IDENTIFYING EXPERTS AND AUTHORITATIVE DOCUMENTS IN SOCIAL BOOKMARKING SYSTEMS

by

Jonathan P. Grady

B.A. Economics & Business Administration, Ursinus College, 1997

M.S. Electronic Commerce, Carnegie Mellon Univ., 2001

Submitted to the Graduate Faculty of

School of Information Sciences in partial fulfillment

of the requirements for the degree of

Doctor of Philosophy

University of Pittsburgh

2013

UNIVERSITY OF PITTSBURGH

SCHOOL OF INFORMATION SCIENCES

This dissertation was presented

by

Jonathan P. Grady

It was defended on

March 27, 2013

and approved by

Peter Brusilovsky, Ph.D., Professor, School of Information Sciences

Daqing He, Ph.D., Associate Professor, School of Information Sciences

Stephen C. Hirtle, Ph.D., Professor, School of Information Sciences

Brian S. Butler, Ph.D., Associate Professor, College of Information Studies,

University of Maryland

Dissertation Advisor: Michael B. Spring, Ph.D., Associate Professor,

School of Information Sciences

IDENTIFYING EXPERTS AND AUTHORITATIVE DOCUMENTS IN SOCIAL BOOKMARKING SYSTEMS

Jonathan P. Grady

University of Pittsburgh, 2013

Social bookmarking systems allow people to create pointers to Web resources in a shared, Web-based environment. These services allow users to add free-text labels, or "tags", to their bookmarks as a way to organize resources for later recall. Ease-of-use, low cognitive barriers, and a lack of controlled vocabulary have allowed social bookmaking systems to grow exponentially over time. However, these same characteristics also raise concerns. Tags lack the formality of traditional classificatory metadata and suffer from the same vocabulary problems as full-text search engines. It is unclear how many valuable resources are untagged or tagged with noisy, irrelevant tags. With few restrictions to entry, annotation spamming adds noise to public social bookmarking systems. Furthermore, many algorithms for discovering semantic relations among tags do not scale to the Web.

Recognizing these problems, we develop a novel graph-based Expert and Authoritative Resource Location (EARL) algorithm to find the most authoritative documents and expert users on a given topic in a social bookmarking system. In EARL's first phase, we reduce noise in a Delicious dataset by isolating a smaller sub-network of "candidate experts", users whose tagging behavior shows potential domain *and* classification expertise. In the second phase, a HITS-based graph analysis is performed on the candidate experts' data to rank the top experts and authoritative documents by topic. To identify topics of interest in Delicious, we develop a

distributed method to find subsets of frequently co-occurring tags shared by many candidate experts.

We evaluated EARL's ability to locate authoritative resources and domain experts in Delicious by conducting two independent experiments. The first experiment relies on human judges' *n*-point scale ratings of resources suggested by three graph-based algorithms and Google. The second experiment evaluated the proposed approach's ability to identify classification expertise through human judges' *n*-point scale ratings of classification terms versus expert-generated data.

TABLE OF CONTENTS

PREFACE XVI

1.0 INTRODUCTION 1

1.1 FOCUS OF STUDY 3

1.2 DELICIOUS 6

1.3 LIMITATIONS AND DELIMITATIONS 8

1.4 DEFINITION OF TERMS 10

1.4.1 Annotation 10

1.4.2 Bookmark 10

1.4.3 Metadata 11

1.4.4 Social Annotation 11

1.4.5 Social Bookmark 11

1.4.6 Tag 11

1.4.7 Resource 12

1.4.8 URL 12

1.4.9 Taxonomy 13

1.4.10 Folksonomy 13

1.4.11 Noise 14

1.4.12 Power set 14

2.0 RELATED WORK 15
2.1 BACKGROUND 15
2.1.1 Annotations 15
2.1.1.1 Bookmarks 18
2.1.1.2 Social Bookmarks 19
2.1.2 Classification 24
2.1.2.1 Approaches to Categorization 26
2.1.2.2 Classification Structures 27
2.1.2.3 Subject Analysis - Classification by Experts 32
2.1.3 Domain Expertise 34
2.2 IDENTIFYING EXPERT USERS IN WEB-BASED SYSTEMS 39
2.2.1 Identifying Expert Users in Bipartite Graphs 39
2.2.2 Identifying Expert Users in Tripartite Graphs 42
2.3 CLASSIFICATION IN SOCIAL ANNOTATION SYSTEMS 45
3.0 PRELIMINARY ANALYSIS 48
3.1 INTRODUCTION 48
3.2 SOCIAL BOOKMARKING SYSTEMS 49
3.2.1 Usage Patterns 50
3.2.2 Topics of Interest in Social Bookmarking Systems 53
3.3 FINDING EXPERTS AND AUTHORITATIVE RESOURCES 54
3.3.1 Defining experts and authoritative resources 54
3.3.2 EARL algorithm 56
3.3.3 Selecting topics of interest 63

3.4 FINDING EXPERTS AND AUTHORITATIVE RESOURCES 68
3.4.1 Candidate Expert Tagging Patterns 68
3.4.2 Topics of Interest 73
3.4.3 EARL versus HITS and SPEAR 77
4.0 RESEARCH DESIGN 81
4.1 DELICIOUS DATA 81
4.2 PRE-PROCESSING OF DATA 84
4.3 METHODOLOGY 86
4.4 EXPERIMENT 1: EVALUATING EARL'S ABILITY TO LOCATE AUTHORITATIVE RESOURCES 87
4.4.1 Participants 87
4.4.2 Variables and Expected Results 88
4.4.3 Hypotheses of the 1st Experiment 89
4.4.4 Subjects, Evaluation, and Analysis Procedure 89
4.5 EXPERIMENT 2: EVALUATING EARL'S ABILITY TO LOCATE DOMAIN EXPERTS 91
4.5.1 Participants 92
4.5.2 Variables and Expected Results 92
4.5.3 Hypotheses of the 2nd Experiment 93
4.5.4 Subjects, Evaluation, and Analysis Procedure 94
4.6 EXPERIMENT 3: EVALUATING TOPICS OF INTEREST TO LOCATE CLASSIFICATION EXPERTS 96
4.6.1 Experimental Data 96

4.6.2 Participants 100
4.6.3 Variables and Expected Results 101
4.6.4 Hypotheses of the 3rd Experiment 102
4.6.5 Subjects, Evaluation, and Analysis Procedure 102
5.0 RESULTS 105
5.1 QUESTIONS USED IN EXPERIMENTS 1 & 2 105
5.2 ASSESSMENT OF THE SUBJECTS' RELEVANCY RATINGS 107
5.2.1 Inter-rater Reliability for Experiments 1 & 2 108
5.2.2 Inter-rater Reliability for Experiment 3 109
5.3 EXPERIMENT 1: RANKING OF AUTHORITATIVE DOCUMENTS .. 110
5.3.1 Analysis of Entry Questionnaire Responses 111
5.3.2 Analysis of Authoritative Document Rankings by Algorithm & Dataset 112
5.4 EXPERIMENT 2: RANKING OF DOMAIN EXPERTS 116
5.4.1 Analysis of Domain Expert Rankings by Algorithm & Dataset: Average Ratings 117
5.4.2 Analysis of Domain Expert Rankings by Algorithm & Dataset: % of Highly-Rated Resources 119
5.5 EXPERIMENT 3: CLASSIFICATION EXPERTISE AND RANKING OF TOPICS OF INTEREST 124
5.5.1 Analysis of Entry Questionnaire Response 125
5.5.2 Analysis of Subjects' Ratings of Classificatory Terms: nDCG 126

5.5.3 Analysis of High-quality Tag Use by Candidate Experts vs. Average Delicious Users 130
5.6 DISCUSSION OF RESULTS 132
5.6.1 Authoritative Resource Rankings 132
5.6.2 Domain Expert Rankings 135
5.6.3 Classification Expertise and Rankings of Topics of Interest 136
6.0 CONCLUSION 138
6.1 CONTRIBUTIONS & IMPLICATIONS 138
6.2 FUTURE WORK 141
APPENDIX A 144
APPENDIX B 154
APPENDIX C 157
BIBLIOGRAPHY 160

LIST OF TABLES

Table 1. Marshall's dimensions of annotations (1998) 17

Table 2. Types of Classification Schemes (from Fettke & Loos, 2002)..... 25

Table 3. An example of a faceted classification of a hypothetical book on 17th century Norwegian architecture using Ranganathan's Colon Classification (reproduced from Garshol, 2004.) 29

Table 4. Langridge's steps in the conceptual analysis phase of subject analysis (reproduced from Appendix 3 of Langridge, 1989.)..... 32

Table 5. Ericsson's list of general theoretical frameworks of domain expertise (2006) 35

Table 6. Chi's list of domain experts' strengths and shortcomings (Chi, 2006)..... 37

Table 7. Noll et al.'s classification of experts and spammers in a social bookmarking system.... 44

Table 8. An example illustrating the calculation of the temporal sequence portion of EARL's weight, factoring in daily bursts of activity. 62

Table 9. The power set elements for a bookmark tag set consisting of the tags "css", "webdesign", and "tips" 64

Table 10. Basic statistics for the preliminary main and candidate expert datasets..... 69

Table 11. Comparison of the top seven tags, frequencies, and usage percentage in the preliminary candidate expert dataset versus the main dataset for three popular resources in Delicious. 71

Table 12. All user versus candidate expert bookmark contributions to resources in the preliminary main dataset with complete histories and Ó200 bookmarks (*n* = 1,678.)..... 72

Table 13. Top 40 topics of interest of candidate experts 76

Table 14. Comparison of HITS', SPEAR's, and EARL's rankings of the top 10 experts and resources in the candidate expert dataset for the topic “design, web”..... 78

Table 15. Comparison of HITS', SPEAR's, and EARL's rankings of the top 10 experts and resources in the candidate expert dataset for the topic “rest webservices”..... 79

Table 16. Basic statistics for the main and candidate expert datasets 83

Table 17. Independent variables and conditions in the first experiment. Each subject ranks results lists from all seven conditions..... 88

Table 18. Independent variables and conditions in the second experiment. Each subject ranks domain expert data from all six conditions..... 92

Table 19. List of resources selected for Experiment 3..... 97

Table 20. The top 20 subsets of frequently co-occurring tags from candidate experts’ bookmarks on http://www.gazelle.com, as identified by the topic of interest process described in Section 3.3.3..... 100

Table 21. List of questions used in Experiments 1 and 2. 106

LIST OF FIGURES

Figure 1. An example of a resource bookmarked on Delicious..... 7

Figure 2. A topic map describing topic maps. Large shapes are topics. Arrows denote relations. The paper icon in the top left corner represents and occurrence (i.e. resource) of topic maps (reproduced from Garshol, 2004.) 31

Figure 3. Frequency-rank distribution of the number of bookmarks per user for all users in the preliminary main dataset. 50

Figure 4. Frequency-rank distribution of the number of tags per bookmark for bookmarks in the preliminary main dataset. 51

Figure 5. Dellschaft and Staab's (2008) comparison between the actual frequency-rank distribution of tags on the NetVibes home page (shown in grey), versus simulated tag stream models (dashed and solid lines) assuming users see the top 7 most popular tags as they enter their own tags. 52

Figure 6. A partial view of a social bookmarking system as a graph. All edges (i.e. bookmarks) are directed from users to resources. 57

Figure 7. Pseudocode for the second stage of EARL. 58

Figure 8. Outlink and inlink adjacency lists used in EARL. 60

Figure 9. Pseudocode for EARL's topic selection approach. 66

Figure 10. Frequency-rank distribution of topics of interest in the preliminary candidate expert dataset. Each topic listed on the horizontal axis represents a decrease of 10,000 in rank position. 74

Figure 11. Experiment 1's user interface. 90

Figure 12. Example of extracting relevant, shared topics of interest from candidate experts' bookmarks of the Google homepage. 99

Figure 13. Experiment 3's user interface. 103

Figure 14. The results of two-way between-subjects ANOVA for Experiment 1 113

Figure 15. The means and standard deviations of nDCG10 for Experiment 1 (n=833.) 114

Figure 16. Comparisons to find significant differences in $nDCG_{10}$ among the ranking algorithms and use of filtering procedure, respectively. 115

Figure 17. The results of the two-way between-subjects ANOVA for Experiment 2, mean ratings of candidate experts top bookmarked resources. 117

Figure 18. The means and standard deviations of $nDCG_{10}$ of candidate expert rankings for Experiment 2, mean ratings of candidate experts' top bookmarked resources 118

Figure 19. The means and standard deviations of the $nDCG_{10}$ of candidate expert rankings for Experiment 2, percentage of highly-rated resources bookmarked (n=714.) 120

Figure 20. The results of the two-way between-subjects ANOVA for Experiment 2, percentage of highly-rated resources bookmarked 121

Figure 21. Comparisons to find significant differences in $nDCG_{10}$ of the candidate expert rankings among the ranking algorithms. 122

Figure 22. Comparisons to find significant differences in the $nDCG_{10}$ of the candidate expert rankings of the main dataset (no filtering procedure) versus the candidate expert dataset (filtering procedure used) for each ranking algorithm. 123

Figure 23. Average Ratings of Self-Assessment Questions (*n*=20) 126

Figure 24. The results of the one-way ANOVA for Experiment 3, means of the $nDCG_{10}$ of the four methods to rank topics of interest. 127

Figure 25. The means and standard deviations of the $nDCG_{10}$ of the topic of interest rankings, Experiment 3 (*n*=100.) 128

Figure 26. Comparisons to find significant differences in $nDCG_{10}$ of the topic of interest rankings among the four conditions. 129

Figure 27. The results of the one-way ANOVA for Experiment 3, percentages of high-quality tag use by candidate experts and average Delicious users..... 130

Figure 28. The means and standard deviations of the $nDCG_{10}$ of the topic of interest rankings, Experiment 3 (*n*=240.) 131

PREFACE

I express my deepest thanks to my advisor and dissertation committee chair, Dr. Michael Spring for his generous help and support over the years in both my research and teaching endeavors. This dissertation would not have been possible without his guidance.

I thank my dissertation committee members - Professors Peter Brusilovsky, Brian Butler, Daqing He, and Stephen Hirtle - or whose comments and suggestions I greatly appreciate. I also thank my colleagues Jon Walker, Sue Yeon Syn, Sung-Min Kim, and Alawya Alawami for their helpful ideas, opinions, and critiques throughout my dissertation research.

I wish to acknowledge Drs. Armando Rotondi and Jennifer Steel for their financial support, as well as the opportunities they afforded me to gain invaluable research experience over the course of my studies.

Finally, a special thank you to my parents, Jane and Fred; my brother, Tim; and my sister-in-law, Malina, for their constant support and encouragement.

1.0 INTRODUCTION

Full-text search engines have become the most popular means of locating information on the Web. Despite their popularity, search engines still suffer from well-known vocabulary problems (i.e., synonymy and polysemy), a lack of well-defined topics or relations, and an increasing amount of noise on the Web. In February, 2011, Google announced changes to its algorithm intended to lower the search result rankings of content farms – mass producers of low-quality content designed to match users' queries (New York Times, 2011.) Google's changes came in response to complaints of irrelevant pages at the top of search results for some queries, illustrating how noise can prevent users from finding useful information on the Web.

Prior to the rise of search engines, human-edited taxonomies of Web resources, such as Yahoo!'s Web directory, were the tools of choice for Web retrieval because of the more precise classification they offered. Directory services like Yahoo! and the Open Directory Project (ODP) continue to be maintained by human editors, but the growth and churn of the Web has made such services too inefficient to maintain.

Berners-Lee, Hendler and Lassila (2001) envisioned a Semantic Web that would address the limitations of HTML and supplant the need for full-text indexing. Documents on the Web would be structured and marked up semantically, allowing machines to analyze and understand their contents. Metadata describing the resources would be written in a standard structured language with standard vocabularies, such as the XML-based Resource Description Framework

(RDF). Ontologies – shared conceptualizations of a domain (Gruber, 2003) – would formally describe the concepts and relationships in knowledge domains, linking resources together and allowing machines to make inferences through the expressed relationships.

Ten years after Berners-Lee, Hendler and Lassila expressed their vision, little of the Semantic Web has actually been built. Generating usable metadata either manually or automatically on the scale of the Web has proven to be an elusive goal. Too many resources and not enough expert human metadata generators exist to perform the necessary annotations. Ontologies built for the Semantic Web become increasingly difficult to maintain as they grow. Agreement upon a single ontology for a domain (let alone the entire Semantic Web) is not feasible in a distributed environment such as the Web (Kalfoglou and Schorlemmer, 2003). Ontology mapping has the potential to solve the single-ontology dilemma, but research in the area is only beginning.

Social bookmarking systems, such as Delicious, CiteULike, and Digg, allow people to create pointers to Web resources in a shared, Web-based environment. These services also allow users to add free-text labels, or "tags", to their bookmarks as a way to classify, organize, and recall resources at a later date. Their ease-of-use, low cognitive barriers, and lack of controlled vocabulary have allowed them to grow exponentially in a matter of a few years. With a wealth of metadata now available on millions of Web resources, researchers are examining ways to use tags on social bookmarks to build classification schemes.

Social bookmarking systems are not without their problems. Without a controlled vocabulary, tags lack the formality of traditional classificatory metadata and suffer from the same vocabulary problems - synonymy and polysemy - as full-text search engines. Despite the growing popularity of social bookmarking systems, the number of annotated resources is a small

fraction of the resources on the web. It is unclear how many valuable resources are untagged, making low annotation coverage a concern. Algorithms for discovering semantic relations among tags work well on small data sets, but do not scale to the Web (Bao et al., 2007.) Annotation spamming (akin to link spamming) is another issue that future research must address (Bao et al, 2007; Hotho et al, 2008; Noll et al, 2009.)

Given these problems with social bookmarking systems, this research introduces and evaluates a novel graph-based Expert and Authoritative Resource Location algorithm (EARL) to find the most authoritative documents and expert users on a given topic in a social bookmarking system. In the first phase of EARL, we reduce the Delicious data to a smaller sub-network of "candidate experts", users whose tagging behavior shows potential domain *and* classification expertise. In the second phase, we perform a HITS-based graph analysis on the candidate experts' data to rank the top experts and authoritative documents by topic. To identify topics of interest in Delicious, we develop and use a distributed method to find subsets of frequently co-occurring tags among the candidate expert users' bookmarks.

1.1 FOCUS OF STUDY

Social bookmarking is the process of users saving pointers (i.e. social bookmarks) to Web-based resources in a shared, online environment, then providing annotations to those resources to facilitate later recall, or "personal re-discovery" (Trant, 2009). Due to their relative ease-of-use, social annotation systems such as Delicious, CiteULike, and Flickr now contain annotated bookmarks to tens of millions of Web resources provided my millions of users. Although social bookmarking is fundamentally a personal endeavor, the aggregation of social annotation data

yields shared meanings of resources and powerful network effects. As a result, social annotation systems have drawn great interest from different segments of the research community. Researchers in the area of information retrieval are exploring ways to use social annotation data to improve indexing and retrieval for Web-based search. Others have analyzed the dynamics of the social annotation systems themselves, exploring network growth and usage patterns over time, identifying communities of interest, and using these shared interests to make personalized resource recommendations. Of particular interest to this work are those studies that attempt to reduce an entire graph to a small sub-sample containing the most influential nodes and edges, as well as work that removes noise from the network. Noise in the context of social bookmarking systems includes bookmarks with misleading annotations, irrelevant or potentially malicious resources (e.g., non-academic articles in CiteULike), and the users who post these annotations and resources, typically in an automated fashion. Finally, many efforts focus on the semantics of annotations, with some attempting to organize social annotations via topic maps, faceted classifications, hierarchical classifications, or lightweight ontologies to improve searching and browsing of resources.

Of the latter group of studies focusing on organizing social annotations, the majority of studies used one or more data clustering algorithms to classify tags, resources, and/or users with mixed results. Some traditional machine learning algorithms, such as self-organizing maps (SOM) and K-means clustering, produced relatively poor results due to their inability to handle polysemous tags, idiosyncratic tags, or tags that are highly-correlated with a large number of other tags (e.g. “web” in Delicious.) Other techniques that allow tags to appear in multiple clusters, such as maximal complete link clustering, produce superior classification schemes, but are too computationally expensive to scale well to large datasets. Hierarchical agglomerative (or

"bottom-up") clustering is the most promising algorithm in the literature in terms of classification quality and computational efficiency, but it is unclear how it will perform on very large samples of tag data.

In this research, we develop an algorithm to locate expert users and authoritative documents in social bookmarking systems more accurately and efficiently than existing algorithms. Because studies have shown that expertise and authoritativeness are topic dependent (Gobet & Simon, 1996; Ericsson & Lehmann, 1996; Ericsson, 2006), we also develop a method for extracting topics of interest from Delicious tag data without using machine learning algorithms. We first begin with the premise that while some proponents of folksonomies as knowledge organization structures argue that all taggers in a social annotation system are equal (Kroski, 2005; Shirky, 2005), some taggers are, in fact, more equal than others. In a preliminary analysis of a large sample of Delicious data (roughly 17 million bookmarks,) we found that 5% of the users contributed approximately 55% of the bookmarks. Furthermore, the majority of these 5% consistently annotated their bookmarks with several tags per bookmark, a rich source of annotations that shows evidence of classification expertise.

Thus, our first step is to reduce the graph around these users (i.e. influential graph nodes) using a series of simple statistics. This initial step does not provide us any clue about the semantics of the annotations, nor does it measure the quality of the resources in the personal collections. Do the annotations reflect a largely personal, idiosyncratic classification of the resources, or are there users who consistently provide annotations that accurately describe the topics of resource based on the community's consensus view? For the users and resources themselves, graph-based algorithms such as HITS and PageRank are effective tools for measuring the importance of nodes in a graph. To determine the true experts and authoritative

documents among the influential graph nodes, we use a graph-based algorithm extended from HITS to rank experts and authoritative documents by topic. We generate the topics for this study by extracting subsets of frequently co-occurring tags within and among the annotations of each expert. We extract these co-occurring tags by determining the power set (i.e., all unique subsets of a given set) of each bookmark. By finding subsets of co-occurring tags that many content experts utilize among their bookmarks, we believe we can uncover sets of tags that, despite the lack of controlled vocabulary, provide good topical descriptors for resources.

In summary, this research addresses the following questions:

- Using the judgments of independent human raters, does the EARL algorithm identify the best experts and most authoritative documents in Delicious on a given topic more *accurately* than existing algorithms, such as HITS, SPEAR, and Google's PageRank?
- Reducing the number of nodes in the Delicious data graph to a much smaller sub-network of candidate experts, does the EARL algorithm produce expert and authoritative document rankings on a given topic more *efficiently* than existing algorithms?
- Can extracting power sets from bookmark tag sets produce meaningful subsets of tags that represent users' topics of interest?

1.2 DELICIOUS

Delicious (http://delicious.com) is a social bookmarking service founded and launched by Joshua Schacter in September, 2003. Originally named and located at the domain name "del.icio.us", Delicious was acquired by Yahoo! in December, 2005, and then re-launched with a new user interface in November, 2007. Yahoo! sold Delicious to AVOS Systems in April, 2011, who re-

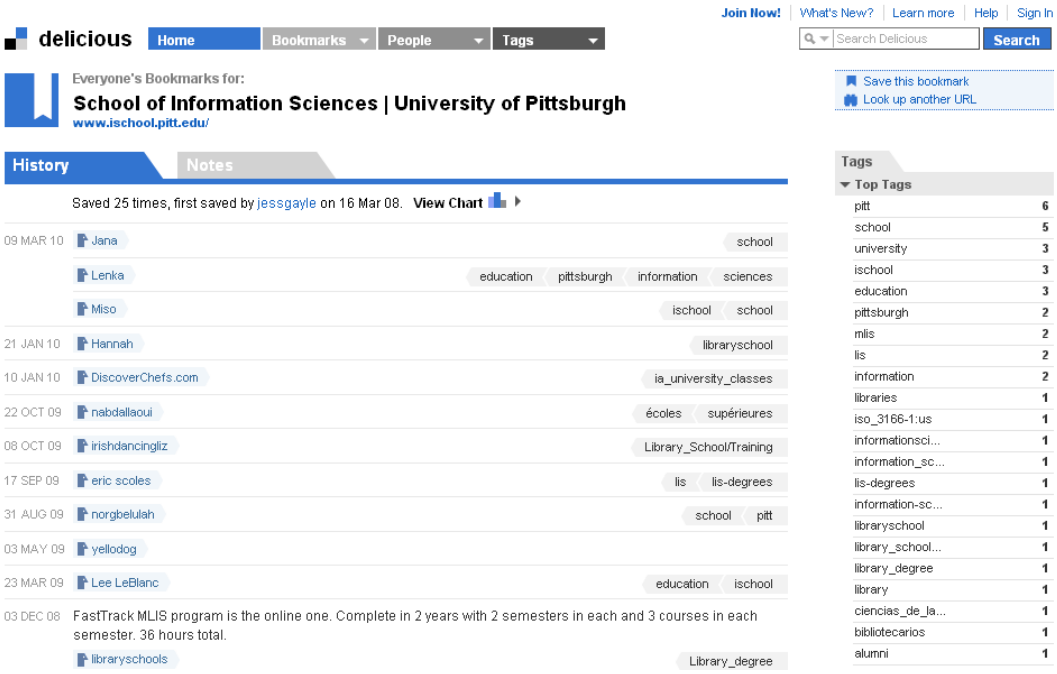


Figure 1. An example of a resource bookmarked on Delicious

launched the site with an updated interface in September, 2011. Delicious' main purpose is to provide a centralized, Web-based system for users to store, organize, and share their bookmarks from any machine with Web access. Delicious allows users to add free-text labels, i.e. "tags", to classify and organize their bookmarks. Users of, and visitors to, Delicious may freely browse the bookmarks of other users, discovering useful resources or tags via HTML or RSS feeds. Figure 1 shows an example of a resource page in Delicious as it appeared in 2010. Delicious also provides a Boolean search feature for users to search bookmarks by one or more tags.

