. In the case of this study, there are also other special features that can be exploited. Examples of other possible features are special trigger phrases/sentences and URL patterns of the personal home pages. For example, the sentence "Welcome to my home page" or "This is XYZ's home page" may be considered as trigger sentences. Many home pages whose owners are in academia also contain some special phrases, such as "Research Interests" and "Education" These kinds of phrases or sentences are good indicators of personal home pages. Some personal home pages have special URL patterns. For example, home pages of NJIT students have the following URL pattern: http://web.njit.edu/~xyz. URL patterns can also be used as class features.

By using appropriate features and training pages, a classifier can be built for identifying personal home pages.

Another future research direction is to extend the people search system to include the "search by page" function. The current people search prototype system is a "search by site" system, which means the query is a personal website. "Search by page" allows the query to be a single web page or even a regular document, such as a plain text file or word file. As mentioned in Section 5.4.8, some personal websites are heterogeneous, which means its pages are not about only one topic, but multiple topics. For example, one page is about computer programming, and another page is talking about this person's hobby - fishing. When using a whole website as a query, if the site is heterogeneous, the search results may also be heterogeneous. For example, using the site mentioned above as the query, some of the returned websites may be related to computer programming, while others may be about fishing. This is not desirable if the user is only interested in one topic, such as computer programming. Usually a single document is homogenous, meaning its content is mainly about one main theme. "Search by page" would allow users to specify more specific information needs through the query page, and the search results would also be more relevant. "Search by page" is not difficult to implement. The only problem is the computation cost – similarities between personal pages, instead of sites, need to be computed, which would exponentially increase the computation cost. Fortunately, this can be done offline.

Personalized people search will also be one of the future research topics. In the current people search prototype system, users can search other people based on a given website (its home page URL) or by a set of keywords. In the last paragraph, "search by

page" has been discussed, which is a step toward personalized search. To make the system more personalized, four more features may be implemented:

1. Users can choose any part of a document (by highlighting it) as the query, such as a paragraph or a couple of sentences. The system will return personal websites relevant to this kind of special query. This function is similar to Yahoo's Y!Q (see Section 2.4.2 for details).

2. The system can display all the content-bearing words of an indexed personal website to users. Users then choose a list of words from them as the query.

3. Users can specify a list of terms that the returned websites should not contain. This works the same way as the Boolean operator 'NOT.'

4. The above three features are for searching. The fourth feature is for indexing. A user can specify which pages of his/her website, instead of the entire site, to be indexed by the system to represent this user online.

The fourth feature is for indexing, so it will not raise any important system performance problem, since the indexing is done offline. The first three features will increase the system response time if users highlight or specify too many terms, since the searching operation is done on the fly.

In this study, a link extracted from the pages of a person's website is considered an internal link if its link contains the root directory of this site (meaning the page pointed to by this link resides in the root directory of this site or one of its sub directory), otherwise, it is considered an outlink of this site. One problem with this standard is: sometimes a link, e.g., a link to a paper in an electronic database written by the site owner, should be considered as an internal link of a site even though the corresponding page is not physically located in that site. However, the Web Page Processing Module of the people search system treats it as an outlink. The main effect of this problem is that the page pointed to by this link will not be included in this site's content similarity calculation, which is not desirable. How to use semantic information around a link to

determine if this link is an internal link will be one of the future research topics. A possible solution is to use a binary classifier to classify each link into one of two categories, internal link category and outlink category. To build this classifier, a training dataset is needed, which contains training samples for the two kinds of links. In the training dataset, a sample link should have the following information: its URL and words appearing inside or near the link anchor. An anchor window is used to specify the size of the text around the anchor. In the training stage, the window size can be trained to find which window size would give the best performance. The terms within the anchor window are used as category features. Some trigger words/phrases could be used as additional features and be given higher weights in the classifier. One example of such words/phrases is "my papers" in the sentence of "one of my papers on digital libraries is available at http://www.xyz.com/publication/abc.html." In this case, the phrase "my papers" indicates that this link should be treated as an internal link, instead of an outlink. Developing such kind of a classifier will be one of the future research topics.

APPENDIX A

TERM LIST, INLINK LIST AND OUTLINK LIST OF A PERSONAL WEBSITE

This appendix contains an example of the term list, inlink list and outlink list of a personal website, http://www.linearity.org/cas/.

Table A.1 Term List of a Personal Website

Word stem	Frequency in this site	Original word forms
theory	63	Theory
logic	59	logics,logic
classical	35	Classical
calculu	31	calculus,calculi
proof	29	proof,proofs
typ	28	type,types
semantic	26	Semantics
program	25	programs,program,programming
languag	22	language,languages
correspondenc	21	correspondence,correspondences
modal	20	Modal
research	19	Research
interest	19	interested,interest,interests
referee	18	referee,refereeing,referees
thesi	18	thesis,theses
linear	18	Linear
interaction	17	interaction
talk	17	talks,talk
system	16	system,systems
computation	16	computation,computations
provid	15	provided,provide,providing
publication	15	publications,publication
net	15	nets,net
scienc	15	science,sciences
arithmetic	14	arithmetic
pag	14	pages,page
graph	14	graph,graphs
university	14	university
abstract	13	abstract,abstracts
reduction	13	reductions,reduction
computer	12	computer
dresden	12	dresden
teach	12	teaching,taught
formulae-as-typ	12	formulae-as-types
account	12	account
functional	12	functional
approach	11	approach,approaches
danc	11	dancing
constructiv	11	constructive
assertion	10	assertion,assertions
stewart	10	stewart
inferential	10	inferential
rol	10	role,roles
show	10	show,shown
term	10	terms
personal	10	personal
charl	10	charles
control	10	control
relationship	10	relationships,relationship
principl	10	principles,principle
workshop	9	workshop,workshops
mean	9	means,meaning
rul	9	rule,rules
work	9	work
optimal	9	optimal
master	9	masters
nam	9	names,naming
supervi	8	supervised,supervising

Table A.1 Term List of a Personal Website (Continued)

Word stem	Frequency in this site	Original word forms
theoretical	8	theoretical
question	8	questions,question
international	8	international
group	8	group
model	7	model,modelling,models
activity	7	activity
chronology	7	chronology
opinion	7	opinion
logical	7	logical
computational	7	computational
doctoral	7	doctoral
foundation	7	foundation,foundations
formalism	6	formalism,formalisms
intuitionistic	6	intuitionistic
lectur	6	lectures,lecture,lecturing
equality	6	equality
lambda	6	lambda
set	6	set,setting
mathematic	6	mathematics
design	6	design
lambda-mu	6	lambda-mu
version	6	version
conferenc	6	conferences,conference
paul	6	paul
grammar	6	grammars,grammar
extend	6	extending,extend
understand	6	understanding,understood
present	6	presented,present
curry-howard	6	curry-howard
policy	6	policy
systematic	6	systematic
continuation	6	continuations
theori	6	theories
condition	5	condition
context	5	context
develop	5	develop,developed
symp	5	symp
proc	5	proc
distribut	5	distributed
michael	5	michael
phiniki	5	phiniki
luk	5	luke
general	5	general
issu	5	issues,issue
structur	5	structures
ong	5	ong
treatment	5	treatment,treatments
berlin	5	berlin
characterisation	5	characterisation,characterisations
formal	5	formal
sequent	5	sequent
hold	5	hold,holds,held
inferenc	5	inference,inferences
submit	5	submit,submitted
express	5	expressed,expressing,express
entitl	5	entitled
shar	5	sharing
lisa	5	lisa
examin	5	examined,examine
paper	5	paper,papers
induction	5	induction
annual	5	annual
representation	5	representation,representations
assert	5	asserting,assert
carry	5	carried,carry
obtain	5	obtaining,obtain,obtained
geometry	5	geometry
implementation	5	implementation,implementations
philosophy	5	philosophy
interpretation	4	interpretation
intensional	4	intensional
claim	4	claim,claiming