The current number of users, resources, tags, and bookmarks on Delicious is unknown. The last official figures published by Yahoo! claimed that Delicious had 5.3 million registered users and 180 million unique URLs bookmarked (Hood, 2008.) Because Delicious is arguably the most popular general-purpose social bookmarking system available to the public, this research uses Delicious tag data collected from December, 2009 to August, 2010 (i.e. prior to the

sale to AVOS) to analyze the effectiveness of EARL and the topics-of-interest extraction method.

1.3 LIMITATIONS AND DELIMITATIONS

This study makes use of data from a single social bookmarking system, Delicious, imposing several limitations on this research:

- The results may not be generalizable to other social annotation services, or even other social bookmarking systems. Vander Wal (2005) notes the significant differences in the network structures of Delicious, a *broad folksonomy* where many users bookmark the same resources, and Flickr, a *narrow folksonomy* where most resources are bookmarked by only one user. Marlow et al. (2006) provide a taxonomy of social tagging systems, highlighting how user motivation, resource types, and tagging support can affect network structure. Santos-Neto, Ripeanu and Iamnitchi (2007) found that in CiteULike and BibSonomy a user's tab vocabulary size was positively correlated with the size of his bookmark collection, but not in Delicious.

- Despite being considered a general-purpose social bookmarking system, Delicious' content skews toward technically-oriented resources (e.g., programming, web design.) Our dataset may be missing a significant number of authoritative resources of non-technical domains that one may find on the Web, but have yet to be bookmarked by a Delicious user.

- After being acquired by Yahoo, Delicious imposed stricter limitations on resource and user crawling. Delicious limits the number of bookmarks per resource page to fifty, does not allow any sorting or filtering of bookmarks chronologically or by tag, and provides only the first forty pages. Thus, our crawlers could only collect the 2,000 most recent bookmarks for a given resource, meaning the crawlers missed a significant percentage of the bookmarks for the most popular resources. Collecting all of the resource's older bookmarks from user-based crawling would be prohibitively time-consuming and expensive with no guarantee that the resource's bookmark history is complete.

  When crawling a particular user, Delicious limits the number of bookmarks per page to 100, provides only the first forty pages, but does allow one to re-sort the bookmarks list in chronological order. Thus, Delicious allows one to crawl a maximum of 8,000 bookmarks. For the most prolific users – e.g., the Delicious user "angusf" has over 86,000 public bookmarks as of March, 2011 - we will inevitably miss some portion of their bookmarks.
- Although our crawl collected 73 million bookmarks, this total did not include all of Delicious' bookmarks as of August, 2010. Thus our dataset is missing some number of users and resources. Delicious' restrictions aside and despite efforts to ensure that our crawling would produce a representative data sample, the data may be biased due to our partial crawl.

1.4 DEFINITION OF TERMS

1.4.1 Annotation

Annotation as defined by the Oxford English Dictionary is *"[a] note added to anything written, by way of explanation or comment*." Vatton et al. (2004) define annotations as "*comments, notes, explanations, or other types of external remarks that can be attached to a Web document or to a selected part of a document*." Petkovic et al. (2005) write that "*annotations can represent comments and remarks users create for themselves or for others, referring to a specific piece of content (word, paragraph, image region etc.)*" Together, these definitions emphasize that annotations 1) are created to communicate information about an entire document, or just part of a document, and 2) may be created for personal or collaborative use.

In this study, annotation is defined as information attached to a resource by a person for the purpose of communicating or summarizing information related to some resource. The annotation may be intended for personal or public use.

1.4.2 Bookmark

Abrams, Baecker, and Chignell (1998) define bookmarks as "file surrogates (aliases) pointing to original files in 'tertiary storage,' the massive distributed file system located in Web servers distributed around the world." Here, bookmarks are defined as locally-stored pointers to URLs of Web resources for later recall. Bookmarks are typically stored locally in a user's Web browser.

1.4.3 Metadata

Metadata as defined by the Oxford English Dictionary is "*a set of data that describes and gives information about other data*." More simply, metadata are data about data. In the context of the World Wide Web, metadata are *structured or semi-structured* data that describe a resource. Here, annotations on Web resources are defined as a form of semi-structured metadata.

1.4.4 Social Annotation

A social annotation is a piece of shared metadata generated by individuals on a collection of Web resources within the confines of Web-based annotation system.

1.4.5 Social Bookmark

A social bookmark is a specific kind of bookmark, defined here as a pointer to a specific resource, created by a specific user, organized via user-defined annotations, or "tags", and stored in a shared, Web-based environment.

1.4.6 Tag

A tag is a free-text, open-ended annotation assigned to a bookmark by a user as metadata to describe a Web resource (Tonkin, 2006). Most tags are arbitrary words or acronyms applied to a resource as a descriptor and/or a mnemonic device for later recall; however, a tag may consist of

any continuous set of characters (e.g., letters, digits, punctuation, or other special characters). In most systems a tag is bounded by spaces.

Additionally, a *compound tag* is a tag comprised of two or more dictionary terms separated by an optional separator character (e.g., "/", "_", "+"). *Variants* of a compound tag are different forms of the tag that consist of the same dictionary terms in the same sequence, but use alternative separator characters. For example, variants of the common compound tag "webdesign" include "web-design", "web_design", and "web+design."

1.4.7 Resource

According to RFC 1736 (1995), an electronic resource may be "animate beings or physical objects with no electronic instantiation" or electronic, networked artifacts such as "an electronic document, an image, a server (e.g., FTP, Gopher, Telnet, HTTP), or a collection of items (e.g., Gopher menu, FTP directory, HTML page)" (RFC 1736, p.3.) In this paper, a resource is a Web-based object - typically a document, image, audio, video, or other multimedia file - that is bookmarked and optionally annotated with tags by some user.

1.4.8 URL

RFC 1738 (1994) defines a Uniform Resource Locator (URL) as "a compact string representation for a resource available via the Internet." (RFC 1738, p.1.) URLs serve as abstract identifiers of a resource's location (p.2.) Multiple URLs may point to a single resource; consequently, social bookmarking systems may contain bookmarks pointing to different URLs that identify a common resource.

1.4.9 Taxonomy

Taxonomy as defined by the Oxford English Dictionary is *"classification, esp. in relation to its general laws or principles; that department of science, or of a particular science or subject, which consists in or relates to classification;* esp. *the systematic classification of living organisms*." Garshol (2004) defines taxonomy as "a subject-based classification that arranges the terms in the controlled vocabulary into a hierarchy without doing anything further" (p.381.)

Here, taxonomy is defined as any hierarchical classification system. The controlled vocabulary aspect of Garshol's definition does not apply to social bookmarking systems, although this research does intend to treat tags as pseudo-subjects, i.e., metadata to group and categorize resources.

1.4.10 Folksonomy

A folksonomy is the aggregate, user-generated network of tags applied to Web resources collected in a social tagging system. The term "folksonomy", attributed to Vander Wal (2005) is derived from a combination of the words "folk" (i.e., non-expert users) and "taxonomy" (i.e. a classification structure). Other terms used in the literature to describe this structure include "social classification" (Tonkin, 2006; Feinberg, 2006), "distributed classification" (Hammond et al., 2005; Speller, 2007), and "collaborative tagging." (Golder and Huberman, 2006.)

Like Trant (2009), this paper views "tagging" as a process and the folksonomy as the resulting "collective vocabulary" (Trant, p.4) and organizational structure of tags.

1.4.11 Noise

In information theory, noise is defined as "statistical and unpredictable perturbations" that interfere with the transmission of a signal (Shannon, 1949.) In the context of social bookmarks, noise refers to irrelevant tags used to classify resources. Here, noise may also refer to idiosyncratic tags applied by only one user of a social bookmarking system.

1.4.12 Power set

The power set of a given set *S* is the set of all possible combinations of elements (i.e., subsets) of *S* (Dyrholm, 2009.) Given that *S* contains *n* elements, the power set *P*(*S*) will contain 2^n subsets, including the empty set. In this paper, *S* is the set of tags applied to a given bookmark, and the power set *P*(*S*) is all possible combinations of tags within *S*.

2.0 RELATED WORK

First, this chapter provides a brief review of research on annotations; bookmarks and social bookmarks - specific forms of Web-based annotations; classification schemes; how expert and novice classifiers classify documents; and domain expertise. Second, it reviews research on methods for identifying expert users in Web-based systems. Finally, the chapter concludes with a discussion of research on automatic classification of social bookmarking data.

2.1 BACKGROUND

2.1.1 Annotations

For centuries, readers have annotated paper-based documents for a variety of purposes. Copy editors annotate manuscripts to give authors feedback and type setters formatting instructions. Many students underline, highlight, or circle passages of text they believe raise salient points, or are otherwise useful to learning. They may jot comments in the margins of text to summarize key topics, or describe in their own words what the author has written. Instructors mark papers with comments, corrections, or counter-arguments as a dialog with their pupils. In fact, Adler (1940) argues that annotating is an essential part of reading a book - "a conversation between [the reader] and the author" – and suggests several ways to properly annotate a document.

In his seminal paper "As We May Think", Bush (1945) describes a prototype hypertext system, "Memex", that would leverage the utility of annotations beyond any system yet described or developed. Bush envisioned a system where researchers could easily copy and store manuscripts, photographs, and other materials on microfilm. Researchers could access items by browsing their collections, or move immediately to an item by typing its assigned mnemonic code. Information could then be edited or annotated in real-time. The key to the system, as Bush notes, is the researcher's ability to link any two items together with codes to build "trails" of information. Bush saw the researcher's "web of trails" as a more natural and efficient means of retrieval then traditional indexing, operating by association much like the human brain.

Research on annotations has begun only recently (Choochaiwattana, 2008). Marshall (1997, 1998) conducted a series of studies examining how people annotate books, and how such annotations could be made, stored, and used on Web documents. Table 1 shows a series of dimensions proposed by Marshall (1998) for describing the forms, functions, and roles of annotations. Marshall also notes the role annotations play when buyers of used textbooks make their selections – experienced buyers prefer books with hand-written annotations in the margins over books with only implicit annotations (e.g., highlighted or underlined passages.) Buyers felt explicit annotations conveyed greater authority - i.e., "notes taken in class" – increasing the perceived value of the book.

Choochaiwattana (2008) classifies the purposes of annotations into four categories:

Table 1. Marshall's dimensions of annotations (1998)

Forms of Annotation	
Formal	Follows well-defined, standardized structural rules.
Informal	Follows no structural rules; ad-hoc.
Tacit	Meaning is understandable only to the annotator.
Explicit	Meaning is understandable to everyone.
Functions of Annotation	
As reading	Organization of content or navigational aids to assist the reader.
As writing	Commentary or explanation beyond the author's text.
Hyperextensive	Focus is on linking documents (i.e., creating hyperlinks)
Extensive	Focus is on organizing similar documents (e.g., bookmarks)
Intensive	Focus is on a single document.
Permanent	Useful for an indefinite period.
Transient	Useful for the current reading session only.
Roles of Annotation	
Published	Everyone is authorized to read.
Private	Only certain individuals or groups are authorized to read.
Global	Audience is everyone.
Institutional	Audience is organization- or enterprise-wide.
Workgroup	Audience is the annotator and his/her colleagues.
Personal	Audience is the annotator.

- *Annotation for Memory* – used to help the reader locate useful sections of a document, or recall important concepts within a text.

- *Annotation for Communication* – used to exchange information between the annotator and the reader. The audience for these types of annotation can be any of the four audience groups (i.e. Global > Personal) defined by Marshall under her annotation roles.

- *Annotation for Collaboration* – similar to communication, but used specifically by workgroups to exchange ideas, provide feedback, or facilitate workflow to achieve a common goal. Collaborative annotations produced electronically may be shared in real-time by team members in different locations.

- *Annotation for Description* – used to describe or classify objects. Descriptive annotations can be used to improve retrieval of documents, images, or other objects.

2.1.1.1 Bookmarks

In the context of Web browsers, bookmarks are locally-stored pointers to URLs of Web resources – very similar in nature to the mnemonic codes in Bush's Memex. Bookmarks (also called "favorites" or "hotlists") first appeared in NCSA's Mosaic browser in 1993 and are now a standard feature of all major browsers. Abrams, Baecker, and Chignell (1998) define bookmarks as "file surrogates (aliases) pointing to original files in 'tertiary storage,' the massive distributed file system located in Web servers distributed around the world." In addition to a resource's URL, a bookmark typically stores the resource's title, an optional user-supplied description, and an optional set of keywords.

Along with queries issued to search engines, bookmarks are one of the most popular ways users locate information on the Web. Abrams, Baecker, and Chignell classify the reasons users create bookmarks into three categories:

1. *Reducing user load* - make it easier to manage URLs; aiding memory and keeping history.

2. *Facilitating navigation/access* - speeding information access; finding Web information.

3. *Collaborating/publishing/archiving* - creating a personal Web information space; authoring and publishing Web pages; collaboratively using Web information.

Keller et al. (1997) note that virtually all browsers allow users to create bookmark folders and organize their bookmarks hierarchically. Abrams, Baecker, and Chignell found that the use

and complexity of a personal hierarchical bookmark structure depended largely on the number of bookmarks a user has saved. Thirty-seven percent of respondents to their bookmark-use survey did not organize their bookmarks in any way, but the majority of this group had less than 35 bookmarks. Users with 26-300 bookmarks were more likely to use a shallow hierarchy of bookmark folders, while users with more than 300 bookmarks tended to use multi-level hierarchies. Abrams et al. also report that creators of multi-level bookmark hierarchies found it difficult to retrieve bookmarks from their collection, an observation consistent with Lansdale's conclusion that users have great difficulty finding objects within deeply-nested hierarchies (Lansdale, 1983). Creating a bookmark is very simple, but choosing the right location in a hierarchy for a new bookmark – let alone creating and maintaining a hierarchical structure – is a laborious process.

2.1.1.2 Social Bookmarks

Keller et al. noted the importance of the bookmark as a tool for storing, organizing, and recalling useful resources on the Web. However, they felt the utility of browser-based bookmarks were limited by 1) a hierarchical organization scheme that is difficult to maintain and navigate, forcing users to place a bookmark in a single folder, 2) an inability to share bookmarks with other users, and 3) an inability to rank bookmarks by utility. The authors built a proxy-based collaborative bookmarking system, WebTagger, that allowed users to categorize bookmarks in multiple categories and share bookmarks with others in a "group memory", or store the bookmark privately in their "personal memory." WebTagger was not the first "public link management application" (Hammond et al., 2005), but was the first to abandon hierarchical folders in favor of multi-faceted, user-defined categories for bookmark organization.

Heymann, Kouritka, and Garcia-Molina (2007), Hotho et al. (2006), and Dellschaft and Staab (2008) all define a *social bookmark* as a 3-tuple consisting of a user *U*, a resource *R*, and a set of tags, *TS*. Golder and Huberman (2006) add that these annotations are "social" because users may view the bookmarks of other people, not just their own (p.201.) Users may freely browse each other's bookmarks to learn what resources interest fellow community members and how they classify these resources.

Social bookmarks may fall into any of Choochaiwattana's four categories of annotations. Many users create social bookmarks to store useful links so they or other users can recall the linked resource at a later date (i.e. annotation for memory.) Some users add tags to their bookmarks to describe the resource's content (i.e. annotation for description.) Social bookmarking systems that focus on communities of interest in the enterprise, such as Dogear (Millen, Feinberg, & Kerr, 2006), or in the general public, such as CiteULike, encourage users to share their bookmarks and tags with other group members (i.e. annotation for collaboration and communication.)

Hammond et al. (2005) provide an early review of social bookmarking systems, including those for general Web resources, such as Delicious, StumbleUpon, and Simpy, as well as systems concentrated in a particular domain, such as CiteULike and Connotea for academic papers. They list the following elements as common characteristics of virtually all social bookmarking systems (p.11):

- Personal user accounts (groups sometimes provided).
- Mechanism for entering links, titles and descriptions.
- Classification by 'open' or 'free' tagging
- Search by tag or user (Boolean combinations sometimes allowed)

- Querying of links based on popularity, users, tags, etc.
- RSS feeds
- Extensions such as browser plug-ins

Trant (2009) provides a review of social bookmarking system research, identifying three main themes in the literature. First, researchers have studied using social bookmark tag sets as metadata to improve information retrieval. Information from bookmarks may be used to enhance indexing algorithms, such as Hotho et al. (2006) and Bao et al. (2007), or to build classification schemes (to be discussed in section 2.3 of this chapter). Hotho et al. (2006) present an adapted version of PageRank called FolkRank that converts the directed edges of users, tags, and resources into an undirected graph, and then calculates a topic-specific FolkRank score in the folksonomy. Bao et al. (2007) develop and test two ranking algorithms for folksonomies: SocialSimRank, a query-dependent score for social annotations that was successful in uncovering latent semantic relations among tags, and SocialPageRank, a query-independent rank to measure the popularity of a resource. Wu et al. (2006) present an approach for disambiguating Delicious tags and uncovering semantic relations, as users annotate the resources they bookmark without using a controlled vocabulary or ontology. Begelman et al. (2006) present several clustering algorithms to improve search results by locating tags semantically related to query terms. Heymann et al. (2008) collect and analyze a large sample of Delicious data to evaluate its utility for improving Web retrieval. They conclude that the service's growth and substantial portion of unindexed pages may make it valuable to search despite the relatively high overlap of resources with prominent search results (p. 199) and tags with page titles and text (p. 202.) Choochaiwattana and Spring (2009) examine methods to use Delicious tag data to improve resource indexing and search result rankings. They found that their Normalized Match Tag

Count method, which rewards resources with the highest percentage of Delicious users annotating them with matching terms for a given query, performs significantly better than methods that rely solely on resource popularity (i.e. bookmark count). From these studies, we conclude that tags can improve the performance of information retrieval, particularly for query-dependent algorithms, as well as when combined with the full text of resources.

Secondly, many studies of social bookmarking system have focused on the tagging behavior of users. Vander Wal (2005) notes the significant differences in the network structures of Delicious, a *broad folksonomy* where many users bookmark the same resources, and Flickr, a *narrow folksonomy* where most resources are bookmarked by only one user. Marlow et al. (2006) provide a taxonomy of social tagging systems, highlighting how user motivation, resource types, and tagging support can affect network structure. Bischoff et al. (2008) found that the prevalence of certain tag types varied among social bookmarking systems, depending on the system's focus. The authors found that "topic" tags – tags describing what a resource is about (e.g. "webdesign" or "java") – are the most common class of tags in Delicious, while "type" tags – tags describing what a resource is (e.g., "mp3", "blog") – appear most often in Last.fm, a social bookmarking system for music. Syn and Spring (2009) examined how well tags on bookmarked resources in CiteULike described content compared to author-assigned keywords from a controlled vocabulary on the same resources in the ACM Digital Library. Among their findings is that although keywords performed better than tags in describing content based on cosine similarity to terms in the titles and abstracts, the performance of tags (and keywords) increased as the number of terms used to annotate the resource increased. When comparing tags to terms at different levels in the ACM Computing Classification systems, the authors found that tags did a significantly better job representing specific topics than general ones.

Golder and Huberman's early paper on tagging patterns illustrated how the broad folksonomy in Delicious follows a Zipf distribution, or the *power law*, in terms of tag usage, resource selection, and system usage by user (Golder and Huberman, 2006). Halpin, Robu, and Shepherd (2007) and Wetzker, Zimmermann, and Bauckhage (2008) confirmed this observation regarding Delicious, noting the exponential growth of the system in a short period of time. While the success of Delicious and other social bookmarking systems is due in large part to the ease with which one can save and freely annotate Web resources, this ease-of-use comes with potential costs. Chi and Mytkowitcz (2008) analyze data from Delicious using several measures of entropy. They conclude that Delicious' tag vocabulary is becoming less efficient, making the site harder to navigate. Guy and Tonkin (2006) provide suggestions to improve the quality of tags to make them more conducive to search and classification. Syn (2010) presents a method for decomposing compound tags and two TF/IDF-inspired metrics, Annotation Dominance (AD) and Cross Resource Annotation Discrimination (CRAD), to reduce tag noise, as well as find professional-quality classificatory metadata in among tags in Delicious.

Finally, a third stream of research examines social bookmarking systems as "socio-technical systems" (Trant, p.17), i.e., how users within a system interact with each other and the system's features. Although Delicious permits users to freely tag their bookmarks with no set vocabulary, researchers have found evidence that the tag vocabularies of individuals tend to stabilize, and even converge, over time. Udell (2005) observed that the number of new tags in a Delicious user's vocabulary gradually decreases over time as he enters new bookmarks. Millen, Feinberg, and Kerr (2005) found a similar trend in their enterprise social bookmarking system, Dogear. Golder and Huberman (2006) found that a resource's top tags tend to stabilize after the first 100 users have bookmarked the item. Dellschaft and Schaab (2008) present a model

showing strong evidence that a Delicious user's own vocabulary and the previous tag assignments on a given resource heavily influence the user's tagging behavior. The authors also found a sharp drop in the frequency-rank distributions of tags on popular resources after Rank 7, possibly due to Delicious presenting users a maximum of seven tag suggestions at the point of bookmark creation. Li, Guo, and Zhao (2008) present their Internet Social Interest Discovery (ISID) system that uses tag-based discovery to cluster users with similar topics of interest, even if those users have no social connections to each other. The authors' algorithm looks for frequent co-occurrence patterns of tags to identify topics, and clusters both Delicious users and documents based on topic/interest similarity. Finally, Hassan-Montero & Herrero-Solana (2006) produce a clustered version of the popular tag cloud often used to visualize tag vocabularies. Rather than present tags in alphabetical order, the authors' tag cloud presents semantically-related tags in horizontal clusters, reducing the semantic density of the tag set.

2.1.2 Classification

Classification is the process of creating relations between objects and a pre-defined set of categories. This pre-defined set of categories and its structure constitute a classification scheme (Fettke and Loos, 2002.) Humans create classification schemes to better organize and retrieve information in a wide variety of domains, such as the Periodic Table of Elements for chemical elements, and the Dewey Decimal system for arranging documents in a library. Although many of the most-widely known classification schemes are hierarchical, several researchers (including Bailey, 1994; Gaus, 1995; Fettke and Loos, 2002) identify four types of classification schemes, shown in Table 2.

Bischoff et al. (2008) showed that the most prevalent tags in Delicious are those that describe what a resource is about. In essence, Delicious users who provide these descriptive tags are classifying their bookmarked resources within their own personal taxonomy. A professional librarian working in a traditional library creates more formal, but similar classificatory metadata for each resource called *subjects*. One of the main objectives of this research is to see if Delicious contains users who annotate resources with comparable expertise to professional cataloguers. In turn, can their annotations be used as "subjects" for a shared classification scheme to improve recall of resources in social bookmarking systems?

This section first explores the main approaches to *categorization*, the ways in which humans recognize and differentiate objects, followed by a review of subject-based classification schemes. The section concludes with a review of the literature on *subject analysis*, the process expert cataloguers use to classify resources.