Table A.1 Term List of a Personal Website (Continued)

Word stem	Frequency in this site	Original word forms
publish	4	publishing,published
natural	4	natural
fil	4	file
idea	4	idea,ideas
property	4	property
press	4	press
author	4	author,authors
report	4	reports,report
discuss	4	discuss,discussed
form	4	form
boston	4	boston
answer	4	answer,answers
practical	4	practical
specification	4	specifications,specification
propositional	4	propositional
seri	4	series
colleg	4	college
anonymou	4	anonymous
basi	4	basis
deep	4	deep
dummett	4	dummett
describ	4	describe
articl	4	article,articles
application	4	applications
gentzen	4	gentzen
enjoy	4	enjoy,enjoys
transformation	4	transformations,transformation
writ	4	write,written
algorithm	4	algorithm,algorithms
conceptual	4	conceptual
structural	4	structural
oxford	4	oxford
ir	4	ir
prawitz	4	prawitz
network	4	networks,networking,network
dag	4	dag
department	4	department
sentenc	4	sentences,sentence
technical	4	technical
port	3	Port
process	3	process,processes
siz	3	Size
extension	3	extension,extensions
mak	3	make,made
pdf	3	Pdf
proceeding	3	Proceedings
reducibility	3	Reducibility
church	3	Church
journal	3	journal,journals
practic	3	Practice
wh-interrogativ	3	wh-interrogatives
content	3	Content
argu	3	argue,argues
direct	3	Direct
lambda-calculu	3	lambda-calculus
construct	3	construct,constructs
abadi	3	Abadi
investigation	3	investigations
determin	3	determined,determine
properti	3	properties
local	3	local
featur	3	features,feature
cas	3	case
referenc	3	referencing,references
permit	3	permits,permitting
current	3	current
dphil	3	dphil
literatur	3	literature
school	3	school
attempt	3	attempting,attempts
partial	3	partial
view	3	view
past	3	past
explain	3	explain
read	3	reading
linguistic	3	linguistics

Table A.1 Term List of a Personal Website (Continued)

Word stem	Frequency in this site	Original word forms
simpl	3	simple
pronoun	3	pronouns
procedur	3	procedure,procedures
result	3	results,result
connection	3	connections,connection
top	3	top
link	3	links
symmetric	3	symmetric
formulation	3	formulation
harmony	3	harmony
hilbert	3	hilbert
stouppa	3	stouppa
ambient	3	ambients,ambient
inversion	3	inversion
point	3	point
stat	3	states,state
technisch	3	technische
implement	3	implements,implementing,implemented
fall	3	falls,fall
gonthier	3	gonthier
studi	3	studies
introduction	3	introduction
section	3	section
rewrit	3	rewrite
dat	3	date
universitaet	3	universitaet
examination	3	examinations,examination
harmonic	3	harmonic
introduc	3	introduce
whilst	3	whilst
great	3	great
pragmatic	3	pragmatics,pragmatic
reason	2	reasoning
phd	2	phd
axiom	2	axiom,axioms
class	2	class,classes
insight	2	insight,insights
academic	2	academic
cut	2	cut
telephon	2	Telephone
call-by-valu	2	call-by-value
man	2	Man
success	2	Success
proof-theoretic	2	proof-theoretic
normal	2	Normal
computationally	2	computationally
zhaohui	2	Zhaohui
student	2	student,students
confident	2	Confident
field	2	field,fields
activ	2	active
nuel	2	nuel
method	2	method
analogou	2	analogous
girard	2	girard
access	2	access
travel	2	travel
robert	2	robert
researcher	2	researcher
number	2	number
streicher	2	streicher
construction	2	construction,constructions
respect	2	respect
abstraction	2	abstraction
luo	2	luo
area	2	area
widespread	2	widespread
manchester	2	manchester
expression	2	expression,expressions
early	2	earlier
elimination	2	elimination
los	2	lost,lose
employ	2	employed
door	2	door
contribution	2	contribution,contributions
wallen	2	wallen
need	2	needed
informativ	2	informative
organisation	2	organisation

Table A.1 Term List of a Personal Website (Continued)

Word stem	Frequency in this site	Original word forms
propo	1	proposed
normalisation	1	normalisation
fortnow	1	fortnow
anger	1	anger
attend	1	attended
devis	1	devise
winter	1	winter
risk	1	risk
predicat	1	predicate
incompatibl	1	incompatible
finally	1	finally
tremendously	1	tremendously
edinburgh	1	edinburgh
enrol	1	enrolled
a-level	1	a-levels
unclear	1	unclear
leed	1	leeds
board	1	boards
bonn	1	bonn
prov	1	proven
dresden-johannstadt	1	dresden-johannstadt
precisely	1	precisely
promi	1	promising
cod	1	coded
reach	1	reach
fakultaet	1	fakultaet
highly	1	highly
generic	1	generic
alan	1	alan
impression	1	impression
relativ	1	relative
suffer	1	suffers
leav	1	leave
address	1	address
statu	1	status
remot	1	remote
purify	1	purified
complaint	1	complaint
gordon	1	gordon
pezz	1	pezze
disrespectful	1	disrespectful
relation	1	relation
integrity	1	integrity
accept	1	accepted
resum	1	resum
establish	1	established
meet	1	met
regard	1	regarded
potential	1	potential
flexibl	1	flexible
slur	1	slur
peopl	1	people
church-rosser	1	church-rosser
plac	1	place
identify	1	identify
inspir	1	inspired
effectiv	1	effective
essential	1	essential
locality	1	locality
usual	1	usual
existenc	1	existence
printer	1	printers
conjunction	1	conjunction
framework	1	frameworks
quantification	1	quantification
largely	1	largely
int	1	int
stand	1	stands
room	1	room
subsequent	1	subsequently
thread	1	threads
visual	1	visual
fin	1	fine
programm	1	programme
compositional	1	compositional
directness	1	directness
constru	1	construed
compil	1	compiling
consequenc	1	consequence
curry	1	curry
preferenc	1	preferences
manner	1	manner
alexander	1	alexander

Table A.1 Term List of a Personal Website (Continued)