Table 2. Types of Classification Schemes (from Fettke & Loos, 2002)

Classification Type	Description
Basic or Enumerative	Each object is an element of one class. Classes are defined by specific characteristics with no overlap. The structure of a basic classification is flat.
Hierarchical	Similar to basic classification, but the classes are ordered hierarchically in a tree-based structure. One super-class can include one or more sub-classes.
Faceted	Each object is classified according to different viewpoints, called facets, which are completely distinct. Each object must be classified according to all facets.
Characteristic-based	Each classification object is characterized by several characteristics. In contrast to faceted classification, the characteristics do not need to be completely distinct

2.1.2.1 Approaches to Categorization

Classical categorization originated with Plato and Aristotle, who were the first in the Western world to consider grouping and labeling objects with shared properties (Langridge, 1989.) In *Categories*, Aristotle theorized that human knowledge may be divided into ten discrete categories. This classification system served as the basis for modern taxonomies – hierarchical classification schemes – in which entities must reside in a single category. Categories derived from the classical approach should be clearly-defined, perfectly discrete (i.e., no overlap or fuzziness), and collectively comprehensive. Philosophically, classical categorization assumes that categories are objective, existing independent from human perception and defined strictly by the properties of its members (Lakoff, 1987.)

Conceptual clustering is a recent derivation of classical categorization that serves as the foundation for unsupervised machine learning, algorithms that "learn from observation" (Michalski and Stepp, 1983a.) In this approach, an algorithm accepts a series of object descriptions, and then uses an evaluation function to define logically disjoint conceptual descriptions. Objects are then classified according to these descriptions. The main goal of early conceptual clustering algorithms, such as CLUSTER/PAF (Michalski and Stepp, 1983b), was to produce a hierarchical classification scheme similar to traditional taxonomies. Later conceptual clustering systems, such as COBWEB (Fisher, 1987), attempted to build hierarchical classifications through incremental learning, a process that better reflects the real-world environments a human might encounter when classifying objects.

The best classification schemes produced from conceptual clustering exhibit high intra-class similarity and low inter-class similarity (Fisher, 1987.) Gluck and Corter (1985) developed a metric known as *category utility*, or category "goodness", to measure this phenomenon. To

Corter and Gluck (1992), a classification scheme is useful if a typical category label helps a person accurately determine what the properties of its objects are, and iteratively, knowledge of the properties helps a person accurately predict in what category an object belongs.

Prototype theory, first described by Rosch (1973), is yet another approach to categorization. Although hierarchical in nature, prototype theory is otherwise a radical departure from classical categorization. According to the prototypical view, "natural" categories tend to be graded or fuzzy, not completely discrete classes as in classical categorization, with members that have similar, but unequal characteristics (Rosch, 1973.) Certain members of a category are more representative, or central, to a category than others – i.e., a "robin" is more prototypical of "bird" than "penguin". Furthermore, categories and their meanings are rooted in, not separate from, human cognition (Lakoff, 1976.) Rosch's famous experiments involving categorization led her to theorize that humans recognize objects at the 'basic level' of understanding – the level at which humans are most likely to interact with them (Tanaka and Taylor, 1991.) Objects may also be described at the super-ordinate (more general) or sub-ordinate (more specific) levels, but the basic level category is the one that is maximally informative, providing the highest category utility.

2.1.2.2 Classification Structures

Garshol (2004) provides an overview of subject-based classification structures and their effectiveness for organizing Web-based resources. The author compares four traditional schemes from library science that used controlled vocabularies – taxonomies, thesauri, faceted classification, and ontologies – to topic maps, a relatively new classification structure:

- *Taxonomies* are hierarchical classification schemes in which objects with similar properties are grouped together. Subjects that encompass a broad array of objects reside

toward the top of a hierarchy (e.g., “animals”), while subjects that describe more specific sets of objects reside toward the bottom (e.g., “dogs”.) Thus, taxonomies imply super-ordinate/subordinate or “parent-child” relationships among subjects. While basic taxonomies are relatively simple structures to understand, they lack the ability to express more complex relationships among subjects (Garshol, 2004.) Locating subjects within a taxonomy may be difficult for users who are unfamiliar with the subject vocabulary, or if the hierarchy is deeply-nested (Lansdale, 1983.)

- *Thesauri* are extensions of taxonomies that are able to describe additional relationships between subjects. Besides hierarchical relationships (i.e., “broader term” and “narrower term”) thesauri typically include synonyms, related terms to a subject that are neither synonyms nor parents/children, and scope notes that provide contextual descriptions of a subject when the subject’s meaning may be unclear (Garshol, 2004.) By retaining the precise subject names and hierarchical structure of taxonomies, but including additional relationships and terms more in tune with how end users view and describe a domain, thesauri are appealing for Web-based resource classification. Still, the number of additional relationships is very limited, so thesauri are typically not powerful enough to precisely describe a domain.

Table 3. An example of a faceted classification of a hypothetical book on 17th century Norwegian architecture using Ranganathan's Colon Classification (reproduced from Garshol, 2004.)

Facet	Description	Example
Personality	Primary subject of the resource; considered the main facet.	Architecture
Matter	Material or substance of the resource's subject	Wood
Energy	Key process or activity described by the resource.	Design
Space	Location of the resource's subject.	Norway
Time	Time period described by the resource	17th Century

- *Faceted classification*, originally proposed by Ranganathan (1963), is a scheme where resources are described by selecting a single term from multiple axes or "facets", allowing for multiple classifications of a set of resources rather than a single taxonomy. Table 3 shows an example of a resource classified using the five facets of Ranganthan's Colon Classification (Garshol, 2004.) Faceted classification generally permits a subject to be included in only one of the facets. Garshol notes that while faceted classification may seem radically different, it may actually be a more disciplined form of thesaurus suitable for classification purposes (p. 383.)

- *Ontologies* in the information sciences are formal, explicit, shared conceptualizations of a domain (Gruber, 1993.) These structures allow classifiers to very precisely describe both an object and its relationships with other objects in a domain. Garshol notes that

ontologies do not use a controlled vocabulary, but efforts such as the Web Ontology Language (OWL) try to standardize descriptions of Web-based ontologies. Ontologies have the most descriptive power of the classification schemes mentioned, but are also the most difficult to build and maintain. High-quality ontologies require significant domain expertise to build. Experts in the same domain may have very different viewpoints about how a domain should be conceptualized. Multiple ontologies may be built if builders cannot come to an agreement, leading to the problem of how to make the ontologies semantically interoperable (i.e., ontology mapping.) Given the fluidity and distributed nature of the Web, ontologies have yet to become a popular form of Web resource classification.

- *Topic maps* are a relatively new form of classification first described in Pepper (2000). As described by Garshol (2004) and shown in Figure 2, topic maps appear to be standardized lightweight ontologies (or meta-structures for other classification schema) where topics (i.e. real-world entities) are linked together through one or more associations to form a semantic network. Occurrences of topics, such as Web resources, typically form a distinct layer separate from the topics and associations (although the occurrence "Curing the Web's Identity Crisis" is depicted on the same layer in Figure 2.) Topic maps themselves are not ontologies, because their main goal is make information easier to locate, not to specify a precise model of a domain. However, Garshol and Pepper argue that topic maps are flexible enough to express any subject-based classification from complex ontologies to basic taxonomies. While this flexibility may be an advantage, it also means that topic maps will suffer from the same problems as the classification scheme they most closely represent. Even as a lightweight ontology, a

topic map may be difficult to build and maintain, may not scale well on the Web, and may be difficult to merge or map with other topic maps.

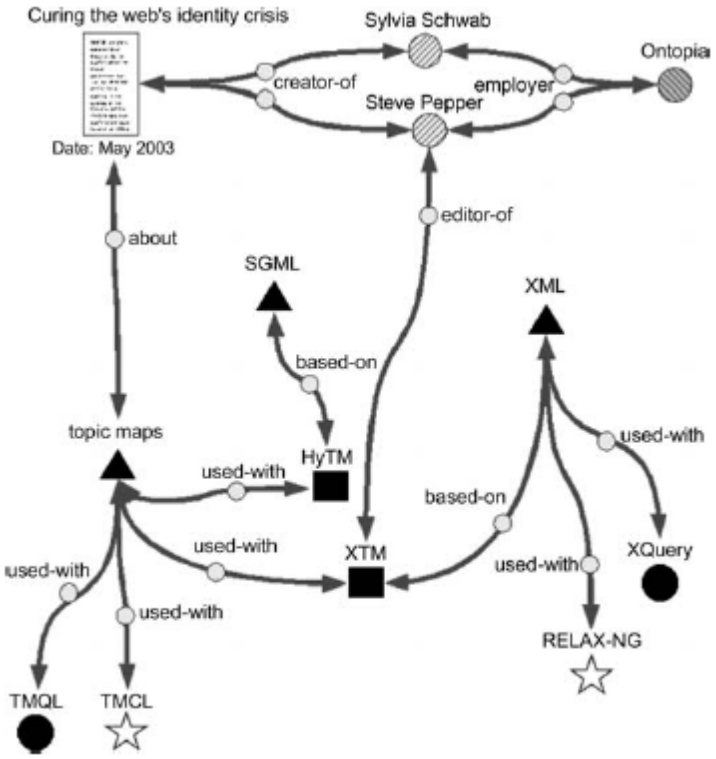


Figure 2. A topic map describing topic maps. Large shapes are topics. Arrows denote relations. The paper icon in the top left corner represents and occurrence (i.e. resource) of topic maps (reproduced from Garshol, 2004.)

2.1.2.3 Subject Analysis - Classification by Experts

Much like annotation, subject analysis - also called "subject indexing" by Voss (2007) - has only become a focus of research in the past twenty-five years. Subject analysis is the process of determining what a resource is about conceptually and expressing this as index terms in the vernacular of a controlled vocabulary (Lancaster, 2003; Langridge, 1989; Taylor, 1999.) Langridge argues that is the most significant activity of information specialists, whose responsibility it is to organize our collective knowledge in as accessible a manner as possible. Langridge (1989) and Voss (2007) also argue that there are two steps in the process – conceptual analysis and translation. While distinct activities, their boundaries are often blurred by those too focused on fitting a resource into a classification's structure.

Table 4. Langridge's steps in the conceptual analysis phase of subject analysis (reproduced from Appendix 3 of Langridge, 1989.)

I. Examination of Text	*II. Analysis of each unit*	*III. Summarization of findings*
1. Preliminaries: Title, sub-title, author, contents list, chapter headings.	1. Determine fundamental form of knowledge (e.g., Science)	Write down complete analysis in own words (i.e. one summary for a homogenous work; separate summaries each unit of composite works.)
2. Read introduction and dust jacket.	2. Determine discipline (e.g., Zoology)	
3. If necessary, sample text, check external information, e.g. book reviews.	3. Determine topic (e.g., Respiration in Fish)	
4. Determine whether homogenous or composite work.	4. Determine nature of thought (e.g. instructive monograph in English, elementary level.)	

Here, we are most interested in how an expert performs conceptual analysis, because we assert "expert" social bookmarking users to engage in a similar activity when tagging their resources. Our interest in translation is limited to the number of terms the cataloguer ultimately selects, which may provide clues into how much metadata a typical expert may add to a resource. Unlike professional cataloguers, users of public social bookmarking do not have to map their annotations to a shared, controlled vocabulary.

Table 4 shows Langridge's suggested steps for conceptual analysis, a process similar to that suggested by Taylor (1999), but only partially observed by Sauperl (2002) in her study of twelve professional cataloguers. Langridge stresses the importance of looking at the title, author(s), dust jackets, introduction, and chapter headings for the author's view of the resource's subject matter. He also reminds cataloguers to examine the resource's forms of knowledge and writing, and determine its topic and discipline to avoid indexing mistakes from taking a title at face-value. Sauperl, however, found little evidence of her subjects using Langridge's theoretical distinctions among knowledge, form, topic, and discipline when selecting tentative headings.

In the summarization step – the one akin to tag selection by social bookmarking system users – Langridge suggests that cataloguers write down concisely the form of knowledge and precise topic of the resource *in their own words* – not necessarily in the vocabulary of a classification scheme's subject headings. This summarization is typically a series of terms, or a few sentences if the resource has multiple units. The goal of summarization is to distinguish the resource as much as possible from others, grouping the resource with the few documents whose conceptual analysis yielded similar results (Langridge, 1989.) Sauperl (2002) notes that this fine-grained classification is what distinguishes domain experts from novices – the expert's schema is more complex, but is better organized and can handle exceptions more efficiently.

Prior to electronic records, the number of subject headings assigned to a resource tended to be low to reduce the number of entries in the card catalogue. Bates (1986) found that the Library of Congress and large academic libraries average about two subject headings per resource. Khosh-khui (1987) confirmed this number, and found no correlation between the number of terms in the subject headings and the number of subject headings applied. As computerized records reduced the cost of adding and maintaining additional subject headings, the number of headings began to steadily rise. Chan and Hodges (2000) note that the Library of Congress recommends that six headings are appropriate, on average, and that ten headings are the maximum.

2.1.3 Domain Expertise

The Oxford English Dictionary defines expertise as "the quality or state of being expert; skill or expertness[1] in a particular branch of study or sport", and defines domain in this context as "a sphere of thought or action; field, province, scope of a department of knowledge, etc." Ericsson (2006, p.3) states that expertise "refers to the characteristics, skills, and knowledge that distinguish experts from novices and less experienced people." Domains may be formal areas of knowledge, such as chemistry and the performing arts, or informal ones like cooking and sewing (Chi, 2006.) Research interest in domain expertise has grown over the past several decades, particularly with the advent of artificial intelligence and expert systems. Most research on expertise has sought to isolate the skills and factors that contribute to expert performance. Some efforts focus on a single domain (e.g. chess), while others attempt to develop a general

[1] The O.E.D. defines expertness, a term that pre-dates 'expertise', as "skill derived from practice; readiness, dexterity."

Table 5. Ericsson's list of general theoretical frameworks of domain expertise (2006)

Theoretical Framework of Expertise	Description
Individual differences in mental capacities	General, hereditary mental capacities lead to expert performance in most domains.
Extrapolation of everyday skill in extended experience	Expertise is a natural extension of years of domain experience; over time, experts learn patterns and strategies to achieve superior performance.
Qualitatively different knowledge representation and organization	Experts store and organize accumulated knowledge differently than non-experts; expert systems codify these knowledge representation patterns to emulate expert performance.
Elite achievement due to superior learning environments	Early instruction, exceptional teachers, and family support lead to expert performance.
Reliably superior performance on representative tasks	Expert performance in many domains can be reproduced and measured in controlled environments through a series of representative tasks.

theoretical framework of expertise across multiple domains. Because social bookmarking systems attract users from various domains, this review focuses on work describing general theoretical frameworks of domain expertise.

Ericsson (2006) divides theoretical frameworks for expertise into five categories (Table 5.) In 1869, Galton proposed the first framework of expertise, arguing that outstanding intellectual achievement was the result of *individual differences in mental capacities*, differences that were hereditary and generalizable across multiple domains. Later research, however, found no evidence to support Galton's hypothesis. For example, Djakow, Petrowski and Rudik (1927) found that expert performance is often very domain-specific and not generalizable to other areas.

Ericsson and Lehmann (1996) concluded that mental capacities are not valid predictors for expertise; any significant performance differences between experts and non-experts resulted from experts acquiring key skills and knowledge during lengthy training.

A second framework views expertise as an *extrapolation of skills and knowledge acquired through extended experience*. Early studies, including Bryan and Harter (1899), believed domain expertise was the natural consequence of lengthy experience; even today, many people equate length of experience with expertise. Research by de Groot in the 1940s, followed by Simon and Chase (1973), found that elite chess players' superiority was due to their ability to recall complex patterns and strategies learned through experience. Simon and Chase's work on experts' pattern formation and memory influenced a third theoretical framework of expertise based on the notion that *experts store and organize knowledge in memory in fundamentally different ways* than non-experts. Early research focused on this framework aimed to build computer-based models, i.e. expert systems, around experts' domain knowledge to replicate expert performance.

Ericsson defines the fourth theoretical framework of domain expertise as *expert performance resulting from superior learning environments*. Bloom (1985) interviewed elite performers from six domains to collect information about the major influences on the performers' development. Bloom et al. found that all participants provided evidence of favorable learning environments – early instruction, supportive families, and exceptional teachers throughout development. Finally, the most recently-developed theoretical framework of expertise centers on the notion that *reliably superior performance on representative tasks in a given domain* measures expertise. Ericsson and Smith (1991) argued that expertise can be studied in a controlled setting, because many domains have specific tasks that serve as good benchmarks

for comparing the performance of experts with non-experts. In fact, Camerer and Johnson (1991) found that in some domains, such as medicine and stock-picking, people identified as experts through reputation and experience performed no better on representative tasks than less-experienced peers. However, using controlled studies, researchers can also look to isolate the skills, abilities, or other characteristics that lead to expert performance. Subsequent studies conclude that consistent, deliberate practice – a planned regimen of persistent learning within a domain – is a better predictor of expertise than the number of years of experience (Ericsson, Krampe, & Tesch-Römer, 1993; Ericsson & Lehmann, 1996.)

Alternatively, Chi (2006) divides the study of expertise into two approaches: *absolute* and *relative*. In the absolute approach, researchers focus solely on exceptional individuals, trying to understand how they achieve superior performance in their respective domains. Exceptional performers may be identified retrospectively (e.g., assessments of bodies of work), concurrently (e.g., results of aptitude tests), or independently through some representative

Table 6. Chi's list of domain experts' strengths and shortcomings (Chi, 2006)

Experts' Strengths	Experts' Shortcomings
• Generating the best solutions faster and more consistently • Strong pattern detection and feature recognition • Qualitative analyses • Keener self-monitoring • Identifying appropriate problem-solving strategies • Opportunistic – better at working with limited resources • Exerting less cognitive effort to retrieve relevant knowledge and strategies	• Expertise is domain-limited • Over-confidence • Glossing over details and surface features • Dependence on context • Inflexible - adapt poorly when confronted with a profound structural change to a problem. • Inaccurate predictions of novice performance • Functional fixedness

domain task. Regardless of the measurement, the implicit assumption of the absolute approach is that the exceptional individual possesses inate talent or characteristics that explain their superior performance. The relative approach directly compares domain experts with non-experts, where "experts" are more knowledgeable or skilled in a domain compared to less proficient "non-experts". This approach assumes that non-experts can attain domain expertise over time; therefore, the goal of research is to identify the processes and factors that allowed experts to become proficient so others can reach the same level.

Chi then summarizes the general strengths and shortcomings of domain experts identified throughout the literature (see Table 6.) Some of the strengths and shortcomings listed by Chi have particular relevance in the context of public social bookmarking systems such as Delicious. Experts' strong feature recognition skills and ability to find the best solutions faster and more consistently than non-experts can help us locate domain and classification experts in Delicious. We expect domain experts in Delicious are those users who, on a consistent basis, identify the best resources in their respective domains faster than most peers. Given that expertise is often domain-limited, we assume domain expertise in Delicious will be topic dependent. Furthermore, because classification is itself a separate domain, we cannot assume that a domain expert in Delicious is also a classification expert across all domains. Domain experts often know highly-relevant terms to describe resources pertaining to their area(s) of expertise, but 1.) they may not use them to annotate their bookmarks, and 2.) they may not be as successful choosing the best terms to describe resources outside their domain expertise.

2.2 IDENTIFYING EXPERT USERS IN WEB-BASED SYSTEMS

When searching for useful information on the Web or enterprise system, users often look to sources – documents or people – they believe are the most authoritative. Such sources are more likely to provide reliable information and solutions to users' queries and problems, leading to reduced search and implementation costs for the user. Locating authoritative people and resources in the Web's vast and ever-expanding information repository, however, continues to be a challenge for researchers. Even with the emergence of social mechanisms such as community ratings in question-and-answer forums or the aggregation of social bookmarking data, the sheer volume of data on the Web causes many helpful resources and people to go undiscovered. This section describes the research on expertise in the bipartite graphs (i.e., users and documents) of traditional Web-based systems and the tripartite graphs (i.e., users, documents, and tags) of social bookmarking systems, both in the enterprise and on the public Web.

2.2.1 Identifying Expert Users in Bipartite Graphs

Ideally, an expert search system will include user profiles consisting of a series of documents (e.g., home page, research interest pages, meeting notes) that describe the expertise of each candidate (Macdonald, Hannah & Ounis, 2008). If such profiles are available, as is often the case in enterprise systems, the search system ranks the profiles against the user's query, providing a list of candidates whose expertise best matches the user's needs. Becerra-Fernandez (2006) provides a survey of early, Web-based expert locator systems in enterprise settings. Virtually all of these systems used a taxonomy of knowledge domains to help define an expert's area(s) of specialization, while the author's own system, Expert Seeker, used a clustering

algorithm to define areas of expertise. Recent work on expert search focuses on techniques to improve candidate rankings, such as using query expansion and query term-proximity within documents (Petkova and Croft, 2006) and improving document clustering to better describe a candidate's expertise (Macdonald, Hannah & Ounis, 2008.)

Profiles work well in an enterprise where user identities can be verified, but are a poor fit for the World Wide Web where such verification is often impossible. In the absence of formal descriptions of document authors, graph-based algorithms are used to find authoritative resources produced by experts. The notion of authority is critical in Web retrieval and a key underpinning of the most recognizable graph-based algorithms, PageRank (Brin and Page, 1998) and HITS (Kleinberg, 1999.) In both PageRank and HITS, a document derives its authority based on the number and quality of its incoming links. If many documents link to a resource *R*, and those documents are also considered authoritative resources themselves, *R*'s PageRank and HITS Authority scores will be high. HITS also computes a Hub score for each resource based on the weights of its outgoing links; that is, if *R* links to many other authoritative documents, *R*'s Hub score will be high. A resource with both high HITS Authority and Hub scores is very much like a human expert who is deemed an authority in an area by many knowledgeable people (i.e., incoming links) and has great command of the area's literature (i.e., outgoing links.)

Several studies have evaluated the performance of PageRank and HITS for finding experts in online communities. Campbell et al. (2003) used HITS and PageRank to find and rank subject experts in email correspondence, finding graph-based networks rank experts better than content analysis. Zhang, Ackerman, and Adamic (2007) tested their ExpertiseRank algorithm, based on PageRank, against other graph-based algorithms for finding expert users in Sun's Java Forum. Two human raters who were Java programming experts judged the expertise of the Java

Forum users, and the algorithms' performance were evaluated against their ratings. The authors found that all graph-based methods did a good job finding experts, but ExpertiseRank did not significantly outperform the simpler algorithms. When the authors tested the algorithms on simulated networks of varying structures, they found the performance of the algorithms varied greatly. The results suggested that a network's structure may be exploited to locate expert users, but that one must factor in the nature of the structure when selecting a technique to rank users.

Bharat and Mahaila (2000) developed a prototype search engine, Hilltop, that employed a relatively small index (2.5 million pages) of "expert documents" to harvest authoritative web pages. These "expert documents" contained at least five (5) non-affiliated links to target pages on a particular topic. In an evaluation, Hilltop performed significantly better than AltaVista and nearly as well as Google in tests of average recall and precision at *k*. Bharat and Mahaila's findings are noteworthy for this research, because they showed that reducing noise to produce a much smaller "expert index" still allowed Hilltop to locate relevant, authoritative documents on par with the top commercial search engines.

The sheer size of Web-scale networks prevents the use of some algorithms for various retrieval and summarization tasks. Evidence by Lee et al. (2006); Leskovec and Faloutsos (2006); and Shi et al., (2008) suggest carefully sampled sub-graphs can provide accurate depictions of the entire underlying graph. Shi et al. introduce the vertex-graph importance synopsis approach, which finds important, highly-connected vertices in a series of web and online social network datasets and efficiently builds accurate synopses of their respective graphs. The authors evaluate their approach on a series of bipartite graphs, including data from BuddyZoo (AOL), TREC Blog-Track, and Xerox PARC's "Web in a box" project. Unlike the findings of Zhang, Ackerman, and Adamic, Shi et al. found that their graph compression scheme

performs consistently regardless of the underlying network's structure. Given that experts in the context of the network tend to be highly-active users with many connections, graph compression may be a promising approach to locating experts in online communities.

2.2.2 Identifying Expert Users in Tripartite Graphs

With the rising popularity of social bookmarking systems – especially systems such as CiteULike that attracts academic professionals, or Delicious with its technical experts – there has been growing interest in utilizing data from these systems to identify users with expertise and the resources they bookmark. The most popular public social tagging systems lack explicit mechanisms for users to proclaim or verify each other's expertise in a particular domain based on tag or resource selection. A few papers have explored ways to implicitly determine user expertise based on a user's tagging patterns. Others have proposed mechanisms that social tagging systems could employ to help the community identify authoritative users and resources. Feinberg (2006) discusses how a user's level of domain expertise may influence the form and semantics of their tags. Van Setten et al. (2006) argue that User A may find User B's annotations more relevant to their goals and needs if they knew who User B was. John and Seligmann (2006) propose a PageRank-based algorithm, ExpertRank, for measuring expertise in an *enterprise* social bookmarking system. Their mechanism assigns an authority weighting based on the number of resources a user has contributed to a particular tag, and by propagating that weighting to highly-related tags. The authors intended to use ExpertRank as one component in determining the expertise of a user in a closed enterprise system containing additional background information on all users.

Finding expertise in public social bookmarking systems is a greater challenge, because they typically lack detailed profiles about users' backgrounds and the institutional controls of an organization or enterprise to discourage abuse of the system, or "spamming". Similar to link farms that attempt to game Web search engines, spammers of social bookmarking systems create hundreds or thousands of bookmarks promoting their own content with popular tags, or listing popular resources with misleading tags, often under multiple user accounts. A few studies propose algorithms and techniques to identify and analyze the effects of malicious tagging behavior. Koutrika et al. (2008) examine how spamming affects different types of social tagging systems and social search models, using both a synthetic dataset and a sample Delicious dataset. They introduce a metric, SpamFactor, to measure the impact of malicious tagging behavior on search results from social tagging data. They conclude that all social tagging systems can tolerate a spam threshold of approximately 15-20% of posts before search performance deteriorates significantly. Systems that allow multiple users to produce their own tag sets on a resource (e.g., Delicious) are less susceptible to spamming than systems that do allow such duplication, requiring users to collectively annotate the resource in a single set (e.g., YouTube, Flickr.) Among the systems that allow tag duplication, those with 1) a small core of active, responsible users and 2) no limit on the number of tags per post – thus encouraging duplication of "good" tags by many users - are less susceptible to spamming than systems with low activity and few tags per bookmark. Search models based on tag coincidences among users and resources, akin to a graph-based model like PageRank, are also less susceptible to spamming than Boolean or tag frequency models.

Table 7. Noll et al.'s classification of experts and spammers in a social bookmarking system

Type of Expert	Description
Geek	A user who is among the most active bookmarkers, and tends to be among the first to bookmark popular resources. These are the "best" experts.
Veteran	Similar to a "geek", but not as active; has significantly more bookmarks than the average user, but significantly less than a geek; is also likely to be among the first to discover popular resources.
Newcomer	A newer user who occasionally discovers new resources, but mainly tags popular resources long after they have been discovered.
Type of Spammer	**Description**
Flooder	A user who bookmarks thousands of popular resources, usually in an automated fashion (i.e. hundreds or thousands of bookmarks posted on the same day); always bookmarks resources long after they have become popular.
Promoter	A user who bookmarks many of their own resources (e.g., postings on their blogs), but has few followers, if any; tends to ignore popular resources.
Trojan	A user who mimics regular users (such as a "newcomer"), but also adds bookmarks to their own malware-infected or phishing resources

Noll et al. (2009) introduce a graph-based algorithm, SPEAR (*SPamming-resistant Expertise Analysis and Ranking*), to produce ranked lists of users in a social bookmarking system that promotes experts on a given topic while demoting spammers. The authors argue that users with great expertise not only identify high-quality resources on the Web, they bookmark them with good descriptive tags before other users. Thus, SPEAR is an extension of HITS, but gives more weight to users who annotate a resource with a given query tag (or set of tags) before others. By injecting simulated users representing different types of experts and spammers (Table 7) into a sample Delicious dataset, the authors show that SPEAR performs significantly better than HITS and simple tag frequencies in ranking experts ahead of spammers. Noll et al. focused solely on the ranking of users; how well SPEAR ranks resources remains an open question.

2.3 CLASSIFICATION IN SOCIAL ANNOTATION SYSTEMS

The aspects of social annotation systems that have made them so successful in terms of user adoption – low barriers to entry, low cognitive load when annotating, no rigid classification rules or controlled vocabulary to follow (Trant, 2009) – also make resource discovery very difficult. The joys of serendipitous browsing aside, social annotation systems can become more useful as sense-making tools if some semantic structure(s) could be teased from the plethora of seemingly unstructured annotations. One major research direction within the area of social annotations and the Social Web is the need to organize and classify tags within various types of semantic structures, including topic maps, hierarchies, ontologies, and faceted classifications.

Many studies have attempted to build classification schemes either using modified or un-modified versions of well-known data clustering algorithms. Some of the most efficient machine learning algorithms, such as self-organizing maps (Choy and Lui, 2007) and K-means clustering (Gemell et al., 2008), produce the worst results if left un-modified, due to their inability to cope with the vocabulary problems associated with social tags. K-means also suffers from the fact that researchers must specify a fixed number of clusters *a priori*, often resulting in either few clusters that are too broad, or many single-tag clusters. Conversely, Gemell et al. demonstrated that maximal complete link clustering produces superior classifications of Delicious data, but is too computationally expensive to scale well to large datasets. In a paper using social annotations to improve indexing for Web retrieval, Ramage et al. (2008) found that adding tags naively to indexed Web page text improves K-means clustering, but concatenating the word and tag vectors for a particular resource allows the author's Multi-Multinomial Latent Dirichlet Allocation (LDA) algorithm to significantly outperform K-means. Krestel, Fankhauser and Nedjl (2009) compared LDA's ability to find and recommend tags belonging to latent topics

to an association rules-based approach proposed by Heymann, Ramage, and Garcia-Molina (2008.) The authors concluded that LDA's tag recommendations were more accurate and specific than those identified by association rules. Begelman et al. (2006) used a far simpler approach, generating clusters from pairs of strongly-related tags based on tag co-occurrences that are more frequent than expected.

The most useful clustering algorithm from the literature appears to be hierarchical agglomerative clustering, which iteratively combines many clusters – each initially containing one item – into a single monolithic cluster containing all items. Hierarchical agglomerative clustering is more computationally efficient than many other algorithms, and has more flexible tuning capabilities. Kome (2005) shows that a large proportion of tags in Delicious fit the hierarchical relationships as defined in the appropriate ANSI/NISO and ALCTL taxonomy standards. Heymann and Garcia-Molina (2006) interpreted this as meaning users annotate resources with tags at multiple levels of their personal mental models, a key notion underlying their hierarchical clustering algorithm. Work by Brooks et al. (2006), Li et al. (2008), and Gemell et al. (2008) also showed hierarchical agglomerative clustering to be effective for building taxonomies and improving personalized search. Li et al.'s work is of particular interest here, as they found tag traces (subsets of 2 or more co-occurring tags) generated from rule-based associations often found in data mining applications produced better results than the tag pairs used by Brooks et al. (2006.) Their dataset was also significantly larger (200,000 users; 4.3 million bookmarks) than those of Brooks et al. and Gemell et al.

Some studies have sought to combine social annotation data with more formal semantic structures, namely ontologies. Mika (2007) discusses and evaluates two lightweight ontologies constructed from Delicious social annotations linking actors (users) and concepts (tags), where a

link's weight is the number of times an actor has a concept as an annotation. In the first ontology, semantic relations are formed between two concepts if they share many common resources. In the second ontology – and the one deemed more accurate by expert judges in an evaluation - concepts are linked semantically if they share many users in common. Mika concludes that identifying communities of interests may yield the best ontological structures. Specia and Motta (2007) generated clusters of tags from Delicious and Flickr by computing an $N \times N$ co-occurrence matrix of all tags, then using cosine similarity on the resulting tag vectors (i.e. matrix rows and columns) to find similar tags. They then queried Swoogle, the semantic web search engine, with tag pairs from each tag cluster to see if their clusters could be mapped to existing ontological concepts. The authors were able to map some tag pairs to existing ontologies, though the number of pairs was small (under 20%), the majority of which only mapped to nodes in WordNet. We also note that the authors performed some simple pre-preprocessing of the data, combining morphologically highly-similar tags into one group via the Levenshtein distance and removing idiosyncratic and non-alphanumeric tags.

3.0 PRELIMINARY ANALYSIS

3.1 INTRODUCTION

The objective of this study is to develop an algorithm that can identify experts and authoritative documents in social bookmarking systems more efficiently and more accurately than existing algorithms. To accomplish this goal, we need to define the following within the context of public social bookmarking systems:

1. *Who is an expert user?*
2. *What is an authoritative resource?*
3. *What tags describe the topic of a resource, or a user's topic of interest?*

This chapter begins by summarizing key observations made on social bookmarking systems that can help us not only identify and rank experts and authoritative documents, but also reduce the size of the data set as one part of improving computability. Other characteristics of tagging in social bookmarking systems, such as tag frequencies and tag co-occurrences, are useful for describing resources' topics or users' topics of interest. After defining expert users and authoritative resources in public social bookmarking systems, we describe the Expert and Authoritative Resource Locator (EARL) algorithm developed in this study for selecting and

ranking candidate experts and resources by topic of interest. The chapter concludes by presenting the results of preliminary analyses with a partially-implemented EARL laying the groundwork for the main studies.

3.2 SOCIAL BOOKMARKING SYSTEMS

This section begins with a few observations about social bookmarking system:

1. Users bookmark resources that are relevant to their interests. Thus, a bookmark is a positive "vote" for a resource.

2. Users may create at most one bookmark for a given resource, and cannot use a given tag on a given resource more than once. More importantly, users cannot explicitly assign weights to tags based on importance.

3. Users create tags for a variety of purposes, including tags to describe the topic of a resource (e.g., "programming"), personal ratings of the resource (e.g., "****"), and personal tags that are idiosyncratic to the user (e.g., "IS3925").

4. Public social bookmarking systems, such as Delicious, have no controlled vocabulary. Users may enter any string of printing characters as a tag. Whitespace indicates a tag boundary.

3.2.1 Usage Patterns

Early research on social bookmarking systems (Shirky, 2003; Vander Wal, 2005; Golder and Huberman, 2006; Millen and Feinberg, 2006) showed that these systems typically evolve as scale-free networks whose structures follow power laws, much like the World Wide Web from which their content is derived. Figure 3 shows a frequency distribution of the number of bookmarks per user from our preliminary Delicious dataset of 30,159,279 bookmarks made by 723,342 users on 12,815,856 unique resources, hereafter referred to as the study's *preliminary main dataset*. Nearly 92% of the users in the preliminary main dataset (664,783 of 723,342 users) have less than ten bookmarks in their accounts, while 1.1% (8,141 users) have more than 1000 bookmarks. Resources exhibit a similar power curve – only 0.3% of the (41,643 out of 12.8 million) have been bookmarked by more than 1000 users.

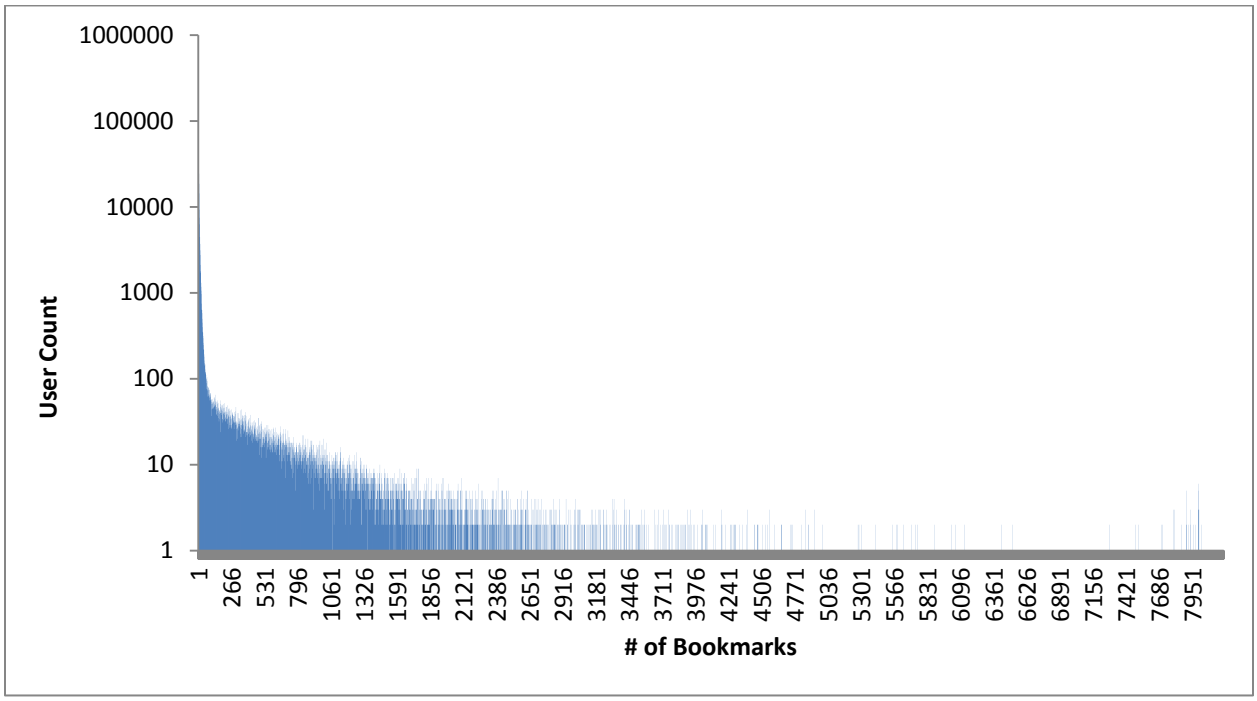


Figure 3. Frequency-rank distribution of the number of bookmarks per user for all users in the preliminary main dataset.

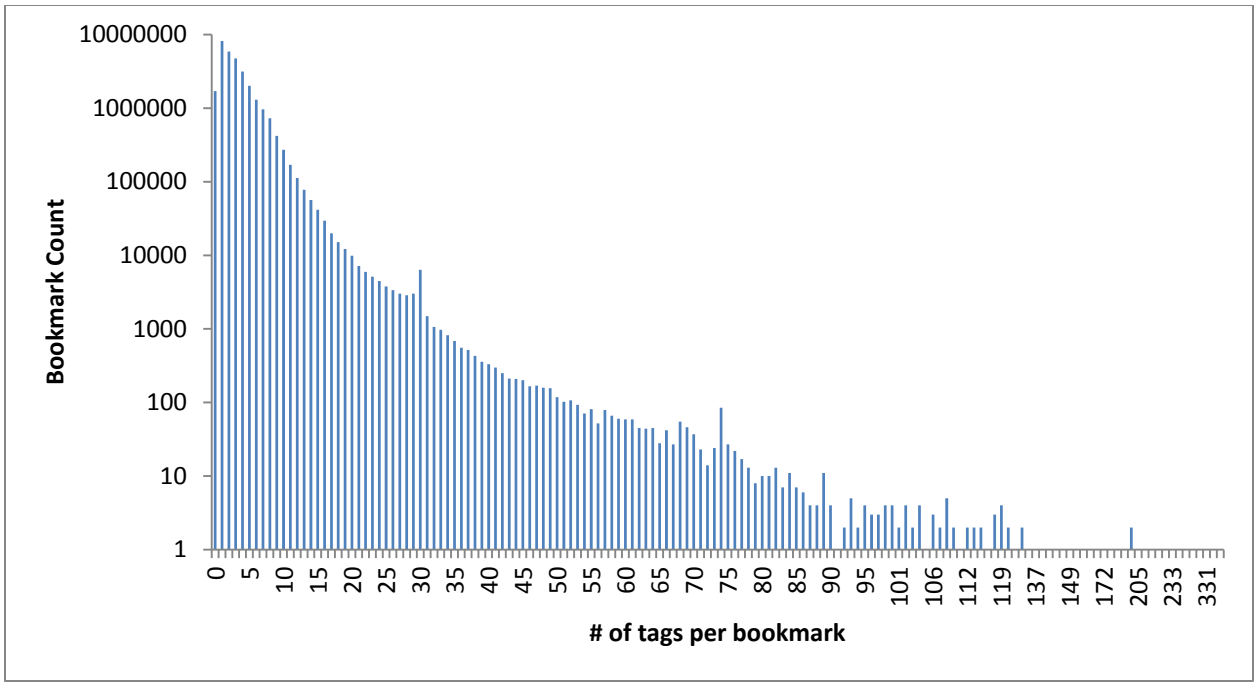


Figure 4. Frequency-rank distribution of the number of tags per bookmark for bookmarks in the preliminary main dataset.

Despite the lack of controlled vocabulary, tags also show similar patterns of usage in terms of tags per bookmark, and tag frequencies per user and resource. As seen in Figure 4, the number of tags per bookmark shows a clear power law distribution starting at one tag. Sixty-three percent of the bookmarks in the preliminary main dataset have 1-3 tags, with 79% of the bookmarks having less than 5 tags. For comparison, Kipp and Campbell (2006) found that 65% of the users in their sample Delicious dataset annotated a bookmark with 1-3 tags. Udell (2005), Golder and Huberman (2006), and Millen and Feinberg (2006) found that a user's tag vocabulary stabilizes over time, while Golder and Huberman observed that a resource's tag distribution tends to stabilize after 100 bookmarks. Delicious' interface promotes reuse of tags during the bookmark creation process, presenting the seven most popular tags on a resource. Dellschaft and

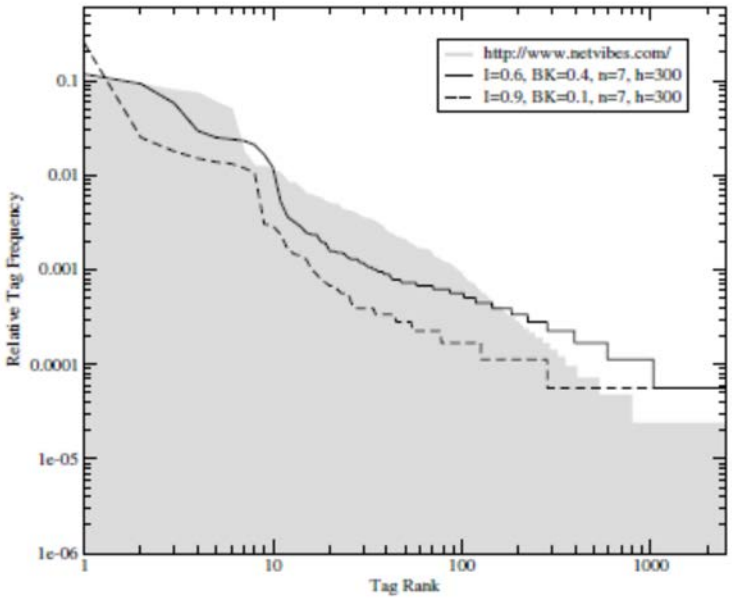


Figure 5. Dellschaft and Staab's (2008) comparison between the actual frequency-rank distribution of tags on the NetVibes home page (shown in grey), versus simulated tag stream models (dashed and solid lines) assuming users see the top 7 most popular tags as they enter their own tags.

Staab (2008) provide evidence (Figure 5) that Delicious' suggestions influence tag selection for a given resource over time, creating a pronounced drop-off after the seventh-ranked tag.

These findings suggest that we can extract a small subset of the preliminary main dataset's graph comprised of active users and resources to find expert and authoritative documents – most Delicious users and resources lack the requisite bookmarks to be identified authorities on some domain. Reducing the nodes in this manner allows us to analyze it more efficiently with less computational complexity while still maximizing the chances of finding domain experts and authoritative documents. Another goal of this study is to find users who do a consistently good job of providing tags that accurately describe the topics of resources. Whether this can be done is less clear. Collectively, users appear to reach a consensus over time about how to describe a given resource, making tags attractive as topical terms. On an individual basis,

however, most users annotate their bookmarks with few or no tags. Again, this suggests that we can focus on the subset of users who consistently use some minimal threshold of tags to describe a *resource*, but it is not clear how many tags are necessary to accurately describe a particular *topic*.

3.2.2 Topics of Interest in Social Bookmarking Systems

We observe several factors that influence what tags make good topic descriptors, and the number of tags needed to accurately classify a resource:

- Resources may be about a single topic (e.g., Roy Fielding's dissertation on REST), or multiple topics (e.g., the W3Schools homepage with tutorials on many web design and development technologies.)

- A single tag may suffice to represent a topic (e.g., "programming"), or multiple tags may be necessary (e.g., "graphic" and "design").

- Because social bookmarking systems do not allow spaces within tags, users concatenate multiple words with strong semantic ties into a "compound" tag. Compound tags may represent phrases or proper nouns found in natural language (e.g., "BillGates"), or represent hierarchical structures (e.g., "programming/java").

- Users commonly annotate resources with tags at multiple levels of categorization, using tags that describe a broad topic (e.g., "programming") along with tags that describe more specific topics (e.g. "java") or even highly-specialized topics (e.g. "jsp").

- Semantically-related tags co-occur frequently in the tag sets of many users.

Li et al. (2008) examine how many tags are necessary to describe a topic. The author's Internet Social Interest Discovery (ISID) algorithm used frequent tag co-occurrences across multiple users to form tag clusters that represent users' topics of interest. They conclude that 1-5 tags can fully describe a *single* topic; anything beyond six tags lacks enough of a consensus to be a reliable topic description. Given that a resource may be about multiple topics, this suggests that a bookmark should contain multiple tags to accurately describe the resource's content.

3.3 FINDING EXPERTS AND AUTHORITATIVE RESOURCES

3.3.1 Defining experts and authoritative resources

A common thread in the literature on expertise in social bookmarking systems is the notion that expertise may be derived from a combination of the user's analytical skills and domain knowledge. Therefore, we define an expert in a social bookmarking system as someone who has both *classification expertise* and *domain expertise*. A user with classification expertise, much like a traditional librarian, carefully selects tags that accurately summarize a resource's content. They are conscientious annotators, consistently applying tags to all of their bookmarks. Though their effort is most likely for personal organization and recall, their tag selections help others in the community find useful resources through social search – much like a librarian who carefully places resources in categories where patrons expect to find those resources. We define a domain expert in a social bookmarking system similarly to Noll et al. – a user who bookmarks many high-quality, authoritative resources on a topic, and is among the first to bookmark those resources. Domain expertise derives from a combination of the quantity and quality of the

resources in their bookmark collection. Some top domain experts may reach their position through sheer bookmarking volume – think of an academic who publishes conference papers prolifically – or by introducing a smaller number of highly-influential resources – the academic whose occasional journal publications become widely-cited.

This study makes the distinction between classification and domain expertise, because we observe that not all domain experts are good classifiers, nor are all good classifiers necessarily domain experts. Many librarians have no expertise on the topics of the resources they catalog, but they know what portions of the resource to look at (e.g., title, table of contents, publisher's notes on the jacket) in order to choose good classification terms. In social bookmarking systems, newer users can easily choose good descriptive tags based on others' tag assignments without understanding the underlying resource. Conversely, some users consistently discover and bookmark authoritative resources, but annotate the bookmarks with idiosyncratic tags, or no tags at all.

This brings us to the next question: what is an authoritative resource? This study defines an authoritative resource as any document that is a valuable source of information on some topic, according to the social bookmarking community. An authoritative resource may be a document with core information about a topic (i.e., an "authority", as defined by Kleinberg, 1999) or a collection of useful links on a topic (i.e., a "hub".) Like experts, a resource derives its authoritativeness based on the number of top experts who have bookmarked the resource, and its topical authority on the number of top experts who annotated those bookmarks with the tag(s) that represent a given topic. Given this mutual reinforcement between experts and authoritative resources, this study introduces and implements the graph-based EARL algorithm to find top

users and resources based on the social bookmarking system's link topology. The next section discusses EARL and how we view a social bookmarking system as a network.

3.3.2 EARL algorithm

The EARL algorithm finds expert users and authoritative documents in a social bookmarking system through a two-stage process that is conceptually similar to Hilltop (Bharat and Mahaila, 2000.) In the first stage, we reduce the number of nodes to a much smaller subset of influential users, who are referred to as "candidate experts". We reduce the nodes by using a series of simple statistics to locate users with consistent patterns of system usage and tagging behavior. We refer to this filtered subset of bookmarks as the *preliminary candidate expert dataset*. In the second stage, we select a topic and use a graph analysis scheme similar to HITS and SPEAR to rank candidate experts and authoritative documents. To find topics, we look for frequent tag co-occurrences shared by multiple candidate experts. We find tag co-occurrences by computing the power set of each bookmark in the preliminary candidate expert dataset, tabulating the subsets within each power set. We then select those subsets that 1) have tags that co-occur more frequently than a defined threshold, and 2) are shared by multiple candidate experts. Each stage is explained in more detail below.