Word stem	Frequency in this site	Original word forms
connectiv	1	connectives
distinction	1	distinction
run	1	running
constituent	1	constituent
larg	1	large
reject	1	rejected
referent	1	referent
global	1	global
opaqu	1	opaque
depend	1	depends
essay	1	essays
cognitiv	1	cognitive
bern	1	berne
clau	1	clauses
william	1	william
chill	1	chilling
influential	1	influential
display	1	display
deductiv	1	deductive
lic	1	lics
happy	1	happy
consideratio n	1	consideration
seminal	1	seminal
fac	1	faces
informazion	1	informazione
minor	1	minor
slid	1	slides
bertrand	1	bertrand
compar	1	compared
quality	1	quality
solv	1	solve
constructivit y	1	constructivity
aegi	1	aegis
two-factor	1	two-factor
lafont	1	lafont
ability	1	ability
generality	1	generality
flow	1	flow
combinator y	1	combinatory
physic	1	physics
analysi	1	analysis
select	1	selected
asymmetric	1	asymmetric
investigat	1	investigate
rescu	1	rescuing
tool	1	tools
machin	1	machine
realisation	1	realisation
exact	1	exact
definitional	1	definitional
rough	1	rough
deriv	1	derived
hsbc	1	hsbc
untersuchun gen	1	untersuchungen
logisch	1	logische
axiomatisati on	1	axiomatisation
modularity	1	modularity
choic	1	choice
observ	1	observing
theoretically	1	theoretically
briefly	1	briefly
semanticall y	1	semantically
attendant	1	attendant
organis	1	organise
behavioural	1	behavioural
commentary	1	commentary
xml	1	xml
reu	1	reus
fruitful	1	fruitful
main	1	main
mail	1	mailing
search	1	search
handout	1	handout
draw	1	draws
peat	1	peat
alternativ	1	alternative
protocol	1	protocol
philosophic al	1	philosophical
letter	1	letter
admit	1	admits
firstly	1	firstly
eta	1	eta
former	1	formers

Table A.1 Term List of a Personal Website (Continued)

Word stem	Frequency in this site	Original word forms	Word stem	Frequency in this site	Original word forms
deserv	1	deserve	constant	1	constants
kpmg	1	kpmg	cofounder	1	cofounder
significant	1	significant	trimester	1	trimesters
situat	1	situate	paraphras	1	paraphrase
superiority	1	superiority	generally	1	generally
forward	1	forward	assist	1	assisted
wh-question	1	wh-question	limitation	1	limitations
organi	1	organised	pari	1	paris
technology	1	technology	modern	1	modern
rely	1	relied	sufficient	1	sufficient
optimism	1	optimism	vicariou	1	vicarious
bear	1	born	peer-review	1	peer-reviewed
specific	1	specific	contact	1	contact
attractiv	1	attractive	independently	1	independently
pawel	1	pawel	upper	1	upper
hour	1	hour	ber	1	ber
marwick	1	marwick	label	1	labelled
dual	1	dual	simply-typ	1	simply-typed
steffen	1	steffen	tremendou	1	tremendous
isomorphism	1	isomorphism	africa	1	africa
effectively	1	effectively	impredicativ	1	impredicative
questioner	1	questioner	important	1	important
download	1	download	first-year	1	first-year
extraction	1	extraction	voic	1	voice
format	1	formatted	politecnico	1	politecnico
augment	1	augmenting	importanc	1	importance
griffin	1	griffin	downloadabl	1	downloadable
contrast	1	contrast	defenc	1	defence
definition	1	definitions	assistant	1	assistant
sens	1	sense	styl	1	style
constitutiv	1	constitutive	systematicity	1	systematicity
conscientiousness	1	conscientiousness	begin	1	begin
heriot-watt	1	heriot-watt	collection	1	collection
focu	1	focus	preprint	1	preprint
chemistry	1	chemistry	standpoint	1	standpoint
accessibility	1	accessibility	community	1	community
inform	1	informing	professionalism	1	professionalism
path	1	path	wilki	1	wilkie
external	1	external	participant	1	participant
hop	1	hopes	goal	1	goal
oversee	1	overseeing			
hom	1	home			

Table A.1 Term List of a Personal Website (Continued)

Word stem	Frequency in this site	Original word forms	Word stem	Frequency in this site	Original word forms
parallel	1	parallel	similarly	1	similarly
imperial	1	imperial	congruent	1	congruent
demonstrat	1	demonstrate	extensional	1	extensional
ferro	1	ferro	enabl	1	enables
conduciv	1	conducive	high-level	1	high-level
albeit	1	albeit	involv	1	involved
editorial	1	editorial	impact	1	impact
tun	1	tuning	call	1	call
expansion	1	expansions	sympathy	1	sympathy
conditional	1	conditionals	requirement	1	requirements
relat	1	relate	peter	1	peter
speech	1	speech	equation	1	equations
central	1	central	overview	1	overview
ad-hoc	1	ad-hoc	uninformati v	1	uninformative
professional	1	professional	acknowledg ement	1	acknowledgemen ts
counteract	1	counteracted	imperativ	1	imperative
unify	1	unifying	hardwar	1	hardware
middl	1	middle	denotational	1	denotational
london	1	london	elsevier	1	elsevier
seat	1	seated	no-on	1	no-one
readback	1	readback	garbag	1	garbage
presuppositi on	1	presupposition	possibility	1	possibility
mobil	1	mobile	germany	1	germany
harry	1	harry	occasional	1	occasional
summary	1	summary	north-holland	1	north-holland
lastly	1	lastly	west	1	west
alongsid	1	alongside	committee	1	committees
summari	1	summarised	anytim	1	anytime
fixpoint	1	fixpoints	dissertation	1	dissertation
canadian	1	canadian	nilsen	1	nilsen
member	1	members	tim	1	time
unpublish	1	unpublished	matter	1	matter
kind	1	kind	professor	1	professor
relegat	1	relegate	admissibilit y	1	admissibility
decidabl	1	decidable	similar	1	similar
analogu	1	analogue	anonymousl y	1	anonymously
shap	1	shaped	bound	1	bounded
axiomatic	1	axiomatic	originat	1	originates
simplifi	1	simplifies	editor	1	editors
oppo	1	opposed	beginner	1	beginner
list	1	listed			
michel	1	michel			
defect	1	defect			

Table A.1 Term List of a Personal Website (Continued)

Word stem	Frequency in this site	Original word forms
jump	1	jump
vein	1	vein
murthy	1	murthy
uk	1	uk
weakness	1	weaknesses
generali	1	generalised
well-behav	1	well-behaved
temporary	1	temporary
anonymi	1	anonymised
underpinnin g	1	underpinnings
speak	1	speak
distinctively	1	distinctively
endanger	1	endanger
effect	1	effect
energy	1	energy
hans-grundig-str	1	hans-grundig-str
moderator	1	moderator
thought	1	thoughts
background	1	background
task	1	task
dog	1	dogs
syndicat	1	syndicate
meaning	1	meanings
location	1	location
unit	1	unite
proposal	1	proposal
institut	1	institute
aceto	1	aceto
cambridg	1	cambridge
dipartiment o	1	dipartimento
tax	1	tax
stability	1	stability
additiv	1	additives
negativ	1	negative
web	1	web
ps	1	ps
construtor	1	construtors
component	1	component
absurdity	1	absurdity
secondary	1	secondary
stak	1	stake
intelligenc	1	intelligence
openly	1	openly
consultant	1	consultant
reread	1	rereading
agg	1	agg
unsign	1	unsigned
mu	1	mu
compatibl	1	compatible
judg	1	judges
respons	1	response
maintain	1	maintained
ma	1	ma
spotlight	1	spotlight
li	1	lies
sway	1	swayed
birmingham	1	birmingham
revision	1	revisions
progress	1	progress
logik	1	logik
msc	1	msc
detail	1	detailed
schem	1	scheme
act	1	acts
wittgenstein	1	wittgenstein
fourth	1	fourth
fundamental	1	fundamental
pur	1	pure
processor	1	processors
constructor	1	constructors
compiler	1	compiler
worcester	1	worcester
tait	1	tait
acount	1	acount
individual	1	individual
concern	1	concerned
distinguish	1	distinguish
incorporatio n	1	incorporation
elaborat	1	elaborated
pragmatist	1	pragmatist
geometric	1	geometric
fail	1	fails
cardelli	1	cardelli
compulsory	1	compulsory
das	1	das