Stage 1: Finding the Influential Users on Delicious

We observed in a preliminary manual examination of users' bookmarks in the preliminary main dataset that some Delicious users appear and disappear very rapidly – experimenting briefly with the system before abandoning it. Another group of users periodically and consistently adds bookmarks to Delicious, but annotate their bookmarks with few tags. We

expect a candidate expert, one who is conscientious about applying terms to describe a resource, to use several tags. For users with few bookmarks and/or few tags per bookmark, we cannot draw any reliable conclusions about their domain or classification expertise; thus, we immediately eliminate any user who 1) has less than 10 bookmarks, or 2) uses less than 5 tags per bookmark, on average. As a result, we eliminated 97.7% of the users in the dataset, leaving a list of 16,981 *candidate experts*. We emphasize that the first stage's goal is to maximize the density of potential (i.e. "candidate") experts in the remaining subset. Some non-experts may remain in the subset, while some experts may have been excluded.

This first stage identifies a much smaller subset of influential users who have created enough bookmarks to be potential domain experts *and* use enough tags to potentially be classification experts. However, we still cannot make any qualified judgments about each

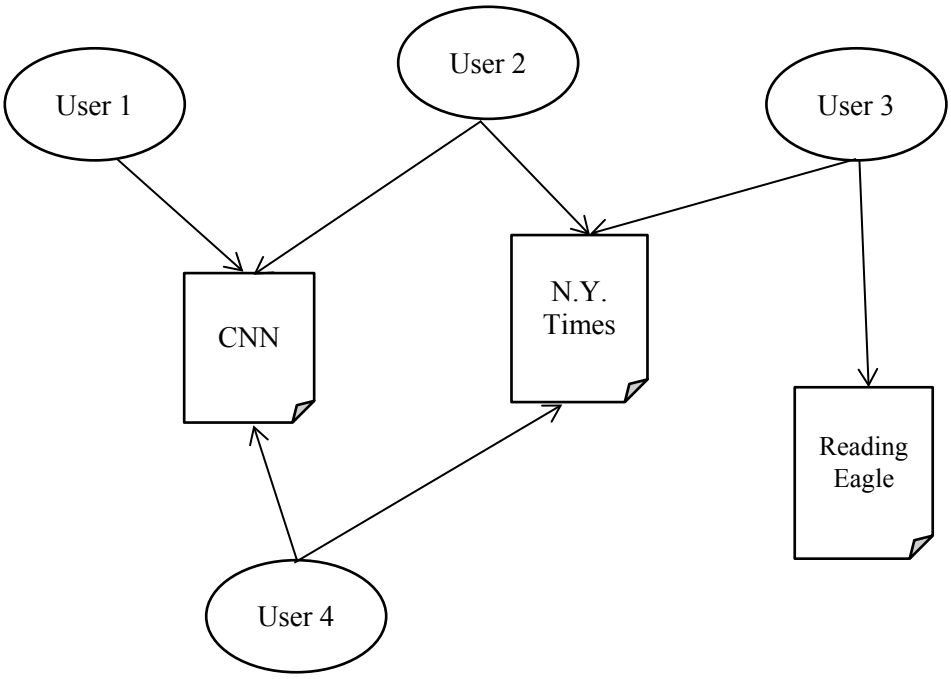


Figure 6. A partial view of a social bookmarking system as a graph. All edges (i.e. bookmarks) are directed from users to resources.

candidate's expertise, because we do not know what resources they have bookmarked or what tags they have used (and how often) on those bookmarks. Who among these candidates are the best experts on "Java programming", "Android development", or some other topic? The next step is to rank the candidate experts by topic based on the number and quality of accurately tagged resources each has bookmarked.

Stage 2: Identifying topical experts and authoritative resources

In this stage, we use an iterative graph-based algorithm similar to HITS and SPEAR to rank experts and authoritative resources by topic. Like Noll et al, we view the topology of a social bookmarking network as a directed graph with two distinct types of nodes, users and resources, with all edges pointing directly from users to resources (see Figure 6.) By creating a

ComputeEARL(Topic *T*)
- Retrieve all bookmarks B_T annotated with *T* from the expert dataset.
- Sort B_T by resource identifier, date bookmarked.
- Set a vector of expertise scores $\vec{E}$ to (1,1,1,…1) with *M* experts.
- Set a vector of authority scores $\vec{A}$ to (1,1,1,…1) with *N* resources.
- For each bookmark b_T in B_T:
  - Set the weight *w* of b_T.
  - Add (b_T, *w*) to the adjacency list of inlinks L_i.
  - Add (b_T, *w*) to the adjacency list of outlinks L_o.
- For *k* iterations, where *k* = 25:
  - Compute authority scores $\vec{A}$ from $\sum E \times L_i$
  - Compute expertise scores $\vec{E}$ from $\sum A \times L_o$
  - Normalize $\vec{E}$.
  - Normalize $\vec{A}$.
- Return list of authoritative documents sorted by authority score in $\vec{A}$.
- Return list of experts sorted by expertise score in $\vec{E}$.

Figure 7. Pseudocode for the second stage of EARL.

bookmark, the user creates an outlink to a resource, but the reverse is not possible – resources cannot bookmark users. Borrowing Kleinberg's terminology, social bookmarking users act as *hubs* to a collection of resources on a particular topic, while resources containing useful information on the topic act as *authorities* with inlinks from one or more users.

As shown in the pseudocode in Figure 7, we begin by choosing a topic *T*, where one or more tags $\{t_1, t_2 \ldots t_n\}$ represents T, such that:

$$\{t_1, t_2 \ldots t_n\} \in T \quad (1)$$

We select all bookmarks B_T annotated with T from the candidate expert dataset sorted in chronological order, where each bookmark b_T is a tuple comprised of a Delicious username *u*, resource identifier *r*, the topic *T*, and the creation date of the bookmark *d*:

$$b_T = (u, r, T, d) \quad (2)$$

We then define two vectors: $\vec{E} = (e_1, e_2 \ldots e_M)$ to hold the *expertise scores,* where *M* is the number of unique candidate experts in B_T, and $\vec{A} = (a_1, a_2 \ldots a_N)$ to hold the *authority scores,* where *N* is the number of unique resources in B_T. The expertise score of a candidate expert *u* depends on the sum of the authority scores of the resources tagged with *T* in his collection, while an authority score of a resource *r* depends on the sum of the expert scores of the candidate experts who annotated the resource with *T*. All scores in $\vec{E}$ and $\vec{A}$ are initialized to 1.

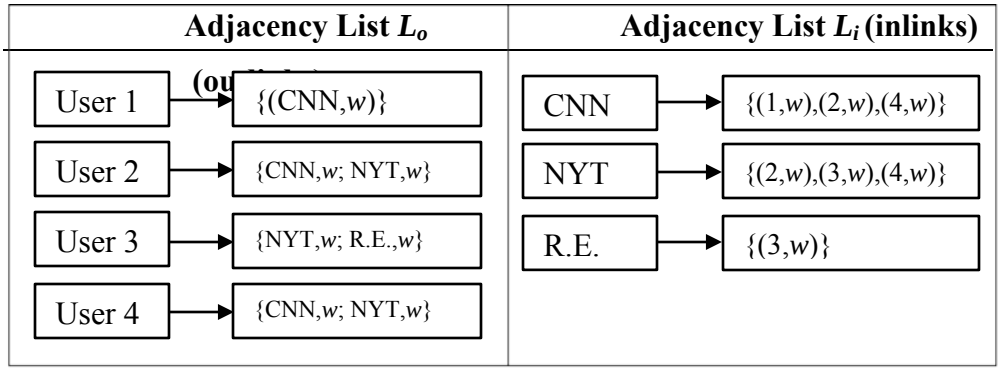


Figure 8. Outlink and inlink adjacency lists used in EARL.

To calculate the scores based on this mutually reinforcing relationship, we run an iterative process similar to HITS and SPEAR. Due to the sparseness of the preliminary candidate expert dataset and limitations of the machine used for this preliminary analysis, we set up two adjacency lists (Figure 8) in lieu of an adjacency matrix to hold the outlinks of all candidate experts (L_o) and the inlinks of all resources (L_i) from B_T. Thus, we only add links to the lists if an edge (i.e., bookmark) exists between a candidate expert and a resource. If we modeled the algorithm directly after HITS, each link would be assigned a weight of 1. In EARL, we assign a positive weight w to the bookmark made by user u_i on resource r_j annotated with T based on additional criteria to measure the bookmark's quality as shown in Equation 3:

$$w_{u_i,r_jT} = \sqrt{\left(B_{r_jT} - m_{d-1} - \frac{n-1}{2}\right) \times \frac{B_{r_jT}}{B_{r_j}} \times \sqrt{B_{r_jd'}}} \quad (3)$$

where B_{r_jT} is the number of bookmarks made on r_j annotated with topic T; m_{d-1} is the number of bookmarks on r_j and annotated with T created before the day u_i bookmarked r_j; n is the number of users who bookmarked r_j on a given day; B_{r_j} is the number of bookmarks made on r_j; and $B_{r_jd'}$ is the number of bookmarks made on r_j since the date d'. For the preliminary analysis, we

set d' to August 5, 2009, which is six months prior to our final collection date of February 5, 2010.

We now describe in greater detail the four criteria used to establish the weights for each link in EARL's adjacency lists:

1. *Temporal sequence*: users who bookmark a resource first with the given topic tag(s) get more credit, an idea adopted from SPEAR. As Noll et al. explain, the best experts are the people who not only have a good command of the literature in their field, but also discover (or even contribute) top resources before others, a more challenging task than adding a bookmark to a resource that is clearly popular. Thus, the links of discoverers in B_T are assigned higher weights than followers. The temporal sequence portion of EARL's weight is based on the number of bookmarks *m* made prior to user u_i creating his/her bookmark.

2. *Normalized expert agreement*: the greater the percentage of candidate experts who applied *T* to a resource, the more credit is given to the link. The goal here is to improve the rankings of resources where *T* is a central topic. For example, suppose 100 candidate experts bookmarked Resource A, and 1,000 candidate experts bookmarked Resource B. Overall, Resource B is the more popular resource. However, suppose 80 of the 100 candidate experts who bookmarked A annotated their bookmarks with *T*, while 200 of Resource B's 1,000 annotators used *T*. Because a greater percentage of users (80%) believe Resource A is about *T* than Resource B (20%), EARL gives greater weight to Resource A's bookmarks.

Table 8. An example illustrating the calculation of the temporal sequence portion of EARL's weight, factoring in daily bursts of activity.

User	Date Bookmarked	m	m_{d-1}	n	B_{rjT}-m	B_{rjT} - m_{d-1} - $(n-1)/2$
User	January 1, 2010	0	0	1	7	7
User2	January 18, 2010	1	1	1	6	6
User3	February 1, 2010	2	2	4	5	3.5
User4	February 1, 2010	3	2	4	4	3.5
User5	February 1, 2010	4	2	4	3	3.5
User6	February 1, 2010	5	2	4	2	3.5
User7	March 15, 2010	6	6	1	1	1

3. *Sustained popularity of a resource*: resources that continue to be bookmarked regularly carry a higher weight than resources that may have been popular in the past, but are no longer favored by the community. Once-popular resources that are no longer bookmarked by experts – for example, defunct search engines such as Cuil and Powerset.com – should have lower authority scores relative to actively-bookmarked resources. We also view this as a way to measure the expertise of a user: people who are considered experts are those who have a strong command of information *currently* deemed most useful by the community. Even if the expert is no longer active, we still consider him or her a valuable hub of information if the experts' bookmarked resources continue to be tagged routinely by others.

4. *Extreme bursts of activity*: in cases where hundreds or thousands of users bookmark a particular resource in a single day, credit is distributed equally among the users for that day. Table 8 shows an example of how this portion of EARL's weight is calculated given a resource's temporal bookmarking sequence among seven users, where Users 3 through 6 bookmarked the resource on the same day. Resources that experience these brief, but very intense bursts of popularity typically do so because of a response to some

external factor – e.g., widespread blog and mainstream media coverage of an event. The users who bookmarked the resource earliest could very well be experts, but it is just as likely they are average users who happened to be the first to respond to the external event. From the perspective of the EARL algorithm, these bursts of activity skew the importance of temporal sequence – why should the first user to bookmark a resource receive so much more credit than the 200th user who bookmarked the resource a mere two hours later? Thus, all users who bookmark a given resource with topic T on the same day receive the same temporal sequence weighting – i.e., the average of the original temporal sequence weights ($B_{r_jT} - m$) assigned to the first and last users to bookmark the resource on that day. The values in the last column of Table 8 for Users 3 through 6 reflect this weight calculation.

Finally, we run this portion of EARL for 25 iterations, then sort the expert and authority scores from highest to lowest.

3.3.3 Selecting topics of interest

Like HITS and SPEAR, EARL is a topic-dependent graph-based algorithm – it ranks experts and authoritative documents in the context of a pre-defined topic. Previous studies have selected topics from popular Delicious tags (Noll et al., 2009), frequently co-occurring Delicious tags (Heymann and Garcia-Molina, 2006; Li et al., 2008), or from external sources such as Open Directory Project categories (Ramage et al., 2009) and Library of Congress subject headings (Smith, 2007.) In this study, we follow Li et al. by examining frequently co-occurring tags among the candidate experts, because 1) tags describe the content of resources according to

users' judgments, and 2) multiple co-occurring tags can provide a more precise description of topics of interest than single tags alone (Li et al, p. 682.) While Li et al.'s ISID used association rules to extract topics, we use a simpler, distributed algorithm that constructs the *power set* of each user's bookmark and tabulates the number of times the elements of all the power sets – i.e., *tag subsets* - occur in the user's bookmark collection *and* with each other. The goal is to find tag subsets that co-occur frequently in the bookmark collections of multiple users – subsets of tags that represent shared topics of interest.

A power set is simply an array of all subsets of a set of elements *S*, including the empty set and *S* itself. Table 9 shows an example of a tag set containing three tags and the resulting power set consisting of eight elements. By definition, *S*'s power set contains 2^n items, where *n* is the number of elements in *S*; thus, if a bookmarks contains 50 tags, the subsequent power set will contain 1.1 *quadrillion* elements.

Table 9. The power set elements for a bookmark tag set consisting of the tags "css", "webdesign", and "tips"

Tag Set : { css, webdesign, tips }

#	**Subset**
1	{}
2	{ css }
3	{ webdesign }
4	{ tips }
5	{ css, webdesign }
6	{ css, tips }
7	{ webdesign, tips }
8	{ css, webdesign, tips }

Our goal is to take each bookmark in a user's collection, generate its power set, and add all *tag subsets* to a frequency table counting the subsets' occurrences throughout the user's bookmarks. Given the limitations of our hardware, we cannot possibly generate and store power sets for all bookmarks, particularly those with more than 32 tags (i.e., > 4 billion tag subsets.) Furthermore, users who have thousands of bookmarks, thousands of unique tags, and high tag-per-bookmark averages produce extremely large tag subset frequency tables – occasionally larger than we can store in memory. We found that for any given user, we can reliably produce power sets and store tag subsets for bookmarks containing 14 or fewer tags. We make no attempt to produce power sets from bookmarks with more than 14 tags. However, after processing all of a user's bookmarks with 14 or fewer tags, we compare and record any matching subsets in the tag subset frequency table and the tag sets of the bookmarks that have more than 14 tags. This method may miss some novel tag subsets within the heavily-tagged bookmarks by only considering existing subsets. Given that only 5% of the bookmarks in the candidate expert data set contain more than 14 tags, it is highly unlikely that we will ignore any meaningful shared topics of interest.

Figure 9 shows the psuedocode of the topic selection process. For each candidate expert, we generate the power set of each bookmark's tag set, tabulating the frequencies of all tag subsets as we go along. Note that given our processing rules, a tag subset contains at least one tag, but no more than 14 tags. After extracting tag subsets from the last bookmark, we eliminate any tag subset that occurs in less than 5 of the candidate expert's tag sets. The choice of five occurrences follows Bharat and Mahaila (2000), who only considered documents with at least 5 links on a given topic as candidate expert documents for Hilltop.

```
FindTopics()
    Retrieve all candidate expert usernames from the expert dataset.
    For each candidate expert:
        Retrieve the candidate expert's bookmarks.
            For each bookmark:
                If the bookmark's tag set has ≤ 14 tags:
                        Generate the tag set's power set.
                        Record each power set element (tag subset) & increment count.
                Otherwise, postpone processing of bookmark.
        Remove all tag subsets appearing in < 5 bookmarks.
        For each bookmark tag set containing > 14 tags:
            Find matches with recorded tag subsets & increment counts.
        Starting at n=1, do while n < 7, where n is the number of tags in a tag subset:
            Set the minimum frequency threshold τ, where 0.5 ≤ τ ≤ 1.
            Select all tag subsets with n tags.
            For each selected tag subset, s_n:
                Retrieve all tag subsets with n+1 tags that contain s_n.
                For each selected tag subset with n+1 tags, s_n+1:
                        Find the percentage of bookmarks tagged with s_n+1 vs. s_n.
                        Record s_n+1 if the percentage ≥ τ.
            Add all recorded tag subsets to the global topic table.
```

Figure 9. Pseudocode for EARL's topic selection approach.

Having filtered out infrequent tag subsets, we then identify tag subsets whose constituent tags frequently co-occur. We do this using an agglomerative, bottom-up approach that matches frequently co-occurring tag *pairs*, then tag *triples*, etc., up to a maximum of 6 tags (as per Li et al.) In other words, we start by finding broad topics described by a few tags, working our way to narrower topics described by several tags. Starting with $n = 1$, where n is the number of tags in a given tag subset, we select all subsets from the candidate's expert's tag subset table containing n tags. For each tag subset s_n, we then find any subset s_{n+1} containing $n + 1$ tags where $s_n \in s_{n+1}$. For example, if the selected s_n is {"ajax"}, occurring in 150 of the candidate expert's bookmarks,

we will grab the s_{n+1} subset {"ajax", "javascript"} occurring in 100 bookmarks. Then, for each selected s_n, we compute the normalized frequency of each s_{n+1} relative to s_n:

$$\frac{s_{n+1}}{s_n} \rightarrow \frac{Count\,\{"ajax",\ "javascript"\}}{Count\,\{"ajax"\}} = \frac{100}{150} = 0.67$$

To put the above example in plain language, when we focus on the candidate expert's 150 bookmarks annotated with the tag "ajax", we see that the tag "javascript" also occurs in 100 of those bookmarks, a two-thirds majority of the time. However, if we instead focus on the s_n {"javascript"} occurring in 400 of the candidate expert's bookmarks, we notice that only one-quarter of those bookmarks also contain "ajax":

$$\frac{s_{n+1}}{s_n} \rightarrow \frac{Count\,\{"ajax",\ "javascript"\}}{Count\,\{"javascript"\}} = \frac{100}{400} = 0.25$$

Our interpretation is that "ajax" and "javascript" are semantically related, where "ajax" is a narrow topic related to the much broader topic "javascript", given that "javascript" occurs in the majority of the bookmarks tagged with "ajax". But how large must the normalized frequency be to accept s_n as a topic of interest? In this study, we compare each normalized frequency to a minimum threshold τ, defined in Equation 4:

$$\tau = 1 - \frac{1}{2^n} \tag{4}$$

If *at least one* of s_{n+1}'s normalized frequencies relative to s_n is greater than τ, we keep s_{n+1} as a topic of interest; otherwise, we remove s_{n+1} from the tag subset table. We increase τ as n increases to reduce the effects of tag noise. Put another way, there is an inverse relationship between tag subset size and frequency in a user's bookmark collection – tag subsets with 5 or 6 six tags usually appear in less than 20 bookmarks. Given these smaller counts, we would be

more likely to erroneously associate unrelated tags to tag subsets if we use a constant threshold. In our example, because "javascript" appears in more than half of the candidate expert's bookmarks tagged with "ajax" ($\frac{s_n+1}{s_n} = 0.67 \geq \tau_{n=1} \rightarrow 0.5$), we keep the tag subset {"ajax", "javascript"} as a topic of interest for further processing. We perform this routine up to $n = 6$ for the current candidate expert, then add all frequently-occurring tag subsets to a global tag subset table. As the remaining candidate experts are processed, we increment the global tag subset frequencies when we discover overlapping topics of interest.

3.4 FINDING EXPERTS AND AUTHORITATIVE RESOURCES

Preliminary analyses using the expert and main datasets were performed to answer the following questions:

1. Is there evidence that candidate experts, as a whole, exhibit both domain and classification expertise?
2. What are the most popular topics, using the topic selection scheme of EARL?
3. How do the rankings of experts and authoritative documents of EARL compare to those of HITS and SPEAR?

3.4.1 Candidate Expert Tagging Patterns

In the first stage of EARL, candidate experts are selected from the main dataset based on two simple criteria, bookmark count (domain expertise) and average tags per bookmark (classification expertise). We eliminate users with less than ten bookmarks, because they

Table 10. Basic statistics for the preliminary main and candidate expert datasets.

	Main Dataset	Candidate Expert Dataset	Candidate Expert %
User Count	723,342	16,981	2.3%
Resource Count	12,815,856	2,076,391	16.2%
Bookmark Count	30,159,279	3,883,661	12.9%
Distinct Tag Count (ignoring case)	1,577,610	505,964	32.1%
Tag Instance Count	94,439,113	25,907,044	27.4%

provide too little information to reliably judge their domain expertise on any topic. While some of these users may actually be domain experts, we cannot verify the expertise of someone who does not share their knowledge. Similarly, we eliminate users who annotate their bookmarks sporadically with very few tags – we cannot tell if they are good classifiers if they use few or no tags on their bookmarks. After eliminating users who seldom bookmark or annotate their bookmarks, we are left with a small subset of users, resources, tags, and bookmarks for the preliminary candidate expert dataset (see Table 10.) To be clear, we do not expect every user in the candidate expert dataset to be a domain expert on one or more topics, nor do we expect each one to be an expert classifier. However, we believe there are some users who do qualify as both domain and classification experts. The first set of analyses explores the tagging patterns of the candidate experts on a per-resource basis to find evidence of the two types of expertise.

Observation 1

Candidate experts use tags on resources with similar relative frequencies as all users, but with greater agreement.

Table 11 shows tag frequency tables for three popular resources in Delicious, comparing the candidate experts' tag frequencies with those of all users in the preliminary main dataset. We observe that for popular resources bookmarked by at least 100 users in both the preliminary main and candidate expert datasets, the top seven tags by popularity are the same for 72.9% of the resources (1027 of 1408), though the rank order was typically different – only 9.5% of the resources had lists that were completely identical in tag composition and rank. However, the percentage of candidate experts who used each top *n* tags on a given resource (i.e., agreement) is always greater than the corresponding percentage among all users. It is possible that this greater tag usage agreement among candidate experts may simply be a by-product of the initial candidate expert selection process that focuses on prolific annotators. By filtering out users who use few or no tags, we remove most of the empty tag sets that contributed to the denominator of the tag usage percentage (i.e., count of users bookmarking the given resource) but not the numerator (i.e., number of users bookmarking and annotating the resource with the given tag), thus raising the percentages. Still, we believe the identical relative frequencies and greater agreement support the idea that the initial selection process helps isolate classification expertise. Not only are the candidate experts consistently using multiple tags on their bookmarks, but they are also using (*and* are more likely to use) tags that reflect the beliefs of the entire community.

Table 11. Comparison of the top seven tags, frequencies, and usage percentages in the preliminary candidate expert dataset versus the main dataset for three popular resources in Delicious.