Table A.2 Inlink List of a Personal Website

Inlink	Frequency
http://alessio.guglielmi.name/res	2
http://alessio.guglielmi.name/res/cos	2
http://alessio.guglielmi.name/res/cos/crt.html	2
http://alessio.guglielmi.name/res/cos/ML	2
http://botw.org/new/all/08192005.cfm	1
http://classical_logic.iqexpand.com/	1
http://community.schemewiki.org/?charles-stewart	2
http://community.schemewiki.org/?p=charles-stewart&c=hv&t=1098807729	2
http://consequently.org/edit/page/Charles_Stewart	3
http://consequently.org/edit/page/PnC_Chapter_2	1
http://consequently.org/edit/page/Users	2
http://consequently.org/writing/invention/	2
http://consequently.org/writing/pc	2
http://crumpled.com/cp/personal/000543.html	2
http://en.wikipedia.org/wiki/Talk:Curry-Howard_correspondence	1
http://en.wikipedia.org/wiki/User:Chalst	2
http://gerhard_gentzen.iqexpand.com/	1
http://homepages.inf.ed.ac.uk/v1phanc1/people.html	2
http://hurryupharry.bloghouse.net/archives/2005/02/14/thousands_of_neonazis_march_in_dresden.php	2
http://iccl.tu-dresden.de/~paola	2
http://lambda-the-ultimate.org/node/view/1078	1
http://libarynth.f0.am/cgi-bin/twiki/rdiff/Main/VisualProgramming	1
http://libarynth.f0.am/cgi-bin/twiki/view/Main/VisualProgramming	1
http://oliverkamm.typepad.com/blog/2004/05/doityourself_ec.html	2
http://robots.net/person/evilrobots/diary.html?start=7	2
http://tar.weatherson.net/archives/004211.html	1
http://thatlogicblog.blogspot.com/2005/08/proofs-as-games.html	2
http://timlambert.org/2004/10/razor2/	2
http://timlambert.org/2005/04/horowitzspam	2
http://types.bu.edu/category.html	2
http://types.bu.edu/participants.html	2
http://types.bu.edu/reports/Ong+Ste:curhff.html	1
http://types.bu.edu/reports/Stewart:fortcf.html	1
http://www.advogato.org/person/chalst/	2
http://www.aloeverasite.com/formulaeforaloeveramoisturizinglotionfree/	1
http://www.bigsearchportal.com/YnNwXzkyNDk2Mg==.aspx	2
http://www.blogger.com/email-post.g?blogID=7108230&postID=110960451238549869	2
http://www.deerlakesearch.com/default?p=924962	2
http://www.findallyouneed.com/cgi-bin/se/smartsearch.cgi?keywords=types	1
http://www.iccl.tu-dresden.de/~ozan/maude_cos.html	2
http://www.ki.inf.tu-dresden.de/~guglielm/group	2
http://www.ki.inf.tu-dresden.de/~guglielm/group/events.html	2
http://www.ki.inf.tu-dresden.de/~guglielm/Research/	2
http://www.ki.inf.tu-dresden.de/~guglielm/Research/list.html	2

Table A.2 Inlink List of a Personal Website (Continued)

Inlink	Frequency
http://www.ki.inf.tu-dresden.de/~guglielm/WPT	2
http://www.ki.inf.tu-dresden.de/~guglielm/WPT05	2
http://www.ki.inf.tu-dresden.de/~guglielm/WPT2	2
http://www.ki.inf.tu-dresden.de/~guglielm/WSPT	2
http://www.ki.inf.tu-dresden.de/Research/IQN/IQN_Events.html	2
http://www.linearity.org/	2
http://www.linearity.org/cas	5
http://www.logicandlanguage.net/archives/2005/04	2
http://www.logicandlanguage.net/archives/2005/04/dummett_on_harm.html	1
http://www.logicandlanguage.net/archives/2005/04/even_more_harmo.html	2
http://www.logicandlanguage.net/archives/philosophy_of_logic	2
http://www.medlina.com/logicians.htm	2
http://www.muffinversion.com/inversionprinciple/	1
http://www.mymbacentre.com/symbiosiscorrespondencemba/	1
http://www.mysociety.org/?p=82	2
http://www.prooftheory.org/list.html	2
http://www.prooftheory.org/sd05	2
http://www.prooftheory.org/sd05/program.html	2
http://www.sstudiesheadlines.com/calculusstewart.html	2
http://www.stephenpollard.net/001667.html	2
http://www.stephenpollard.net/001819.html	2
http://www.stephenpollard.net/cgi-bin/mt/mt-comments.cgi?entry_id=1667	2
http://www.stevepavlina.com/blog/2005/09/are-humans-carnivores-or-herbivores-2	2
http://www.thegolf-3.com/golfertomwatson/	1
http://www.ucalgary.ca/~rzach/logblog/2005/02/proofs-and-types.html	1
http://www.ucalgary.ca/~rzach/logblog/2005/03/eliminating-cuts.html	2
http://www.ucalgary.ca/~rzach/logblog/2005/03/new-blog-tonk-and-normalization.html	2
http://www.ucalgary.ca/~rzach/logblog/2005/04/modal-logic-textbooks.html	2
http://www.ucalgary.ca/~rzach/logblog/2005/04/motivating-intro-logic-for-philosophy.html	2
http://www.ucalgary.ca/~rzach/logblog/2005_02_01_archive.html	2
http://www.wv.inf.tu-dresden.de/~hein	2
http://www.yourchemistrynews.info/formulae.html	3
http://yglesias.typepad.com/matthew/2005/04/relativism_and_.html	2