Resource URL: script.aculo.us

Candidate Expert Dataset			Main Dataset		
Tag	***Use Count***	***Expert %***	***Tag***	***Use Count***	***All Users %***
javascript	591	87.8%	javascript	3571	67.7%
ajax	540	80.2%	ajax	2877	54.6%
programming	418	62.1%	web2.0	1429	27.1%
web2.0	385	57.2%	programming	1402	26.6%
web	342	50.8%	web	1086	20.6%
webdesign	321	47.7%	webdesign	1144	21.7%
css	280	41.6%	css	824	15.6%
Total Bookmarks:	**673**			**5272**	

Resource URL: kuler.adobe.com

Candidate Expert Dataset			Main Dataset		
Tag	***Use Count***	***Expert %***	***Tag***	***Use Count***	***All Users %***
color	414	82.6%	color	1609	56.3%
design	387	77.2%	design	1333	46.7%
webdesign	354	70.7%	webdesign	889	31.1%
tools	332	66.3%	tools	733	25.7%
adobe	325	64.9%	adobe	594	20.8%
graphics	268	53.5%	graphics	380	13.3%
colour	202	40.3%	colour	375	13.1%
Total Bookmarks:	**501**			**2857**	

Resource URL: www.alvit.de/handbook/

Candidate Expert Dataset			Main Dataset		
Tag	***Use Count***	***Expert %***	***Tag***	***Use Count***	***All Users %***
webdesign	446	80.4%	webdesign	2329	52.4%
css	439	79.1%	css	2230	50.2%
reference	385	69.4%	reference	1531	34.5%
web	353	63.6%	web	1312	29.5%
design	335	60.4%	design	1217	27.4%
development	310	55.9%	development	996	22.4%
html	293	52.8%	html	876	19.7%
Total Bookmarks:	**555**			**4444**	

Observation 2

Candidate experts, as a whole, introduce few resources and their corresponding popular tags to Delicious.

The initial candidate expert filtering process produces the candidate expert subset based on one element of domain expertise – number of bookmarks. As Noll et al. demonstrated, bookmark count alone does a poor job measuring expertise in a public social bookmarking system. To make a preliminary assessment of the level of domain expertise in the candidate expert dataset, we examined how many resources in the main dataset were introduced by candidate experts. We have complete histories for some, but not all of the resources in the preliminary main dataset; therefore, we selected the data for all resources from the preliminary main dataset 1) with complete histories and 2) bookmarked by at least 200 users – popular resources with stabilized tagging patterns that may serve as authoritative documents.

Table 12 shows the results of the analysis on the 1,678 resources with at least 200 bookmarks and complete histories in the preliminary main dataset. Of the 870,595 total bookmarks in this sample, candidate experts contributed 84,146, or 9.7%, well below their contribution of 12.9% of all preliminary main dataset bookmarks (Table 10). Candidate experts

Table 12. All user versus candidate expert bookmark contributions to resources in the preliminary main dataset with complete histories and ≥ 200 bookmarks (*n* = 1,678)

	Avg. per resource	All Users	Candidate Experts	Candidate Expert %
Count of All Bookmarks:	519	870,595	84,146	9.7%
Count of First Bookmarks:	-	1,678	157	9.3%
Count of Bookmarks until all top 7 tags appear:	28	46,988	3,808	8.1%

were the first to discover and bookmark 157 (9.3%) of the resources, also below their contribution percentage to the preliminary main dataset. If we expand the analysis to include all bookmarks made by early adopters – bookmarks made until each of the resources' top seven tags appear in at least one tag set – the percentage of candidate expert bookmarks falls to 8.1%. We conclude that the initial candidate expert selection process, relying on a single element of domain expertise (bookmark count), does not isolate domain experts in Delicious.

One reason for the low percentage of domain experts in the preliminary candidate expert dataset may be that users who qualify as good classifiers – i.e., multiple tags per bookmark – are more likely to be copiers than early adopters of popular resources, because they have the benefit of Delicious' tag suggestions to select good descriptive tags. Annotating bookmarks with multiple tags takes far less cognitive effort and analytical skill when the system's interface provides the top tags. A more optimistic interpretation is that the combination of domain and classification expertise is a rare breed in social bookmarking systems, as it is in the real world. If the initial selection process focused primarily on domain expertise – for example, selecting the first *n* bookmarks from popular resources and extracting the top *m* users who contribute the most bookmarks to this subset – we will likely find that few of the candidates exhibit classification expertise.

3.4.2 Topics of Interest

Before beginning studies with EARL, it is important to identify topics we can use to measure the domain expertise of specific candidate experts. Although we consider any tag based on dictionary terms (including compound tags) as potential topics, this exploratory analysis focuses on combinations of tags that describe topics. One goal of the topic extraction is to explore the

semantic relations among topics' constituent tags. For instance, popular tags that provide little informational content in isolation, such as "web" or "tools", become more useful for classification when combined with semantically related tags. Another goal of extracting topics of interest is simply to get a sense of the breadth and depth of the candidate experts' interests. Because we are using data from Delicious, we expect topics related to information technology to dominate the list – so much so that the technical bias is considered a limitation of this work.

Using the technique described in section 3.3.3., topics of interest were extracted from the bookmark collections of all 16,981 candidate experts. Overall, the candidate expert dataset contains 216,183 topics of interest comprised of at least two tags and with a minimum of two

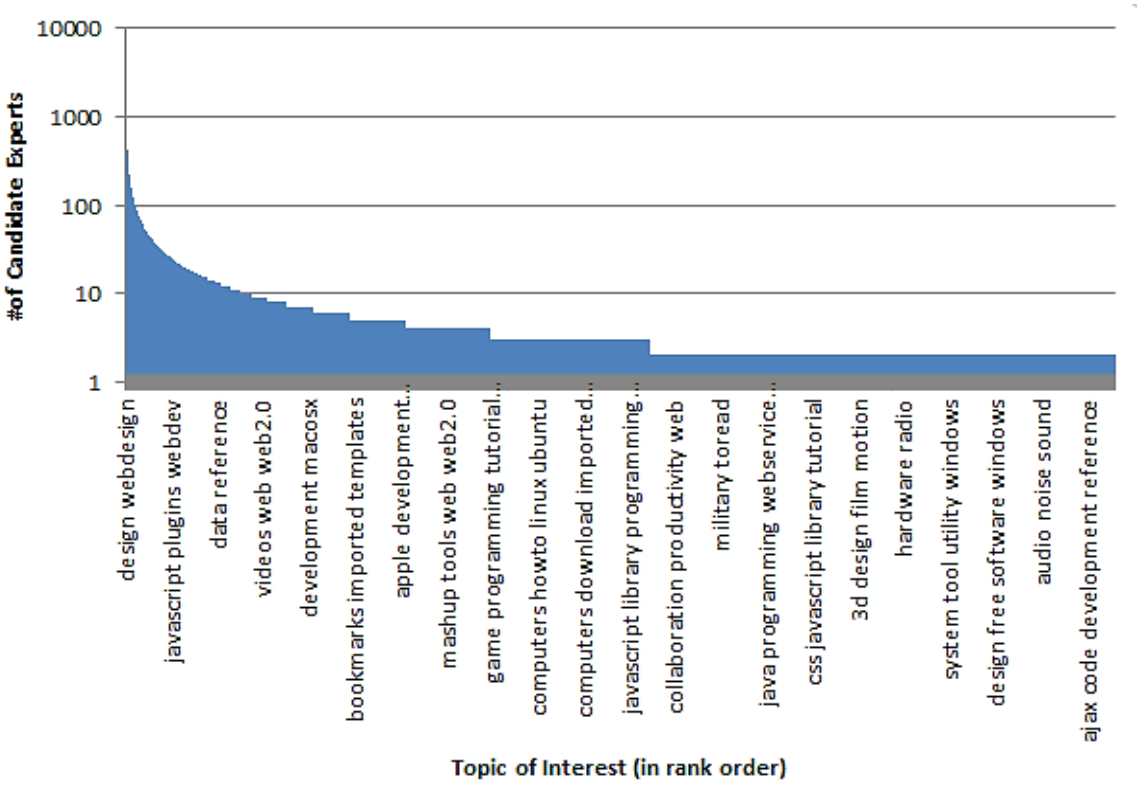


Figure 10. Frequency-rank distribution of topics of interest in the preliminary candidate expert dataset. Each topic listed on the horizontal axis represents a decrease of 10,000 in rank position.

users contributing to the topic. Figure 10 shows the frequency-rank distribution for all topics, where each topic listed on the horizontal axis represents a decrease in rank of 10,000, starting from the most popular topic "design webdesign" at rank 1. Even at logarithmic scale, the topic distribution shows a clear power curve; only 1,861 of the topics have at least 100 candidate experts contributing to the topic, given the topic extraction rules described in section 3.3.3. Table 13 shows the top forty topics of interest by candidate expert count, all of which are directly or indirectly related to information technology. In fact, an overwhelming majority of the topics in the complete list deal with information technology. Popular, non-technical topics such as "cooking food recipes" (rank 450) and "finance money" (rank 149) are rare, confirming our suspicions of poor topic coverage in non-technical domains. In both Figure 10 and Table 13, the tags within each topic of interest are listed in alphabetical order.

Looking at the topics in Table 13, we can infer some of the semantic relationships among the component tags. Although the reader may argue that we misclassified some of the following topics, we believe all of the topics consist of tags with strong semantic ties:

- *Named Entity*: "mac osx" (17), "apple mac" (31).
- *Parent-Child*: "design webdesign" (1), "javascript ajax" (11, order reversed), "software windows" (27).
- *Synonyms*: "development programming" (9), "fonts typography" (37).
- *Singular-Plural/Part-Whole:* "tutorial tutorials" (12), "blog blogs" (18), "tool tools" (29).
- *Compound tag with component tags:* "design web webdesign" (8).

For the purpose of this research, we enumerate the types of semantic relationships mainly to show that EARL's topic extraction method effectively finds and groups semantically-related tags together based on frequent co-occurrences.

Table 13. Top 40 topics of interest of candidate experts

Rank	Topic	# of Candidate Experts
1	design webdesign	2229
2	css webdesign	1886
3	design inspiration	1662
4	web webdesign	1560
5	css design	1455
6	design graphics	1430
7	css design webdesign	1379
8	design web webdesign	1316
9	development programming	1302
10	freeware software	1290
11	ajax javascript	1246
12	tutorial tutorials	1216
13	art design	1216
14	photo photography	1207
15	software tools	1156
16	photography photos	1155
17	mac osx	1118
18	blog blogs	1117
19	css web webdesign	1106
20	design typography	1081
21	webdesign webdev	1062
22	opensource software	1055
23	mp3 music	1048
24	css html	1048
25	howto tutorial	1019
26	design inspiration webdesign	1006
27	software windows	1005
28	funny humor	972
29	tool tools	966
30	tools utilities	940
31	apple mac	934
32	html webdesign	925
33	css design web webdesign	917
34	css design web	912
35	design web	908
36	css web	859
37	fonts typography	857
38	tools web2.0	813
39	css html webdesign	800
40	audio music	792

3.4.3 EARL versus HITS and SPEAR

The second phase of EARL introduces three factors to better identify experts and authoritative resources – normalized expert agreement, sustained resource popularity, and extreme bursts of activity – and adopts a fourth factor, temporal sequence from Noll et al.'s SPEAR algorithm. To gauge EARL's effectiveness, we implemented and conducted preliminary tests with HITS, SPEAR, and EARL on 25 topics of interest, including single-tag topics (e.g., "javascript".) Preliminary tests were run only on the candidate expert dataset. We present the results for one very popular topic, "design, web" (rank: 35) in Table 14, and one moderately popular topic, "rest webservices" (rank: 3940) in Table 15. Because Noll et al.'s research focused exclusively on expert rankings, these preliminary tests were the first opportunity we had to compare all three algorithms' abilities to rank both experts and resources.

We observe the following regarding the results shown in Tables 14 and 15:

- The expert rankings vary greatly between HITS and SPEAR/EARL, but vary little between SPEAR and EARL. For moderately popular topics, HITS tends to favor users with the largest number of bookmarks on the given topic, while SPEAR and EARL favor users who are early bookmarkers of popular resources, regardless of how many bookmarks they have tagged with the given topic query. This suggests that temporal sequence, introduced in SPEAR, is an important factor for ranking experts, but that EARL's factors have little effect.
- In terms of resources, all three algorithms seem to do a good job identifying resources *on topic*, even though we are only using tags and ignoring resource titles and content. For moderately popular topics, there tends to be a stronger correlation between rank and resource count in SPEAR and EARL than in HITS.

Table 14. Comparison of HITS', SPEAR's, and EARL's rankings of the top 10 experts and resources in the candidate expert dataset for the topic "design, web"

Topic: "design, web"			
	HITS	**SPEAR**	**EARL**
Rank	*User ID*	*User ID*	*User ID*
1	cristhianfs	cristhianfs	cristhianfs
2	Bhooshan	clouseau	clouseau
3	andysowards	ballicky	Boubahou
4	blackveins	everlaster	ceez
5	cmrsampaio	Boubahou	ballicky
6	cerasoli	ceez	everlaster
7	2raj	Elix	Elix
8	everlaster	chriskeane	adamharte
9	clouseau	chosco	chriskeane
10	dedesk	5ndime	chosco
Rank	*URL*	*URL*	*URL*
1	alvit.de/handbook/	alvit.de/handbook/	alvit.de/handbook/
2	www.smashing-magazine.com/2007/...	www.smashing-magazine.com/2007/...	www.oswd.org/
3	browsershots.org	browsershots.org	webdesignfromscratch.com
4	www.oswd.org	www.oswd.org	www.smashing-magazine.com/2007/...
5	webdesignfromscratch.com	webdesignfromscratch.com	browsershots.org
6	www.csszengarden.com	www.csszengarden.com	www.csszengarden.com
7	960.gs	alistapart.com	alistapart.com
8	www.cssbeauty.com	960.gs	bestwebgallery.com
9	typetester.maratz.com	www.cssplay.co.uk	960.gs
10	alistapart.com	bestwebgallery.com	www.designmeltdown.com

- Although SPEAR's and EARL's resource rankings only differ at rank 9 for the topic "rest webservices" in Table 15, their tends to be more variability in their resource rankings on par with the results for "design, web" in Table 14, regardless of resource popularity.

Note that for this preliminary study, two aspects of the EARL algorithm are not implemented: sustained popularity of a resource, and extreme bursts of activity. We believe both will affect EARL's expert and resource rankings, especially those of rapidly-evolving topics in which once-

Table 15. Comparison of HITS', SPEAR's, and EARL's rankings of the top 10 experts and resources in the candidate expert dataset for the topic "rest webservices"

Topic: "rest, webservices"			
	HITS	**SPEAR**	**EARL**
Rank	*User ID*	*User ID*	*User ID*
1	dhinchcliffe	clouseau	clouseau
2	bruce.healy	domix	bcp
3	clouseau	behruz	domix
4	bcp	divadsirrah	behruz
5	cmrsampaio	detobin	divadsirrah
6	domix	bcp	detobin
7	behruz	CAStrauss	CAStrauss
8	drawkbox	colin.surprenant	colin.surprenant
9	berberich	cpjobling	durdn
10	evangineer	durdn	dobersch
Rank	*URL*	*URL*	*URL*
1	www.ics.uci.edu/~fielding/pubs/dissertation/top.htm	www.xfront.com/REST-Web-Services.html	www.xfront.com/REST-Web-Services.html
2	www.xml.com/pub/at/34	www.restlet.org/	www.restlet.org/
3	www.xml.com/pub/a/2004/12/01/restful-…	www.ics.uci.edu/~fielding/pubs/dissertation/top.htm	www.ics.uci.edu/~fielding/pubs/dissertation/top.htm
4	duncan-cragg.org/blog/post/strest-service-trampled…	java.sun.com/developer/technicalArticles/…	java.sun.com/developer/technicalArticles/…
5	www.prescod.net/rest/mistakes/	www.infoq.com/articles/rest-introduction	www.infoq.com/articles/rest-introduction
6	www.prescod.net/rest/rest_vs_soap_overview/	enunciate.codehaus.org	enunciate.codehaus.org
7	particletree.com/features/how-to-add-an-api-….	en.wikipedia.org/wiki/Representational_...	en.wikipedia.org/wiki/Representational_...
8	hinchcliffe.org/archive/2005/08/18/1675.aspx	wadl.dev.java.net	wadl.dev.java.net
9	www.infoq.com/articles/tilkov-rest-doubts	jersey.dev.java.net	http://www.infoq.com/articles/sanjiva-rest-myths
10	hinchcliffe.org/archive/2008/02/27/16617.aspx	bitworking.org/news/201/RESTify-DayTrader	bitworking.org/news/201/RESTify-DayTrader

popular resources can quickly become obsolete. It is also worth noting a difference in how candidate experts for the topic generation process are selected versus the computation of EARL scores. For the topic generation process, only candidate experts who have annotated at least five of their bookmarks with a given series of tags that form a topic of interest as contributors to that

topic are considered. In other words, at the very least, a user must demonstrate consistent interest in a topic before we can even consider that person an expert. However, for the calculation of EARL's rankings, this minimum threshold is not used.

4.0 RESEARCH DESIGN

The goal of this research was to develop an algorithm that can identify experts and authoritative documents in social bookmarking systems more efficiently and more accurately than existing algorithms. We expect enhanced efficiency will be achieved by reducing the nodes in the Delicious data graph to a smaller subset of active users who consistently use several tags on their bookmarks. The additional factors in EARL used to model expertise are expected to lead to more accurate rankings of expert users and authoritative documents for a given topic.

The main questions we address in this research are:

- Does the EARL algorithm identify the experts and authoritative documents on a given topic in Delicious more accurately and more efficiently than existing algorithms?

- Does node reduction of the Delicious data graph to a smaller, sub-network of candidate experts produce expert and authoritative document rankings on a given topic that are on par with, or better, than those produced from the entire Delicious network?

4.1 DELICIOUS DATA

This research used data collected from the social bookmarking system, Delicious. Delicious was founded in 2003 by Joshua Schachter, and acquired by Yahoo! in December, 2005. Yahoo! then

sold Delicious to AVOS Systems in April, 2011. The current number of bookmarks, users, and resources on Delicious is unknown. The last public disclosure of these statistics was made by Delicious in 2008, stating that the site had 5.3 million registered users with bookmarks on 180 million unique resources.

The data used in this research were initially crawled between November, 2009 and February, 2010. After completing the preliminary studies discussed in the previous chapter, we expanded the dataset with a subsequent crawl of Delicious between May, 2010 and August, 2010. Due to limitations in crawling, we are unable to collect all bookmarks for all users and resources. For instance, Delicious restricts the viewing (and thus, crawling) of resource bookmarks to the most recent 2,000 entries. Despite these limitations, our goal for crawling was to collect as many bookmarks as possible, and construct a sample dataset that was representative of Delicious in its entirety. Bookmarks were collected on a per-user basis and per-resource basis, with care taken to ensure that tags were stored in the same order and case as originally entered by their authors. We accepted all bookmarks regardless of tag semantics, language, resource popularity, or user history. The main dataset used in this study includes 73,223,114 bookmarks made by 723,342 users (identical to the preliminary studies' main dataset) on 41,469,488 unique resources.

Based on the expanded main dataset, an initial list of candidate experts was generated. This research identifies candidate experts as Delicious users who have bookmarked at least 10 resources and used, on average, at least four tags per bookmark – as opposed to five in the preliminary analysis. The four-tag cutoff was used for the following reasons:

1. The four-tag cutoff follows Li et al.'s (2008) conclusion that one to five tags best represent a *single* topic of a resource, as well as the observations of Bates (1986) and

(2000) that multiple Library of Congress subject headings are appropriate for resource classification.

2. Many resources present information about more than one topic. In turn, some topic terms contain more than one word. Users may use a compound tag to represent the multiple-word term, or use separate tags for each word on the term.
3. We expect good classifiers to assign tags that describe the content of the resource at more than one level. For example, the tag set of a bookmarked resource about Java Servlets would not only contain specific topical tags (i.e. "servlets", "java"), but also more general topical tags (e.g., "programming", "webdev".)
4. A four-tag cutoff ensures adequate topic coverage within the candidate expert dataset. Many users who barely met the cutoff in the preliminary analysis fell below the five-tag threshold after gathering more of their bookmark data.

By reducing the cutoff to four tags per bookmark, the candidate expert dataset includes 23,066 users, or 3.2% of all users in the expanded main Delicious dataset. Table 16 summarizes the user, resource, and bookmark statistics of the main and candidate expert datasets.

Table 16. Basic statistics for the main and candidate expert datasets

	Main Dataset	**Candidate Expert Dataset**	**Candidate Expert %**
User Count	723,342	23,066	3.2%
Resource Count	41,469,488	4,493,594	10.8%
Bookmark Count	73,216,330	8,794,186	12.0%

4.2 PRE-PROCESSING OF DATA

Prior to using the main and candidate expert datasets for experiments, the following steps were taken to prepare the data:

1. **Convert all tag instances to lowercase**. Delicious does not impose any restrictions on case when users enter tags. Case is not important for this study's purposes, so all alphanumeric characters were converted to lowercase.
2. **Remove bookmarks with bogus dates, as it appears to be corrupt data on Delicious**. While crawling on a per-user basis, our crawlers occasionally collected bookmarks dated prior to the start of Delicious, evidence of data corruption. These bookmarks appear under the user's bookmark list on Delicious, but not the corresponding resource's bookmark list. With 40 million unique resources in the database, we do not have the time and resources to collect the 'first bookmarked' date of all resources. Any bookmark with a creation data before February 24, 2002 - the date of Joshua Schacter's earliest bookmarks[2] and the first bookmarks posted to Delicious was removed.
3. **If a resource has multiple URLs, combine all bookmarks under one resource ID**. Many popular resources on Delicious may be accessed on the Web via multiple URLs, and thus, have multiple URLs within Delicious. Multiple URLs dramatically affects the rankings of HITS, SPEAR, and EARL, especially when the bookmarks of a few popular resources are involved. Consider the users who bookmarked the main Google page. Some Delicious users bookmarked the URL "google.com", while others bookmarked "www.google.com." In Delicious, the two URLs have distinct Delicious IDs – Delicious

[2] http://www.delicious.com/joshua?sort=userdate&order=asc

creates its URL identifiers by hashing the URL – yet both URLs ultimately point to the same resource, the main Google page. Delicious assumes that when you look up either URL for Google (http://delicious.com/url/), you'd like to see everyone's bookmarks for the resource, not just that URL. Thus, the bookmark lists on Delicious for "google.com" and "www.google.com" are identical. Considering that we crawled Delicious on a per-user basis AND a per-resource basis, we find one of three problems in our main dataset for bookmarks on resources with multiple URLs:

- At one extreme, if 1,000 users bookmarked Google, *and we collected the bookmarks on a per-user crawl,* our data has 1,000 bookmarks. Five hundred of the bookmarks use the "google.com" URL identifier, while the other 500 bookmarks use the "www.google.com" identifier.
- At the other extreme, if 1,000 users bookmarked Google, *and we crawled both URL identifiers on a per-resource crawl*, our data will have 2,000 bookmarks for Google – i.e., two entries for each user, one with the "google.com URL identifier and a second with the "www.google.com" identifier.
- In most cases, the third scenario is a mix of the two: some users have two bookmarks for Google, while most have only one bookmark with one of the two identifiers.

Unfortunately, Delicious does not provide a mechanism that lists all the URL identifiers for a particular resource. We combine multiple URLs with a semi-automatic procedure used during the preliminary work for this dissertation. First, we run EARL, SPEAR, and HITS on some topic and list all relevant URLs, their corresponding Delicious URL identifiers, and their bookmark counts. We sort the list of URLs alphabetically, and manually group "sibling" URLs that point to the same resource. In most cases, these siblings

only differ by the presence or absence of a leading "www" in the domain name, or trailing "index.*" or "home.*" page name in the full address. In other cases, sibling URLs may include a query-string with referrer information that is harder to detect automatically. Finally, for each URL group representing a common resource, we select the identifier of the most popular URL based on bookmark count, then update the identifiers of all the *resource's* bookmarks in our data to the most popular identifier.

4.3 METHODOLOGY

To evaluate the performance of EARL versus other ranking algorithms, this research uses relevance measurements made by expert judges on documents from Delicious and Google. Documents were presented in random order to the judges, who rated each document's relevancy on a graded scale. We use Normalized Discounted Cumulative Gain (NDCG) to measure the performance of the ranking algorithms against the experts' ratings. The expert judges' collective ratings are considered ideal.