Table A.3 Outlink List of a Personal Website

Outlink	Frequency
ftp://achilles.bu.edu/pub/cas/marburg-handout.ps	1
ftp://achilles.bu.edu/pub/cas/marburg-slides.ps	1
ftp://achilles.bu.edu/pub/cas/popl97.ps	1
http://achilles.bu.edu/cas/index.html	4
http://achilles.bu.edu/cas/publications.html	4
http://alessio.guglielmi.name	1
http://fortnow.com/lance/complog/2004/11/public-referee-reports.html	1
http://oldwww.comlab.ox.ac.uk/oucl/courses/undergrad/fp-ad.html	1
http://oldwww.comlab.ox.ac.uk/oucl/courses/undergrad/imper.html	1
http://oldwww.comlab.ox.ac.uk/oucl/courses/undergrad/log-hw.html	1
http://radio.weblogs.com/0110772/2004/11/05.html#a1643	1
http://tfs.cs.tu-berlin.de	2
http://types.bu.edu/progthewebfall00.html	1
http://types.bu.edu/progthewebfall99.html	1
http://users.comlab.ox.ac.uk/luke.ong	1
http://web.comlab.ox.ac.uk/oucl/research/areas/foundations	2
http://web.comlab.ox.ac.uk/oucl/work/luke.ong	4
http://www.cl.inf.tu-dresden.de/compulog	2
http://www.cl.inf.tu-dresden.de/compulog/lectures/winter04/logic2004.html	1
http://www.cl.inf.tu-dresden.de/compulog/lectures/winter04/scl2004.html	1
http://www.cs.auc.dk/~luca/pa-diary/05-11-2004.html	1
http://www.cs.brandeis.edu	3
http://www.cs.brandeis.edu/~mairson	1
http://www.cs.bu.edu/groups/church/home.html	3
http://www.cs.bu.edu/groups/church/progtheweb.html	1
http://www.cs.man.ac.uk/aiml04/index.html	2
http://www.dur.ac.uk/~dcs0zl	2
http://www.iccl.tu-dresden.de	4
http://www.inf.ethz.ch/personal/meyer/publications/online/whysign	1
http://www.ki.inf.tu-dresden.de/~guglielm/wpt	3
http://www.linearity.org	2
http://www.linearity.org/cas	3
http://www.linearity.org/linear	2
http://www.math.rutgers.edu/~zeilberg/opinion3.html	1
http://www.math.rutgers.edu/~zeilberg/opinion61.html	1
http://www.math.uni-bonn.de/people/fotfs/iv	1
http://www.mathematik.tu-darmstadt.de/ags/ag14/mitglieder/streicher-en.html	1
http://www.mathengine.com/investors/management.html	2
http://www.qinfo.org/people/nielsen/blog/archive/000146.html	1
http://www.swan.ac.uk/compsci/eventsfolder/abstracts.ps	1
http://www.wv.inf.tu-dresden.de	2
http://www.wv.inf.tu-dresden.de/people/index.php?hoelldobler.html	1
http://zls.mimuw.edu.pl/~urzy/home.html	1

APPENDIX B

HUMAN EVALUATION CONSENT FORM

Appendix B contains the consent form used in the human evaluation of this study.

CONSENT TO PARTICIPATE IN A RESEARCH STUDY

TITLE OF STUDY:

Search People Sharing Similar Interests from the Web

RESEARCH STUDY:

I have been asked to participate in a research study under the direction of Quanzhi Li . Other professional persons who work with them as study staff may assist to act for them.

PURPOSE:

To evaluate the effectiveness of algorithms for finding similar people based on their personal websites.

DURATION:

My participation in this study will last for 1 day -3 weeks

PROCEDURES:

I have been told that, during the course of this study, the following will occur:

1. I will be asked to voluntarily use an online search system which is used to search people sharing similar interests based on their personal websites.

2. I will be asked to voluntarily evaluate the search results and complete the pre and post questionnaires.

PARTICIPANTS:

I will be one of about 40 participants to participate in this trial.

EXCLUSIONS:

I will inform the researcher if any of the following apply to me:

- I do not wish to use the system for any reason.

- I do not wish to complete the questionnaires for any reason.

RISKS/DISCOMFORTS:

I have been told that the study described above may involve the following risks and/or discomforts:

None known or anticipated discomforts. Security of the system might be at risk of computer hacking, as it is in any computer system. Every effort (e.g., blocking the unused ports, update the system with the latest patches, and checking system logs as frequently as possible to catch abnormal usage) will be made to keep the system secure from hacking.

There also may be risks and discomforts that are not yet known.

I fully recognize that there are risks that I may be exposed to by volunteering in this study which are inherent in participating in any study; I understand that I am not covered by NJIT's insurance policy for any injury or loss I might sustain in the course of participating in the study.

CONFIDENTIALITY:

I understand confidential is not the same as anonymous. Confidential means that my name will not be disclosed if there exists a documented linkage between my identity and my responses as recorded in the research records. Every effort will be made to maintain the confidentiality of my study records. If the findings from the study are published, I will not be identified by name. My identity will remain confidential unless disclosure is required by law.

PAYMENT FOR PARTICIPATION:

I have been told that I will receive no monetary compensation for my participation in this study.

RIGHT TO REFUSE OR WITHDRAW:

I understand that my participation is voluntary and I may refuse to participate, or may discontinue my participation at any time with no adverse consequence. I also understand that the investigator has the right to withdraw me from the study at any time.

INDIVIDUAL TO CONTACT:

If I have any questions about my treatment or research procedures, I understand that I should contact the principal investigator at:

Quanzhi Li
GITC5500
Information System department,
New Jersey Institute of Technology
Newark, NJ07102
Tel:(973) 596-5655, Email: QL23@njit.edu

If I have any addition questions about my rights as a research subject, I may contact:

Dawn Hall Apgar, PhD, IRB Chair
Jersey Institute of Technology
Martin Luther King Boulevard
NJ07102
(973) 642-7616
dawn.apgar@njit.edu

SIGNATURE OF PARTICIPANT

I have read this entire form, or it has been read to me, and I understand it completely. All of my questions regarding this form or this study have been answered to my complete satisfaction. I agree to participate in this research study.

My Name: _____ **Date:** _____

APPENDIX C

PRE-EVALUATION QUESTIONNAIRE

Appendix C contains the pre-evaluation questionnaire used in the human evaluation of this study.

Pre-evaluation Questionnaire:

Dear participant,

Before participating in the study, please take a few minutes to fill in this questionnaire. The information you provide will help us achieve a better understanding of the evaluation results. Your answers are strictly confidential.

1. My major / work area: _____

2. I have _____ years of experience using computer, and _____years of experience using Internet.

3. On average, how many hours per day do you spend on Internet?

   ___Less than 1 ___ 1- 2 ___ 2 - 4 ___ 4 - 6 ___ 6 - 8

   ___ More than 8

4. I have _____ years of experience using search engines. The search engines I usually use are: _____

5. My experience in using search engines (please check one):

   (Novice) 1 2 3 4 5 6 7 (Expert)

6. I use search engines for (please check all that applies):

   __My research or my work _ my study __entertainment (search music, movie, etc.) ___news ___knowledge acquirement (history, politics, etc.) __ search people Others (please specify)_____

7. If you have searched or will search people online, what are your purposes? (Please check all that applies.)

   __Find other people in my interest areas __ Find experts in certain areas

   __Find a person I am interested in __ Search celebrities

   Others (please specify)_____

8. How do you find information of people you are interested in or find people sharing similar interests with you (e.g., having similar research interests) from web? (Please check all that applies)

   __Use search engines _____use online community ___use online directory (e.g., yahoo directory) ___ from paper citations. Others (please specify)_____

APPENDIX D

POST-EVALUATION QUESTIONNAIRE

Appendix D contains the post-evaluation questionnaire used in the human evaluation of this study.

Post-evaluation Questionnaire:

Dear participant,

Thank you very much for participating in this experiment!! Please take a few minutes to give us some feedback about this pilot experiment. Your answers are strictly confidential and highly appreciated.

1. I would like to use a people search system similar to this one in the future

   Strongly Disagree 1 2 3 4 5 6 7 Strongly Agree

2. To search similar people on the Web, I prefer the method used in this experiment over other ones.

   Strongly Disagree 1 2 3 4 5 6 7 Strongly Agree

3. Please input your comments, opinions or suggestions about the people search system you very much for participating in this experiment!! Please take a few minutes to give used in this experiment in this box.

APPENDIX E

STOP WORDS

Appendix E contains the stop words used in the stop words removal process during automatic indexing of the dataset.

Table E.1 Stop Words List

a	appreciate	brief	does
able	appropriate	but	doing
about	apr	by	done
above	april	c	Down
according	are	came	downwards
accordingly	around	can	During
across	as	cannot	E
actually	aside	cant	e.g
after	ask	cause	e.g.
afterwards	asking	causes	each
again	associated	certain	edu
against	at	certainly	eg
all	aug	changes	eight
allow	august	clearly	either
allows	available	co	else
almost	away	com	elsewhere
alone	awfully	come	enough
along	b	comes	entirely
already	be	concerning	especially
also	became	consequently	et
although	because	consider	etc
always	become	considering	even
am	becomes	contain	ever
among	becoming	containing	every
amongst	been	contains	everybody
an	before	corresponding	everyone
and	beforehand	could	everything
another	behind	course	everywhere
any	being	currently	ex
anybody	believe	d	exactly
anyhow	below	dec	example
anyone	beside	december	except
anything	besides	definitely	F
anyway	best	described	far
anyways	better	despite	feb
anywhere	between	did	february
apart	beyond	different	few
appear	both	do	fifth

Table E.1 Stop Words List (Continued)

finally
first
five
followed
following
follows
for
former
formerly
forth
four
friday
from
fully
further
furthermore
g
get
gets
getting
given
gives
go
goes
going
gone
got
gotten
greetings
h
had
happens
hardly
has
have
having
he
hello
help
hence
her
here
Hereafter
Hereby
Herein
Hereupon
Hers
Herself

hi
highly
him
himself
his
hither
hopefully
how
howbeit
however
i
i.e
i.e.
ideally
ie
if
ignored
immediate
impossible
in
inasmuch
inc
include
includes
indeed
indicate
indicated
indicates
inner
insofar
instead
into
inward
is
it
its
itself
j
jan
january
Jul
July
Jun
June
Just
K
keep
keeps

Kept
know
known
knows
L
larger
largest
Last
lately
Later
latter
latterly
least
Less
Lest
Let
like
liked
likely
little
look
looking
looks
ltd
m
mainly
many
mar
march
may
maybe
me
mean
meanwhile
merely
might
monday
more
moreover
most
mostly
much
must
my
myself
N
Name
namely

nd
near
nearly
necessary
need
needs
neither
never
nevertheless
new
next
nine
no
nobody
non
none
noone
nor
normally
not
nothing
nov
novel
november
now
nowhere
o
obviously
oct
october
of
off
often
oh
ok
okay
old
on
once
one
ones
only
onto
or
other
others
otherwise
ought

Table E.1 Stop Words List (Continued)

our	secondly	Taken	truly
Ours	see	tell	try
Ourselves	seeing	tends	trying
Out	seem	th	tuesday
Outside	seemed	than	twice
Over	seeming	thank	two
Overall	seems	thanks	u
Own	seen	thanx	un
P	self	that	under
Particular	selves	thats	unfortunately
Particularly	sensible	the	unless
Per	sent	their	unlikely
Perhaps	sep	theirs	until
Placed	september	them	unto
Please	serious	themselves	up
Plus	seriously	then	upon
Poor	seven	thence	us
Possible	several	there	use
Presumably	shall	thereafter	used
Probably	she	thereby	useful
provides	should	therefore	uses
q	since	therein	using
que	six	theres	usually
quite	so	thereupon	uucp
qv	some	these	v
r	somebody	they	value
rarely	somehow	think	various
rather	someone	third	very
rd	something	this	via
re	sometime	thorough	viz
ready	sometimes	thoroughly	vs
really	somewhat	those	w
reasonably	somewhere	though	want
regarding	soon	three	wants
regardless	sorry	through	was
regards	specified	throughout	way
relatively	specify	thru	we
respectively	specifying	thursday	wednesday
right	still	thus	welcome
s	sub	to	well
said	successful	today	went
same	successfully	together	Were
saturday	such	too	What
saw	sunday	took	whatever
say	sup	toward	when
saying	sure	towards	whence
says	t	tried	whenever
second	take	tries	where

Table E.1 Stop Words List (Continued)

whereafter	whither	wish	yes
whereas	who	with	yet
whereby	whoever	within	you
wherein	whole	without	your
whereupon	whom	wonder	yours
wherever	Whose	would	yourself
whether	Why	wrong	yourselves
which	Will	X	z
while	Willing	Y	zero

REFERENCES

Ackerman, M. S. (1994). Augmenting the organizational memory: A field study of answer garden, in Proceedings of the Conference on Computer Support Cooperative Work (CSCW'94). Chapel Hill, NC. ACM Press, pp. 243-252.

Ackerman, M. S. and McDonald, D. W. (1996). Answer Garden 2: Merging organizational memory with collaborative help, in Proceedings of the ACM Conference on Computer Supported Cooperative Work (CSCW'96), pp. 97-105.

Adamic, L. A. and Adar, E. (2003). Friends and Neighbors on the Web. Social Networks. 25(3), pp. 211-230.

Alkamha, R. and Embley, D. W. (2004). Grouping Search Engine Returned Citations for Person Name Queries, in Proceedings of the Sixth ACM International Workshop on Web Information and Data Management (WIDM 2004), November 12-13, Washington, DC, USA.

Bachelor, J. and Eaton, E. A. (1980). The combined use of bibliographic coupling and co-citation for document retrieval, JASIST, 31(7), pp. 278-282.

Bates, M. J., & Lu, S. (1997). An exploratory profile of personal home pages: content, design, metaphors. Online & CD ROM Review, 21(6), pp. 331-340.

Bekkerman, R. and McCallum, A. (2005). Disambiguating Web appearances of people in a social network, in Proceedings of WWW '05, Chiba, Japan..

Belkin, N. and Croft, W. (1987). Retrieval techniques. In M. Williams, editor, Annual Review of Information Science and Technology (ARIST), Elsevier Science Publishers, 22(4), pp. 109-145.

Bharat, H. and Henzinger, M. R. (1998). Improved algorithms for topic distillation in hyperlinked environments, in Proceedings of the 21st International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 104-111.

Bollacker, K. D., Lawrence, S, and Giles, C. L. (1998). Citeseer: an autonomous web agent for automatic retrieval and identification of interesting publications, in Proceedings of the 2Second International Conference on Autonomous agents, pp. 116-123.

Bollmann-Sdorra, P. and Raghavan, V. (1993). On the Delusiveness of Adopting a Common Space for Modeling IR Objects: Are Queries documents? Journal of the American society for information and technology (JASIST), 44(10), pp. 320-330.

Borodin, A., Roberts, G. O., Rosenthal, J. S. and Tsaparas, P. (2001). Finding authorities and hubs from link structures on the World Wide Web, in Proceedings of the 10th International World Wide Web Conference (WWW'01), pp. 415-429.

Brauen, T. L. (1969). Document Vector Modification. Scientific Report ISR-17.

Brin, S. and Page, L. (1998). The anatomy of a large-scale hyper textual web search engine, in Proceedings of the Seventh International World Wide Web Conference, Brisbane, Australia.

Budzik, J., Bradshaw, S., Fu, X., and Hammond, K. J. (2002). Clustering for opportunistic communication, in Proceedings of the International WWW Conference. Honolulu, HA.