NDCG is a metric developed by Jarvelin and Kekalainen (2002) to assess how well information retrieval (IR) systems rank documents in response to a given query compared to an ideal ranking based on graded relevance judgments:

$$NDCG_q = M_q \sum_{j=1}^{K} \frac{2^{r(j)} - 1}{\log(1 + j)}$$

where *r(j)* is an integer denoting a graded relevance judgment (e.g., 1 = "irrelevant", 2 = "somewhat relevant", and 3 = "highly relevant") for a document at position *j*; *K* is the length of

the result vector to evaluate (i.e., the top K documents); and M_q is a normalization constant such that a perfect ordering of documents for the given query q gets a value of 1. The underlying notion behind NDCG is that an IR system should present highly relevant documents at the beginning of a ranked result list, followed by marginally relevant documents, followed by irrelevant documents. Highly relevant documents should be presented in the top positions (Jarvelin and Kekalainen, 2002). When calculating NDCG, a document's contribution to the final score directly relates to its position in the ranked list – the higher its position in the list, the more it contributes to the final NDCG score. Thus, algorithms that place the most highly relevant documents in the top K ranking positions achieve the highest NDCG scores.

4.4 EXPERIMENT 1: EVALUATING EARL'S ABILITY TO LOCATE AUTHORITATIVE RESOURCES

In the first experiment, we evaluated a technique for filtering candidate experts from Delicious and three algorithms for ranking authoritative resources in Delicious. The goals of the first experiment are 1) to discover which algorithm does the best job ranking authoritative resources on a given topic in Delicious, and 2) to test how effectively the candidate expert filtering procedure identifies Delicious users who possess domain expertise.

4.4.1 Participants

Thirty participants were recruited from the University of Pittsburgh's School of Information Sciences and Department of Computer Science. The sample size was chosen according to power

analysis (Cohen, 1988) for a two-way Analysis of Variance (ANOVA.) The power analysis suggested a minimum sample size of twenty participants assuming a large effect size (f = .75) and a significance level of p = .05 with a confidence of 0.8. Because a pilot test showed that six tasks required too much time for subjects to complete comfortably in one session, each subject's workload was reduced to four tasks. We recruited thirty participants and assigned them four tasks, such that each of the six tasks was performed by twenty participants.

4.4.2 Variables and Expected Results

For the first experiment, the two independent variables are 1) the ranking algorithm (EARL, HITS, SPEAR, and Google) and 2) the selected dataset (the main dataset and the candidate expert dataset.) Table 17 summarizes the independent variables and seven conditions in the first experiment. The dependent variable is the mean of $nDCG_{10}$ for a given ranking algorithm and dataset selection; i.e., the performance of each method's resource rankings against the ideal rankings of authoritative documents.

Table 17. Independent variables and conditions in the first experiment. Each subject ranks results lists from all seven conditions.

	Ranking algorithm			
Dataset	EARL	HITS	SPEAR	Google
Main	1	3	5	7
Expert	2	4	6	

We expect EARL to outperform HITS, SPEAR, and Google in ranking authoritative resources. We also expect the use of the filtered expert dataset to produce authoritative resource rankings as good as, or better than, an unfiltered Delicious dataset.

4.4.3 Hypotheses of the 1st Experiment

$H_{1\text{-}0}$: There is no statistically-significant difference among the means of the $nDCG_{10}$ of Google and the HITS-based ranking algorithms. ($\mu_{GOOGLE} = \mu_{EARL} = \mu_{SPEAR} = \mu_{HITS}$)

$H_{1\text{-}1}$: There is a statistically-significant difference among the means of the $nDCG_{10}$ of Google and the HITS-based ranking algorithms. ($\mu_{GOOGLE} \neq \mu_{EARL} \neq \mu_{SPEAR} \neq \mu_{HITS}$)

$H_{2\text{-}0}$: There is no statistically-significant difference between the means of the $nDCG_{10}$ of the main and expert datasets. ($\mu_{MAIN} = \mu_{EXPERT}$)

$H_{2\text{-}1}$: There is a statistically-significant difference between the means of the $nDCG_{10}$ of the main and expert datasets. ($\mu_{MAIN} \neq \mu_{EXPERT}$)

4.4.4 Subjects, Evaluation, and Analysis Procedure

Thirty students from the School of Information Sciences and the Department of Computer Science were recruited as subjects for the experiment[3]. To be eligible for the experiment, a student must have completed one course in the Java programming language, or have developed an application using the language[4]. Each subject was given four questions related to Java programming (please see section 5.1.) Prior to the start of the experiment, each subject was given a brief training session to ensure that they met the minimum requirements, understood their tasks, and understood how to use the experimental system. Subjects then formulated queries to locate resources that helped them answer each given question. They were asked to rate

[3] The study was approved as 'exempt' by the Institutional Review Board of the University of Pittsburgh (PRO12010167).
[4] The courses that appeared in the recruitment announcement were INFSCI 0017 (Fundamentals of Object-Oriented Programming) and CS 0401 (Intermediate Programming using Java). Equivalent courses at other schools were also accepted.

the relevancy of the retrieved resources to their given queries on a scale of 1 to 5, where "1" is "completely irrelevant" and "5" is "highly relevant." Figure 11 shows the interface subjects used to rate the relevancy of retrieved resources.

To retrieve a list of resources, the experimental system submits the subject's query to both Google and our own social annotation-based retrieval system. The experimental system only selects resources that *exactly match* the subject's query; i.e., *all* terms must have been used as tags on the resources from the Delicious datasets, or appear in the documents retrieved from Google. The experimental system receives the top 20 search results from Google, as well as *separate* top 20 lists from the social annotation-based retrieval system using each of the three HITS-based algorithms – EARL, SPEAR, and HITS – on both the main and candidate expert Delicious datasets. The experimental system combines the results from the seven conditions, removes duplicate results, and presents the combined result list to the subject in randomized

Figure 11. Experiment 1's user interface.

order. Subjects were reminded before submitting each query that their results sets appear in random order. For a given query, a subject rates a maximum of 140 results (i.e., the results sets of all seven conditions are completely distinct), and a minimum of twenty (i.e., the results sets overlap perfectly.) We expected significant overlap in the results sets from all four search algorithms, but did not expect the result sets to overlap perfectly.

The system recorded all relevancy ratings for each resource appearing in the subjects' result set lists. Using the subjects' ratings and the rank positions of resources for a given query and experimental condition (i.e. dataset and algorithm combination), we calculate the value of $nDCG_{10}$ for each dataset/ranking method based on each query.

Two-way between-subjects Analysis of Variance (ANOVA) is applied to test the hypotheses. The null hypothesis is rejected if the results from the F-test show a significant difference at the 0.05 confidence level. If one of the null hypotheses is rejected, all pairwise differences are examined with the Scheffe procedure.

4.5 EXPERIMENT 2: EVALUATING EARL'S ABILITY TO LOCATE DOMAIN EXPERTS

In the second experiment, we evaluate a technique for filtering candidate experts from Delicious, as well as three algorithms for ranking candidate experts with domain expertise. The goals of the second experiment are 1) to discover which algorithm does the best job ranking domain experts on a given topic in Delicious, and 2) to test how effectively the candidate expert filtering procedure identifies Delicious users who possess domain expertise.

4.5.1 Participants

The thirty participants recruited from the University of Pittsburgh's School of Information Sciences for Experiment 1 also participated in Experiment 2. Because all of the resources in all of the rank lists produced in Experiment 1 were present in at least one of the top candidate expert's bookmark lists in Experiment 2, it was feasible to utilize the participants' ratings for both experiments. Similar to experiment 2, the sample size was chosen according to power analysis (Cohen, 1988) for a two-way Analysis of Variance (ANOVA.) The power analysis suggested a minimum sample size of twenty participants assuming a large effect size (f = .75) and a significance level of p = .05 with a confidence of 0.8. Because a pilot test showed that six tasks required too much time for subjects to complete comfortably in one session, each subject's workload was reduced to four tasks. Thus, we recruited thirty participants and assigned them four tasks, such that each of the six tasks was performed by twenty participants.

4.5.2 Variables and Expected Results

For the second experiment, the two independent variables are 1) the ranking algorithm and 2) the selected dataset. Table 18 summarizes the independent variables and the six conditions in the

Table 18. Independent variables and conditions in the second experiment. Each subject ranks domain expert data from all six conditions.

	Ranking Algorithm		
Dataset	EARL	HITS	SPEAR
Main	1	3	5
Expert	2	4	6

second experiment. The dependent variable is the mean of $nDCG_{10}$ for a given ranking algorithm and dataset selection; i.e., the performance of each method's resource rankings against the ideal ranking of domain experts.

We expect EARL to outperform HITS and SPEAR in locating expert users and ranking domain expertise. We also expect the use of the filtered expert dataset to produce domain expert rankings as good as, or better than, an unfiltered Delicious dataset.

4.5.3 Hypotheses of the 2nd Experiment

$H_{1\text{-}0}$: There is no statistically-significant difference among the means of the $nDCG_{10}$ of the candidate expert rankings for EARL, SPEAR, and HITS. ($\mu_{EARL} = \mu_{SPEAR} = \mu_{HITS}$)

$H_{1\text{-}1}$: There is a statistically-significant difference among the means of the $nDCG_{10}$ of the candidate expert rankings for EARL, SPEAR, and HITS. ($\mu_{EARL} \neq \mu_{SPEAR} \neq \mu_{HITS}$)

$H_{2\text{-}0}$: There is no statistically-significant difference between the means of the $nDCG_{10}$ of the candidate expert rankings for the candidate expert and main datasets. ($\mu_{MAIN} = \mu_{EXPERT}$)

$H_{2\text{-}1}$: There is a statistically-significant difference between the means of the $nDCG_{10}$ of the candidate expert rankings for the candidate expert and main datasets. ($\mu_{MAIN} \neq \mu_{EXPERT}$)

4.5.4 Subjects, Evaluation, and Analysis Procedure

Thirty students from the School of Information Sciences and Department of Computer Science were recruited as subjects for the experiment[5]. As stated in section 4.5.1, the same thirty subjects who participated in Experiment 1 also participated in Experiment 2. To be eligible for the experiment, a student must have completed one course in the Java programming language, or have developed an application using the language[6]. Each subject was given four questions related to Java programming (please see section 5.1.) Prior to the start of the experiment, each subject was provided with a brief training session to ensure that they met the minimum requirements, understood their tasks, and understood how to use the experimental system.

Providing subjects with lists of candidate experts (i.e. usernames) to rate directly will not provide reliable ratings of domain expertise. To assess domain expertise, subjects were asked to rate the *resources* bookmarked by the highest-ranked candidate experts by each algorithm. We expect that the top experts in Delicious on a given topic have bookmarked the top authoritative resources. Similar to the first experiment, subjects formulated topic queries to locate *resources* that provide relevant information on each topic. They were asked to rate the relevancy of the retrieved resources to their given queries on a scale of 1 to 5, where "1" is "completely irrelevant" and "5" is "highly relevant." Subjects used the same interface (Figure 11) as in Experiment 1 to rate the relevancy of resources.

---

[5] The study was approved as 'exempt' by the Institutional Review Board of the University of Pittsburgh (PRO12010167).

[6] The courses that appeared in the recruitment announcement were INFSCI 0017 (Fundamentals of Object-Oriented Programming) and CS 0401 (Intermediate Programming using Java). Equivalent courses at other schools were also accepted.

To generate a list of results for subjects to rate, the experimental system submits the subject's topic query to the social annotation-based retrieval system. As in Experiment 1, the experimental system only selects resources that *exactly match* the subject's query; i.e., *all* terms must have been used as tags on the resources from the Delicious datasets. In return, the experimental system receives separate top 15 lists of candidate experts for each of the three ranking algorithms on both the main and candidate expert Delicious datasets. For each retrieved candidate expert in the six conditions, the experimental system extracts the expert's top resources by authority score, up to a maximum of ten. The experimental system then combines the resource lists, removes any duplicate results, and presents the filtered result list to the subject in randomized order. Subjects were reminded before submitting each query that their search results would appear in random order. For a given topic query, a subject may rate a maximum of 900 results (i.e., the top 10 resources of each expert, as well as the list of experts from all six conditions, are completely distinct), and a minimum of fifteen (i.e., the same top 15 experts in all six conditions, as well as each expert's top 10 resources.) We expected significant overlap in the results sets from all six conditions, but did not expect the result sets to overlap perfectly.

The system records all relevancy ratings for each resource appearing in the subjects' result set lists. Using the subjects' ratings of resources for a given query, we calculated two composite scores for each expert. The first composite score is the mean rating of a candidate expert's top resources by authority score matching that query, up to a maximum of ten resources. The second composite score is the percentage of high-quality resources (i.e., rated "4" or above by subjects) bookmarked by the candidate expert, assuming a minimum of five high-quality resources bookmarked. The choice of five resources follows Bharat and Mahaila's (2000) criteria used to select "expert" documents for inclusion in Hilltop's index. After generating the two

composite scores for each retrieved candidate expert, we calculated separate $nDCG_{10}$ values for the candidate expert rankings for each query and experimental condition (i.e. dataset and algorithm combination.)

Two-way between-subjects Analysis of Variance (ANOVA) is applied to test the hypotheses. The null hypothesis is rejected if the results from the F-test show a significant difference at the confidence level of $\alpha = 0.05$. If one of the null hypotheses is rejected, all pairwise differences are examined with the Scheffe procedure.

4.6 EXPERIMENT 3: EVALUATING TOPICS OF INTEREST TO LOCATE CLASSIFICATION EXPERTS

The third experiment evaluates a technique that filters candidate *classification* experts from Delicious, generates power sets of the candidate experts' tag sets, and selects frequently co-occurring terms shared by many candidate experts to classify resources. The goals of the third experiment are 1) to test the effectiveness of aggregating shared power sets among many users for finding good classification terms, and 2) to test how well the candidate expert filtering procedure identifies Delicious users with classification expertise.

4.6.1 Experimental Data

The third experiment utilized data from three sources: the candidate expert and main Delicious datasets, and category labels collected from the Open Directory Project (ODP), a hierarchical directory of web resources maintained by volunteer editors. The Delicious datasets represent

classification terms created by novice users, and the ODP categories represent those generated by human-expert classifiers using a controlled vocabulary.

Twenty-five web resources (Table 19) were randomly selected from ODP and the Delicious datasets that meet the following criteria:

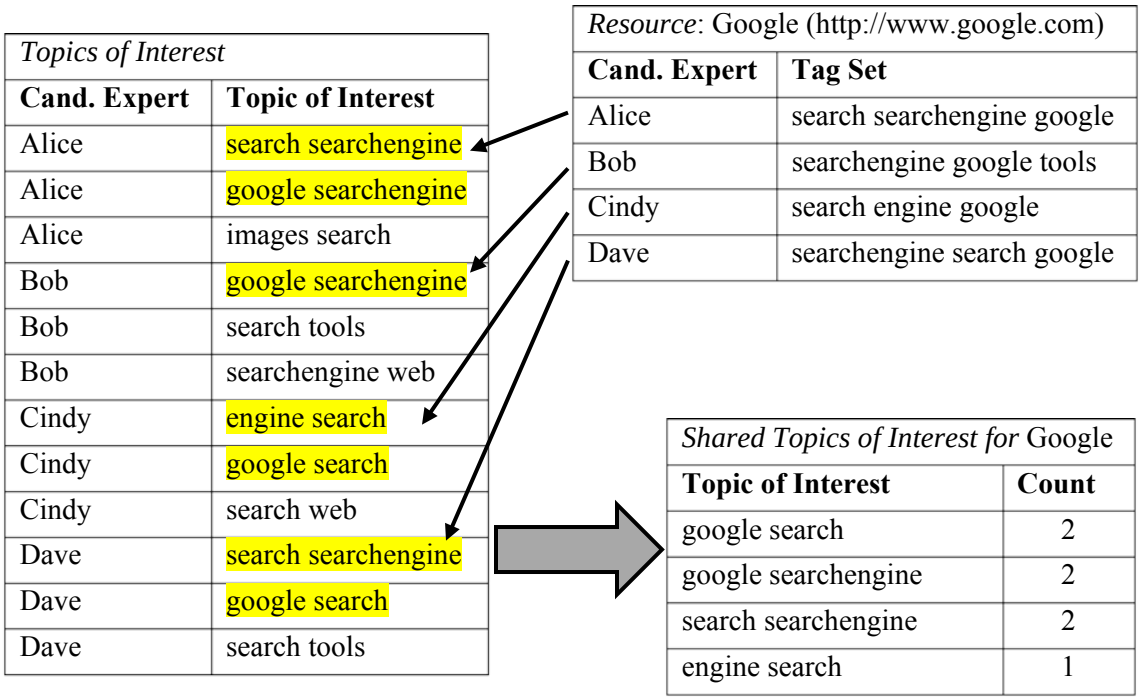
1. The resource must be found in all three data sources.
2. The resource must have been bookmarked by at least 100 users in each of the Delicious datasets to ensure that the resources' tagging patterns have stabilized.
3. The resource is currently available on the Web.

Table 19. List of resources selected for Experiment 3

	Title	URL
1	Gazelle	http://www.gazelle.com/
2	MIT OpenCourseWare	http://ocw.mit.edu/OcwWeb/index.htm
3	Android Developers	http://developer.android.com/
4	Geni	http://www.geni.com
5	Lynda.com	http://www.lynda.com/
6	Clearleft	http://www.clearleft.com/
7	Monster	http://www.monster.com/
8	MOO	http://www.moo.com/
9	WordReference.com	http://www.wordreference.com/
10	HubbleSite	http://hubblesite.org/
11	Twitter	http://twitter.com
12	EasyBib	http://www.easybib.com/
13	timeanddate.com	http://www.timeanddate.com/
14	Paint.NET	http://www.getpaint.net/
15	Python	http://www.python.org/
16	Toggl	http://www.toggl.com/
17	Yahoo! Finance	http://finance.yahoo.com/
18	Alexa	http://www.alexa.com/
19	Hulu	http://www.hulu.com/
20	Wired.com	http://www.wired.com/
21	Wikispaces	http://www.wikispaces.com/
22	PayPal	http://paypal.com/
23	Wolfram MathWorld	http://mathworld.wolfram.com/
24	ipl2	http://www.ipl.org/
25	Free Music Archive	http://freemusicarchive.org/

Satisfying these criteria limited the selected resources to high-level web pages (i.e., home pages of web sites), as opposed to low-level, specific web pages (e.g., news articles), because ODP tends to include only high-level web pages in its collection.

Classification terms were collected for each resource from its ODP category labels, the ten most frequently-used tags in the main Delicious dataset, and tags gathered from the twenty most frequently-shared topics of interest related to the resource among candidate experts. Figure 12 illustrates how classification terms were extracted from candidate experts' topics of interest. For each resource, we selected all of its bookmarks in the candidate expert dataset. For each bookmark, we identified the candidate expert who made the bookmark, and selected all of their topics of interest previously collected using the technique described in section 3.4.2. Finally, we compared the bookmark's set of tags to each topic of interest. If the bookmark's tag set contains *all* of the terms in the topic of interest, we select that topic of interest as a potentially good source of classification terms for the given resource. As we iterated over bookmarks for the given resource and selected matching topics of interest, we kept a running tally of the number of candidate experts who share a particular topic of interest on the given resource. We repeated this process on the candidate expert dataset for all twenty-five resources, storing the twenty most frequently-shared topics of interest of each resource.



Topics of Interest

Cand. Expert	Topic of Interest
Alice	search searchengine
Alice	google searchengine
Alice	images search
Bob	google searchengine
Bob	search tools
Bob	searchengine web
Cindy	engine search
Cindy	google search
Cindy	search web
Dave	search searchengine
Dave	google search
Dave	search tools

Resource: Google (http://www.google.com)

Cand. Expert	Tag Set
Alice	search searchengine google
Bob	searchengine google tools
Cindy	search engine google
Dave	searchengine search google

Shared Topics of Interest for Google

Topic of Interest	Count
google search	2
google searchengine	2
search searchengine	2
engine search	1

Figure 12. Example of extracting relevant, shared topics of interest from candidate experts' bookmarks of the Google homepage.

Additionally, we generated potential topics of interest using the same power set generation process described in section 3.3.3, but tabulated the subset frequencies on a *per-resource* basis. For each resource listed in Table 19, we generated and tabulated the top 20 subsets of frequently co-occurring tags by count based on the resource's bookmarks in the candidate expert dataset. Table 20 shows the top 20 subsets of frequently co-occurring tags from candidate experts' bookmarks on the Gazelle home page (http://www.gazelle.com.)

Table 20. The top 20 subsets of frequently co-occurring tags from candidate experts' bookmarks on http://www.gazelle.com, as identified by the topic of interest process described in Section 3.3.3.

Rank	Tag Subset	Count
1	electronics recycle	69
2	electronics gadgets	67
3	electronics shopping	63
4	electronics recycling	59
5	gadgets shopping	57
	gadgets recycle	57
7	recycle shopping	56
8	electronics gadgets recycle	55
9	electronics used	54
	electronics gadgets shopping	54
	sell shopping	54
12	electronics sell	53
	recycle sell	53
14	shopping used	52
	electronics recycle shopping	52
16	gadgets recycling	51
	electronics gadgets recycling	51
18	selling shopping	48
	electronics recycle sell	48
	recycling used	48

4.6.2 Participants

Twenty participants were recruited from the University of Pittsburgh's School of Information Sciences, Pittsburgh libraries, and other libraries in Pennsylvania. The sample size was chosen according to power analysis (Cohen, 1988) for a one-way Analysis of Variance (ANOVA.) The power analysis suggested a minimum sample size of sixteen participants assuming a large effect size (f = .75) and a significance level of p = .05 with a confidence of 0.8. Thus, we recruited twenty participants to perform Experiment 3.

Cataloging knowledge and skill – whether through coursework or professional experience – was a critical factor in recruiting subjects for this experiment. Participants were expected to analyze a series of resources and rate the relevancy of potential classificatory terms as information organization professionals. Therefore, we focused our recruitment efforts on persons who would most likely have classification expertise: professional librarians and graduate students in the Library and Information Science program who have completed courses in information organization[7].

4.6.3 Variables and Expected Results

The independent variable is the source of classification terms (ODP category labels, Top 10 Delicious tags, or Candidate Expert power sets.) The dependent variable is the mean of NDCG of a given source of classification terms; i.e., the rankings of the classification terms selected from a data source versus the ideal rankings of classification terms generated by subjects.

We expected the classification terms selected by the power sets of candidate experts' tag sets for a given resource to be as good as, or better, than the resource's Top 10 tags from the main dataset, the ODP category terms, and the terms selected from the power sets of the resource's tag sets. We also expected to find that the candidate experts are more likely to tag resources with good classification terms than the average Delicious user.

[7] The courses that appeared in the recruitment announcement were LIS2005 (Organizing & Retrieving Information) and LIS2405 (Introduction to Cataloging).

4.6.4 Hypotheses of the 3rd Experiment

$H_{1\text{-}0}$: There is no statistically-significant difference among the means of the NDCG of classification terms selected by candidate experts' power sets, top 10 tags from the main dataset, the ODP category terms, and the most-frequently co-occurring subsets of tags among a resource's tag sets. ($\mu_{POWERSETS\_EXPERT} = \mu_{TOP10} = \mu_{ODP} = \mu_{SUBSETS\_RESOURCE}$)

$H_{1\text{-}1}$: There is a statistically-significant difference among the means of the NDCG of classification terms selected by candidate experts' power sets, top 10 tags by popularity from the main dataset, the ODP category terms, and the most-frequently co-occurring subsets of tags among a resource's tag sets. ($\mu_{POWERSETS} \neq \mu_{TOP10} \neq \mu_{ODP} \neq \mu_{SUBSETS\_RESOURCE}$)

$H_{2\text{-}0}$: There is no statistically-significant difference between the mean percentages of candidate experts using high-quality tags (i.e., tags rated as "good" or "excellent" classification terms) on resources versus all users in the main dataset. ($\mu_{EXPERT\_RATINGS} = \mu_{MAIN\_RATINGS}$)

$H_{2\text{-}1}$: There is a statistically-significant difference between the mean percentages of candidate experts using high-quality tags on resources versus all users in the main dataset. ($\mu_{EXPERT\_RATINGS} \neq \mu_{MAIN\_RATINGS}$)

The null hypotheses are rejected if the results from the corresponding F-test indicate a significant difference at the 0.05 level. If a null hypothesis is rejected, all pairwise differences are examined to find which dataset yielded the most relevant terms for classification.

4.6.5 Subjects, Evaluation, and Analysis Procedure

Twenty participants were recruited as subjects for the third experiment, including students from the University of Pittsburgh's Library and Information Science program, professional librarians

at the University, catalogers at the Carnegie Library System, and professional librarians from Berks County, Pennsylvania[8]. To be eligible for the experiment, a participant must have either 1) completed one course in classification, or 2) have professional cataloging experience. Prior to the start of the experiment, each subject was provided with a brief training session to ensure that they understood their tasks, and understood how to use the experimental system.