Buten, J. (1996). First world wide web personal home page survey. http://www.nicoladoering.de/Hogrefe/buten/www.asc.upenn.edu/usr/sbuten/survey2.htm

Calvo, R. A., Lee, J. M. and Li, X. (2004). Managing Content with Automatic Document Classification, Journal of Digital Information, 5(2), Article No. 282.

Chandler, D. and Roberts-Young, D. (1999). The Construction of Identity in the Personal Homepages of Adolescents, in Proceedings of IN-TELE 98 - European Conference on Educational Uses of the Internet and European Identity Construction.

Cohen, D., Jacovi, M.,Maarek, Y. S., and Soroka, V. (2002). Livemaps for collection awareness, International Journal of Human-Computer Study, 56(1), pp. 7-23.

De Saint-Georges, I. (1997). Click here if you want to know who I am: Deixis in personal homepages, in Proceedings of the Hawaii International Conference on Systems Science.

Dean, J. and Henzinger, M. R. (1999). Finding related web pages in the World Wide Web, in Proceedings the 8th International World Wide Web Conference (WWW8), pp. 389-401.

Deshpande, M. and Karypis, G. (2004). Item-based top-n recommendation algorithms, ACM Transactions on Information Systems, 22(1), pp. 1-34.

Dillon, A., and Gushrowski, B. (2000). Genres and the web: is the personal home page the first uniquely digital genre? Journal of the American Society for Information Science, 51(2), pp. 202-205.

Dominick, J. R. (1999). Who do you think you are? Personal home pages and self-presentation on the WWW, Journalism Quarterly, 76(4), pp. 646-658.

Doring, N. (2002). Personal Home Pages on the Web: A Review of Research, Journal of Computer-Mediated Communication, 7(3).

Erickson, T. (1996). The World Wide Web as social hypertext, Communications of the ACM, 39(1), pp. 15-17.

Fellbaum, C. (1998). WordNet: An Electronic Lexical Database. Cambridge, MIT Press.

Fleischman, M. B. and Hovy, E. (2004). Multi-Document Person Name Resolution, in Proceedings of the 42nd Annual Meeting of the Association for Computational Linguistics, Barcelona, Spain.

Fogaras, D. and Racz, B. (2005). Scaling Link Based Similarity Search, in Proceedings of WWW 2005, Chiba, Japan.

Goldberg, D. E. (1989). Genetic Algorithms in Search, Optimization, and Machine Learning, Addison-Wesley Professional, Boston, MA.

Goodman, L. A. and Kruskal, W. H. (1954). Measures of association for cross-classifications, Journal of America Statistics Association, 49, pp. 732-764.

Google, (1999). Google's New GoogleScout Feature Expands Scope of Search on the Internet, http://www.google.com/press/pressrel/pressrelease4.html

Google, (2005a). Preventing comment spam, http://googleblog.blogspot.com/2005/01/preventing-comment-spam.html

Google, (2005b). Google bombing failure, http://googleblog.blogspot.com/2005/09/googlebombing-failure.html

Holland, J. H. (1975). Adaptation in natural and artificial systems. Ann Arbor, Michigan, University of Michigan Press.

Han, H., Giles, H., Zha, H., Li, C., and Tsioutsiouliklis, K. (2004). Two Supervised Learning Approaches for Name Disambiguation in Author Citations, in Proceedings of the Joint Conference on Digital Libraries, Tucson, Arizona.

Haveliwala, T. H., Gionis, A., Klein, D. and Indyk, P. (2002). Evaluating Strategies for Similarity Search on the Web, in Proceedings of WWW2002, Honolulu, Hawaii, USA.

Heaps, H. S. (1978). Information retrieval, computational and theoretical aspects. Academic Press.

Henzinger, M. (2000). Link analysis in web information retrieval. Bulletin of the Technical Committee on Data Engineering, IEEE Computer Society, pp. 3-8.

Heylighen F. (2001). Collaborative Filtering, Principia Cybernetic Project report http://pespmc1.vub.ac.be/http://%EF/COLLFILT.html.

Hill, W. C. and Terveen, L. G. (1996). Using frequency-of-mention in public conversations for social filtering, in Proceedings of the ACM Conference on Computer Supported Cooperative Work (CSCW'96), Boston, MA. pp. 106-112.

Jeh, G. and Simrank, J. M. (2002). A measure of structural-context similarity, in Proceedings of the Eighth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Edmonton, Alberta, Canada.

Jin, R. and Dumais, S. (2001). Probabilistic combination of content and links, in Proceedings of the Annual International ACM/SIGIR Conference on Research and Development in Information Retrieval, New Orleans, LA.

Kamp, H. A. (1981). Theory of Truth and Semantic Representation, Formal Methods in the Study of Language (J. Groenendijk, T. Janssen, and M. Stokhof Eds.), 1, Mathema-tische Centrum.

Kantrowitz, M., Mohit, B. and Mittal, V. (2000). Stemming and its effects on TFIDF ranking, in Proceedings of the Annual ACM Conference on Research and Development in Information Retrieval, New York, NY, USA, pp. 357-359.

Karlsson, M. (1998). Selves, frames and functions of two Swedish teenagers' personal home pages. in Proceedings of the Sixth International Pragmatics Conference, Reims/Frankreich. Available: http://www.nordiska.su.se/personal/karlsson-a-m/ipra.htm

Kautz, H., Selman, B., and Shan, M. (1997). ReferralWeb: Combining social networks and collaborative filtering, Communication of. ACM, 30(3).

Kessler, M. (1963). Bibliographic coupling between scientific papers, American Documentation, 14, pp. 10-25

Kleinberg, J. M. (1999). Authoritative sources in a hyperlinked environment, Journal of the ACM, 46(5), pp. 604-632.

Korfhage, R. R. (1997). Information Storage and Retrieval. Wiley Computer Publishing.

Kucera, H. and Francis, W. N. (1967). Computational Analysis of Present-Day American English, Brown University Press, Providence, Rhode Island.

Lee, D. L., Chuang, H. and Seamons, K. (1997). Document ranking and the vector-space model, IEEE Software, 14(2), pp. 67-75.

Lempel, R. and Moran, S. (2000). The stochastic approach for link-structure analysis (SALSA) and the TKC effect, Computer Networks, 33(6), pp. 387-401.

Letsche, T. and Berry, M. (1997). Large-scale information retrieval with latent semantic indexing, Information sciences, 100(1), pp. 105-137.

Li, Q., Wu, Y. B., Bot, R. S., Chen, X. (2004). Incorporating Document Keyphrases in Search Results, in Proceedings of the Tenth Americas Conference on Information Systems, New York, New York.

Linden, G, Smith, B. and York, J. (2003). Amazon.com Recommendations, http://hugo.csie.ntu.edu.tw/~yjhsu/courses/u2010/papers/Amazon%20Recommendations.pdf

Liu, W., Huang, G., Liu, X., Zhang, M., Deng, X. (2005). Detection of Phishing Webpages based on Visual Similarity, in Proceedings of the 14th International World Wide Web Conference (WWW'05), Chiba, Japan, pp. 1060-1061.

Lovins, J. B. (1968). Development of a stemming algorithm. Mechanical Translation and Computational linguistic, 11, pp. 22-31.

Luhn, H. P. (1958). The Automatic Creation of Literature Abstracts, IBM Journal of Research and Development, 2(2), pp. 159-165.

McDonald, D. W. and Ackerman, M. S. (2000). Expertise recommender: A flexible recommendation architecture, in Proceedings of the ACM Conference on Computer-Supported Cooperative Work (CSCW'00), pp. 231-240.

McDonald, D. W. (2001). Evaluating expertise recommendations., in Proceedings of the International Conference on Supporting Group Work (GROUP'01).

Menczer, F. (20040. Combining link and content analysis to estimate semantic similarity, in Proceedings of WWW2004, New York, New York.

Miller, H. (1995). The presentation of self in electronic life: Goffman on the Internet. Paper presented at Embodied Knowledge and Virtual Space conference, London. Available: http://ess.ntu.ac.uk/miller/cyberpsych/goffman.htm

Mobasher, B., Jin, X. and Zhou, Y. (2004). Semantically Enhanced Collaborative Filtering on the Web, Web Mining: From Web to Semantic Web, Bettina Berendt et al. (eds.), LNAI, Springer.

Narsesian, S. (2004). Personal home pages as an information resource, Webology, 1(2)

Nichols, D. M. (1998). Implicit Rating and Filtering, in Proceeding of the Fifth DELOS Workshop on Filtering and Collaborative Filtering, Budapest, Hungary, pp. 31-36.

Page, L. Brin, S., Motwani, R. Winograd T. (1998). The PageRank Citation Ranking: Bringing Order to the Web, Stanford Digital Library Technologies Project, http://citeseer.ist.psu.edu/page98pagerank.html

Papacharissi, Z. (2002a). The self online: the utility of personal home pages, Journal of Broadcasting & Electronic Media, 46(3), pp. 346-368.

Papacharissi, Z. (2002b). The presentation of self in virtual life: characteristics of personal home pages, Journalism and Mass Communication Quarterly, 79(3), pp. 643-660.

Plu, M., Bellec, P. and Agosto, L. (2002). The Web of people: A Dual View on the WWW, in Proceedings of WWW'02.

Porter, M. F. (1980). An algorithm for suffix stripping, Journal of Program, 14, pp. 130-137.

Raghavan, V. V. and Wong, S. K. M., (1986). A critical analysis of vector space model for information retrieval, Journal of the American Society for Information Science, 37(5), pp. 279-87.

Rijsbergen, J. (1979). Information retrieval, 2nd edition. London, Butterworths.

Rui, Y., Huang, T. S. and Chang, S. F. (1997). Image Retrieval: Past, Present, And Future, in Proceeding of International Symposium on Multimedia Information Processing.

Salton, G. (1983). Introduction to Modern Information Retrieval. McGraw-Hill.

Salton, G. (1989). Automatic Text Processing: The Transformation, Analysis and Retrieval of Information by Computer. Addison-Wesley.

Salton, G., Wong, A. and Yang, C. S. (1975). A vector space model for automatic indexing, Communications of the ACM, 18, pp. 613-620.

Salton, G. and Buckley, C. (1996). Term Weighting Approaches in Automatic Text Retrieval, Information Processing and Management, 32(4), pp. 431-443.

Sarwar, B., Karypis, G., Konstan, J. and Riedl, J. (2001). Item-based collaborative filtering recommendation algorithms, in Proceedings of the Tenth International WWW Conference, Hong Kong.

Shakes, J., Langheinrich, M. and Etzioni, O. (1997). Dynamic reference sifting: a case study in the homepage domain, in Proceedings of WWW'97, pp. 189-200.

Sherman, C. (2005). Yahoo offers new contextual search tools, http://searchenginewatch.com/searchday/article.php/3467911, Search Engine Watch.

Siegel, S. and Castellan, N. J. (1988). Nonparametric Statistics for the Behavioral Sciences. McGraw-Hill.

Small, H. (1973). Co-citation in the scientific literature; a new measure of the relationship between two documents, Journal of the American Society for Information Science, 24, pp. 265-269.

Snow, C. E. and Ferguson, C. A. (1997). Talking to Children: Language Input and Acquisition, Cambridge, Cambridge University Press.

Svensson, M., Hook, K., Laaksahoti, J., and Waern, A. (2001). Social navigation of food recipes, in Proceedings of Computer Human Interaction (CHI'01), pp. 341-348.

Terveen, L. G., Hill, W., Amento, B., McDonald, D., and Creter, J. (1997). PHOAKS: A system for sharing recommendations, Communication of ACM, 40(3), pp. 59-62.

Terveen, L. and McDonald, D. W. (2005). Social matching: a framework and research agenda, ACM transaction on Computer-Human Interaction, 12(3), pp. 401-434.

Terveen, L. G., Selfridge, P. G., and Long, M. D. (1995). Living design memory: Framework, implementation, lessons learned, Human-Computer Interaction, 10(1), pp. 1-38.

Turney, P. D. (2000). Learning algorithm for keyphrase extraction, Information Retrieval 2(4), pp. 303-336.

Vazire, S. and Gosling, S. D. (2004). e-Perceptions: Personality Impressions Based on Personal Websites, Journal of Personality and Social Psychology, 87(1), pp. 123-132.

Walther, J. (1996). Computer-mediated communication: Impersonal, interpersonal, and hyperpersonal interaction, Communication Research, 23, pp. 3-43.

Wan, X., Gao, J., Li, M. and Ding, B. (2005). Person resolution in person search results: WebHawk, in Proceedings of International Conference for Information and Knowledge Management (CIKM'05), Bremen, Germany, pp. 163-170.

Weaver, A. E. (2000). Personal web pages as professional activities: an exploratory study, Reference Services Review, 28(2), pp. 171-177.

White, H. D. and Griffith, B. C. (1980). Author co-citation: A literature measure of intellectual structure, Journal of the American society for information and technology (JASIST), 31(3), pp. 163-171.

Whitely, D., (1989). The Genitor algorithm and selective pressure, in Proceedings of the third international conference on Genetic Algorithms, CA, pp. 116-121.

Wilkinson, R. and Hingston P. (1991). Using the cosine measure in a neural network for document retrieval, in Proceedings of the 14th Annual International ACM/SIGIR Conference on Research and Development in Information Retrieval, Chicago, pp. 202-210.

Wu, Y. B., Li, Q. Bot, R. and Chen, X. (2006). Finding nuggets in documents: A machine learning approach, Journal of the American society for information and technology (JASIST), 57(6), pp. 740-752.

Wynn, E., & Katz, J. E. (1997). Hyperbole over cyberspace: Self-presentation and social boundaries in home pages and discourse. The Information Society, 13(4), pp. 297-328. Available at: http://www.slis.indiana.edu/TIS/articles/hyperbole.html

Yahoo. (2005). Y!Q, search in context, http://yq.search.yahoo.com/publisher/faq.html.

Yang, J. J. and Korfhage R. R. (1992). Adaptive information retrieval systems in vector model, Symposium on document analysis and information retrieval, Las Vegas, NV, pp.134-150.

Zhang, P., Benbasat, I., Carey, J., Davis, F., Galletta, D. and Strong, D. (2002). Human-Computer Interaction Research in the MIS Discipline, Communications of the AIS, 9(20), pp. 334-355

Zobel, J. and Moffat, A. (1998). Exploring the Similarity Space, in Proceedings of the 21st Annual International ACM/SIGIR Conference on Research and Development in Information Retrieval, Melbourne, Australia.