Each subject was presented with all twenty-five resources (Table 19) and a list of terms corresponding to each resource. For each resource, the system selects and presents terms from the matching candidate expert/resource power sets, the top seven tags by popularity in the main dataset, the ODP category terms, and the terms from the most frequently co-occurring subsets of tags among the resource's tag sets. Any duplicate terms among the four sources were removed, so that subjects do not rate the same term more than once. If a term is a compound tag, the

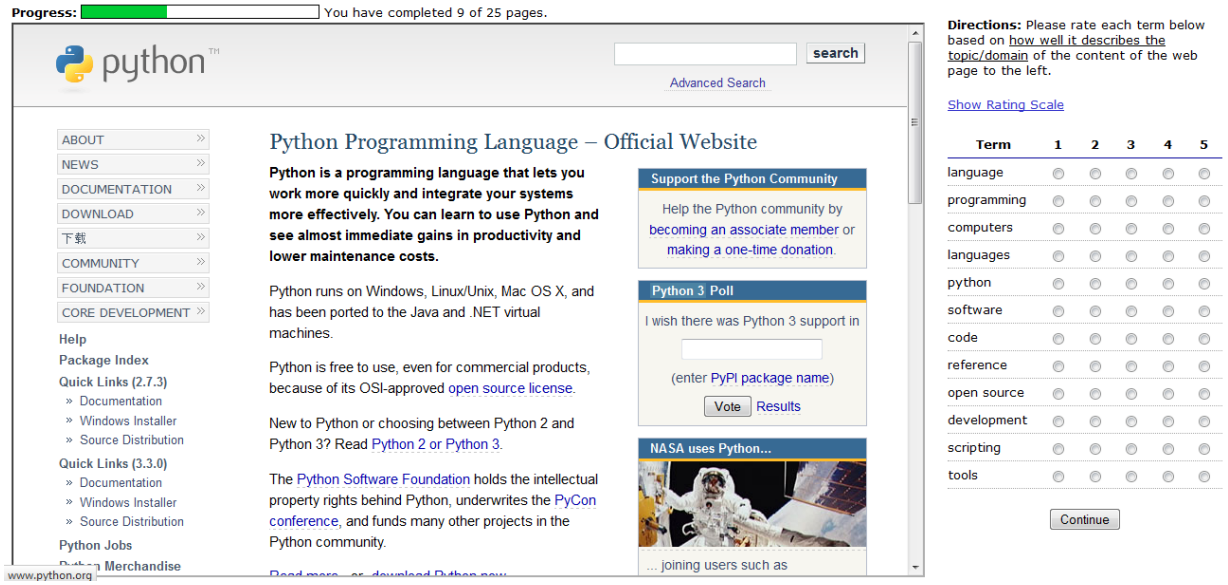


Figure 13. Experiment 3's user interface.

[8] The study was approved as 'exempt' by the Institutional Review Board of the University of Pittsburgh (PRO12010167).

system splits the compound tag and present the resulting terms separated by a space (e.g., "webdesign" is presented as "web design".) Subjects were asked to rate each term's relevancy to the given resource on a five-point scale, where "1" means the term is a very poor classificatory term for the resource; "3" is an acceptable term; and "5" is an excellent term. Because the subjects recruited for this experiment are classification experts, their ratings are considered ideal. Figure 13 shows the interface participants used to rate the relevancy of terms to a given resource.

Two analyses were performed using the subjects' ratings. In the first analysis, the relevance ratings of tags from the Delicious datasets and ODP's expert-generated category terms were evaluated using a one-way Analysis of Variance (ANOVA) test. Prior to running the test, we calculated a composite rating for each candidate expert power set whose component terms were presented to subjects. A power set's composite rating was computed as the mean rating of the set's component terms. For the second analysis, the percentages of candidate experts and average Delicious users using high-quality tags were evaluated using a one-way Analysis of Variance (ANOVA) test.

5.0 RESULTS

This chapter presents the results of the three experiments. The first section of the chapter describes the selection process for the Java programming-related questions presented to subjects when collecting ratings data for Experiments 1 and 2. The second section presents an analysis of the consistency of subjects' ratings collected for Experiments 1 and 2, as well as the participants' ratings in Experiment 3. The third, fourth, and fifth sections review the results of Experiments 1 to 3, respectively. The final section of the chapter provides a discussion of the results.

5.1 QUESTIONS USED IN EXPERIMENTS 1 & 2

As discussed in Sections 4.4.1 and 4.5.1, thirty participants were recruited from the University of Pittsburgh to provide ratings data for Experiments 1 and 2. Because all resources in all of the authoritative resource rank lists were present in at least one of the top candidate expert's bookmarked resource lists, subjects produced the ratings data for both experiments in a single session. Each subject was given four tasks to complete, all related to Java programming. This research relies heavily on subjects' relevancy ratings to evaluate the performance of the proposed candidate expert filtering and ranking algorithms. Thus, it was important to identify question topics familiar to the subject population, either through coursework in Java programming or practical experience building a Java application.

Table 21. List of questions used in Experiments 1 and 2.

Question	Question Text/Task Description
A	There are many different sorting algorithms, such as Bubble Sort, Merge Sort, and Heapsort. Find web pages that explain sorting algorithms.
B	Programmers use Integrated Development Environments (IDEs) to help them develop applications. Find web pages that provide information on an IDE for Java.
C	An error or exception can disrupt the normal flow of a program. Find web pages that explain exceptions in Java.
D	Students and professionals often expand their knowledge of a programming language by studying working examples of code. Find web pages that provide examples of Java code.
E	The Java Collections Framework provides a set of ready-to-use data structures, such as Lists, Queues, and Maps. Find web pages that discuss Collections in Java.
F	"Swing" is the name of Java's main toolkit of components for building graphical user interfaces (GUIs). Find web pages that present a tutorial related to Java Swing.

To select appropriate topics for the experiments' questions, syllabi from two University of Pittsburgh undergraduate courses were reviewed: INFSCI 0017 (Fundamentals of Object-Oriented Programming) and CS 0401 (Intermediate Programming in Java.) Table 21 shows the six questions written and used for Experiments 1 and 2 based on the material covered in the two Java programming courses. Questions A through D are similar to those used by Choochaiwattana (2008.) All questions are exploratory in nature, asking for broader information about a topic rather than specific answers to narrowly-defined problems. Choosing exploratory questions for the experiments allows us to better analyze both the breadth and depth of candidate experts' knowledge in Java programming topics.

As mentioned in Section 4.5.1, each of the thirty subjects who rated web resources for Experiments 1 and 2 completed search tasks for four of the six questions. Because a pilot test

showed that six tasks could not be completed comfortably by a subject within a single session, the workload for subjects was reduced to four tasks. Using a Latin square, the choice and sequence of questions for each subject session were assigned prior to the experiments. Questions were assigned to subjects such that each task would be completed by twenty subjects.

5.2 ASSESSMENT OF THE SUBJECTS' RELEVANCY RATINGS

To assess the inter-rater reliability of subjects' ratings, Fleiss' kappa (Fleiss, 1971) was calculated separately on the ratings of Java programming resources collected in Experiments 1 and 2, as well as those of the classificatory terms produced in Experiment 3. Fleiss' kappa is a statistical measure of inter-rater reliability among multiple raters who assigned ratings to items based on a fixed-number of categories (e.g., a five-point Likert scale.) Equation 5 defines Fleiss' kappa as:

$$\kappa = \frac{\bar{P} - \bar{P_e}}{1 - \bar{P_e}} \tag{5}$$

where the denominator $1 - \bar{P_e}$ represents the level of inter-rater agreement that can possibly be obtained above random, and the numerator $\bar{P} - \bar{P_e}$ is the actual, observed level of agreement among raters achieved beyond random. A κ value of 1 indicates perfect agreement among the raters, while a value of 0 indicates no agreement among raters beyond what could be expected from chance.

5.2.1 Inter-rater Reliability for Experiments 1 & 2

As explained in the previous chapter, the subjects recruited to provide the ratings used in Experiments 1 and 2 were given a series of questions related to Java programming, then asked to formulate their own topic queries to locate resources that provide useful information for answering those questions. Subjects were permitted to generate their own queries for each question in order to imitate a real-world, information-seeking scenario. Although we expected and observed some overlap in their queries for a particular question, subjects typically issued diverse queries to the experimental system. As a result, the experimental system returned different sets of search results for a given question to each subject, meaning not all resources were rated by the same number of users. This poses a problem when calculating Fleiss' kappa, because the calculation assumes that all items have been rated by an equal number of raters.

To assess the consistency of subjects ratings on the resources presented in Experiments 1 and 2 and despite the limitation of unequal of numbers of raters, we proceed in calculating using the ratings of those resources judged by at least 50% of the subjects for a given question. The maximum number of subjects that could potentially rate a resource is twenty; therefore, we select all resources that were judged by at least ten subjects. Of the 1,576 resources presented to subjects across the six questions, 525 resources (33.3%) were rated by at least ten subjects. Using the ratings on these 525 resources (shown in Appendix A), Fleiss' kappa is calculated as follows:

$$\kappa_{1\&2} = \frac{\bar{P} - \bar{P_e}}{1 - \bar{P_e}} = \frac{.4 - .21}{1 - .21} = 0.239 \qquad (6)$$

Based on a $\kappa_{1\&2}$ value of 0.239, we conclude that there was fair agreement[9] among the subjects. However, we acknowledge that the unequal number of raters per resource limits our ability to draw conclusions about the true reliability of the subjects' ratings based on the statistic as computed.

5.2.2 Inter-rater Reliability for Experiment 3

The twenty participants in Experiment 3 were asked to judge the relevancy of a fixed set of classification terms on a series of twenty-five web resources using a five-point scale. Unlike the design of Experiments 1 and 2, there was no variability in the information presented to Experiment 3's participants; i.e., all twenty subjects rated identical sets of terms on the same twenty-five web resources. Thus, the Fleiss' kappa statistic can be calculated using all ratings provided by subjects on all classification terms.

During an experimental session, each subject rated 425 classification terms over the twenty-five web resources presented to them. Using the ratings assigned by subjects on the 425 items, Fleiss' kappa is computed as follows:

$$\kappa_3 = \frac{\bar{P} - \bar{P_e}}{1 - \bar{P_e}} = \frac{.32 - .23}{1 - .23} = 0.125 \qquad (7)$$

Based on a κ_3 value of 0.125, we conclude that there was only slight agreement among the subjects. The level of agreement is lower than expected, considering the cataloging experience of the subjects and the consistency with which the experiment task was explained to subjects. On the other hand, subjects were asked to rate the relevancy of terms as keywords to a particular

[9] Fleiss and Koch (1977) provide a table to interpret the resulting κ value. According to Fleiss and Koch, a κ value between 0.21 and 0.40 represents "fair" agreement.

resource based on their own judgments without the benefit of a controlled vocabulary. Subjects tended to agree much more with each other when rating very specific classificatory terms, or terms that appeared prominently on the resource. However, most of the terms shown to subjects represented broader categories or descriptive terms that did not appear prominently on resources, leaving the relevancy of the terms more open to interpretation.

5.3 EXPERIMENT 1: RANKING OF AUTHORITATIVE DOCUMENTS

The goals of the first experiment are 1) to discover which ranking algorithm – HITS, SPEAR, or EARL – does the best job ranking authoritative resources on a given topic in Delicious, and 2) to test how effectively the candidate expert filtering procedure identifies domain experts in Delicious who are good sources of bookmarks on authoritative resources. The results of the three ranking algorithms are compared to those from Google, which is considered the top commercial Web retrieval system for locating authoritative documents. The details of Google's current algorithm are not publicly available, nor is it known how much the PageRank algorithm influences Google's search results. We also compare the ranked lists presented by the three ranking algorithms incorporated in our social annotation-based retrieval system when using the filtered candidate expert dataset versus the unfiltered main dataset.

For each of the four questions randomly assigned to them (Table 21), subjects were asked to formulate topic queries and rate the relevancy of each returned result to the given question. Please note that the subjects' ratings of Java programming resources were used for both Experiments 1 and 2.

5.3.1 Analysis of Entry Questionnaire Responses

Prior to the experiment, subjects were asked to fill out an entry questionnaire (Appendix B) similar to the one used by Choochaiwattana (2008), but with age, gender, and age of schooling questions removed. Table 22 summarizes the questionnaire responses. Of the thirty subjects recruited to provide ratings of Java programming resources for Experiments 1 and 2, 73% were students in either the undergraduate (BSIS) or graduate (MSIS) Information Sciences programs at the University of Pittsburgh; 13% were Computer Science students; 10% were students in the Telecommunications program; and one was a student in the Computer Engineering program. Seventy-seven percent of the subjects self-reported their knowledge of Java as "Intermediate"; 16.7% reported their knowledge level as "Novice"; and 6.7% reported their knowledge as "Expert". The subjects who reported their knowledge level of Java as "Novice" were monitored throughout the experiment to be sure they understood the question topics.

Fifty-three percent of the subjects had used Java for 1 to 3 years; 20% for more than 4 years; and 26.7% for less than one year. The most commonly-reported programming languages learned other than Java were C (63%) and C++ (43%.) Most of the subjects (63%) reported issuing fifteen or more queries to a search engine per day. Finally, subjects were also asked to self-rate their success rate in finding relevant information though a search engine. The majority of subjects (77%) said they are successful most of the time.

5.3.2 Analysis of Authoritative Document Rankings by Algorithm & Dataset

The following hypotheses were tested in Experiment 1:

$H_{1\text{-}0}$: There is no statistically-significant difference among the means of the $nDCG_{10}$ of Google and the HITS-based ranking algorithms. ($\mu_{GOOGLE} = \mu_{EARL} = \mu_{SPEAR} = \mu_{HITS}$)

$H_{1\text{-}1}$: There is a statistically-significant difference among the means of the $nDCG_{10}$ of Google and the HITS-based ranking algorithms. ($\mu_{GOOGLE} \neq \mu_{EARL} \neq \mu_{SPEAR} \neq \mu_{HITS}$)

$H_{2\text{-}0}$: There is no statistically-significant difference between the means of the $nDCG_{10}$ of the main and expert datasets. ($\mu_{MAIN} = \mu_{EXPERT}$)

$H_{2\text{-}1}$: There is a statistically-significant difference between the means of the $nDCG_{10}$ of the main and expert datasets. ($\mu_{MAIN} \neq \mu_{EXPERT}$)

Two-way between-subjects Analysis of Variance (ANOVA) was used to test the two sets of hypothesis. Figures 14 and 15 show the results of Experiment 1. Please note that one observation is missing from each condition (n=119) due to the lack of data for one task by one subject. We reject both null hypotheses, $H_{1\text{-}0}$ and $H_{2\text{-}0}$, as there is evidence that the means of the $nDCG_{10}$ of Google and the HITS-based ranking algorithms are significantly different at the $\alpha = .05$ level, as well as the means of the $nDCG_{10}$ of the main and expert datasets, $F(6, 832) = 41.241$, $p < .001$, and $\eta^2 = .230$.

Pairwise comparisons using the Scheffe procedure were then performed to determine the pattern of differences among the ranking algorithms. Because there are only two datasets, pairwise comparisons using the Bonferroni adjustment were used in lieu of post-hoc comparisons with the Scheffe procedure to find the pattern of differences between the main and expert dataset. Figure 16 shows the results of the comparisons. The $nDCG_{10}$ of Google was significantly higher than all three of the HITS-based ranking algorithms, but there were no significant differences in resource ranking performance among the three HITS-based algorithms. The comparisons of the datasets suggest that the candidate expert filtering procedure (*EXPERT*) performed significantly worse in ranking resources compared to no filtering (*MAIN*) for the HITS algorithm only. There were no significant differences in resource rankings for SPEAR or EARL when the candidate expert filtering procedure was applied compared to when the filtering procedure was not applied.

Tests of Between-Subjects Effects

Dependent Variable: nDCG

Source	Type III Sum of Squares	df	Mean Square	F	Sig.	Partial Eta Squared
Corrected Model	4.154[a]	6	.692	41.121	.000	.230
Intercept	421.415	1	421.415	25029.787	.000	.968
algorithm	3.090	3	1.030	61.171	.000	.182
dataset	.097	1	.097	5.775	.016	.007
algorithm * dataset	.083	2	.042	2.471	.085	.006
Error	13.907	826	.017			
Total	448.105	833				
Corrected Total	18.061	832				

a. R Squared = .230 (Adjusted R Squared = .224)

Figure 14. The results of two-way between-subjects ANOVA for Experiment 1

Descriptive Statistics

Dependent Variable: nDCG

algorithm	dataset	Mean	Std. Deviation	N
HITS	main	.7060205715	.1385082790	119
	expert	.6574545339	.1258828130	119
	Total	.6817375527	.1342898392	238
SPEAR	main	.6987225488	.1373082277	119
	expert	.6731006360	.1247076297	119
	Total	.6859115924	.1315104510	238
EARL	main	.7019239580	.1674062855	119
	expert	.7060952202	.1263965405	119
	Total	.7040095891	.1480270723	238
GOOGLE	main	.8862640782	.0664704230	119
	Total	.8862640782	.0664704230	119
Total	main	.7482327891	.1545315070	476
	expert	.6788834634	.1269448639	357
	Total	.7185116495	.1473362693	833

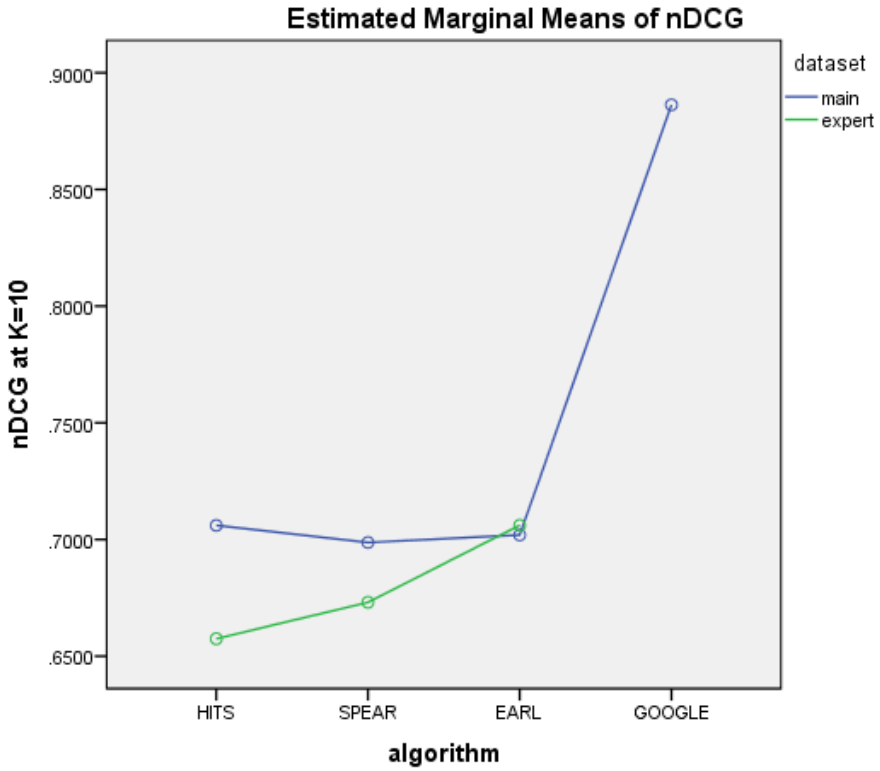


Figure 15. The means and standard deviations of nDCG10 for Experiment 1 (n=833.)

Multiple Comparisons

Dependent Variable: nDCG

Scheffe

(I) algorithm	(J) algorithm	Mean Difference (I-J)	Std. Error	Sig.	95% Confidence Interval	
					Lower Bound	Upper Bound
HITS	SPEAR	-.0041740397	.0118946877	.989	-.0374941271	.0291460478
	EARL	-.0222720363	.0118946877	.321	-.0555921238	.0110480511
	GOOGLE	-.204526525*	.0145679578	.000	-.2453351317	-.1637179192
SPEAR	HITS	.0041740397	.0118946877	.989	-.0291460478	.0374941271
	EARL	-.0180979967	.0118946877	.510	-.0514180842	.0152220908
	GOOGLE	-.200352486*	.0145679578	.000	-.2411610920	-.1595438795
EARL	HITS	.0222720363	.0118946877	.321	-.0110480511	.0555921238
	SPEAR	.0180979967	.0118946877	.510	-.0152220908	.0514180842
	GOOGLE	-.182254489*	.0145679578	.000	-.2230630953	-.1414458828
GOOGLE	HITS	.2045265254*	.0145679578	.000	.1637179192	.2453351317
	SPEAR	.2003524858*	.0145679578	.000	.1595438795	.2411610920
	EARL	.1822544891*	.0145679578	.000	.1414458828	.2230630953

Based on observed means.

The error term is Mean Square(Error) = .017.

*. The mean difference is significant at the .05 level.

Pairwise Comparisons

Dependent Variable: nDCG

algorithm	(I) dataset	(J) dataset	Mean Difference (I-J)	Std. Error	Sig.[b]	95% Confidence Interval for Difference[b]	
						Lower Bound	Upper Bound
HITS	main	expert	.049*	.018	.007	.014	.084
	expert	main	-.049*	.018	.007	-.084	-.014
SPEAR	main	expert	.026	.018	.151	-.009	.061
	expert	main	-.026	.018	.151	-.061	.009
EARL	main	expert	-.004	.018	.815	-.039	.031
	expert	main	.004	.018	.815	-.031	.039

Based on estimated marginal means

*. The mean difference is significant at the .050 level.

b. Adjustment for multiple comparisons: Bonferroni.

Figure 16. Comparisons to find significant differences in $nDCG_{10}$ among the ranking algorithms and use of filtering procedure, respectively.

5.4 EXPERIMENT 2: RANKING OF DOMAIN EXPERTS

The goals of the second experiment are 1) to discover which ranking algorithm – HITS, SPEAR, or EARL – does the best job ranking domain experts on a given topic in Delicious, and 2) to test how effectively the candidate expert filtering procedure identifies domain experts in Delicious who possess expertise. Please note that the same thirty subjects from Experiment 1 provided ratings data concurrently for Experiment 2. As in Experiment 1, subjects were asked to formulate topic queries to locate resource relevant to the four questions randomly assigned to them (Table 20), then rate the relevancy of each returned result to the given question.

The following hypotheses were tested in Experiment 2:

$H_{1\text{-}0}$: There is no statistically-significant difference among the means of the $nDCG_{10}$ of the candidate expert rankings for EARL, SPEAR, and HITS. ($\mu_{EARL} = \mu_{SPEAR} = \mu_{HITS}$)

$H_{1\text{-}1}$: There is a statistically-significant difference among the means of the $nDCG_{10}$ of the candidate expert rankings for EARL, SPEAR, and HITS. ($\mu_{EARL} \neq \mu_{SPEAR} \neq \mu_{HITS}$)

$H_{2\text{-}0}$: There is no statistically-significant difference between the means of the $nDCG_{10}$ of the candidate expert rankings for the candidate expert and main datasets. ($\mu_{MAIN} = \mu_{EXPERT}$)

$H_{2\text{-}1}$: There is a statistically-significant difference between the means of the $nDCG_{10}$ of the candidate expert rankings for the candidate expert and main datasets. ($\mu_{MAIN} \neq \mu_{EXPERT}$)

5.4.1 Analysis of Domain Expert Rankings by Algorithm & Dataset: Average Ratings

The first analysis evaluates the composite candidate expert scores calculated from the mean ratings of each candidate expert's top resources by authority score, up to a maximum of ten resources. Two-way between-subjects Analysis of Variance (ANOVA) was used to test the two sets of hypothesis. Figures 17 and 18 present the results of the analysis of the average ratings of the resources bookmarked by candidate experts. Please note that one observation is missing from each condition (n=119) due to the lack of data for one task by one of the subjects. Using the composite scores based on the mean ratings of candidate experts' top resources, we accept both null hypotheses, $H_{1\text{-}0}$ and $H_{2\text{-}0}$, as there is no significant difference in the means of the $nDCG_{10}$ of the candidate expert ratings among EARL, SPEAR, and HITS ($\mu_{EARL} = \mu_{SPEAR} = \mu_{HITS}$) at the $\alpha = .05$ level; nor is there a significant difference in the means of the $nDCG_{10}$ of the candidate expert ratings between the main and expert datasets ($\mu_{MAIN} = \mu_{EXPERT}$), $F(5, 713) = 2.382$, $p < .037$, and $\eta^2 = .017$.

Tests of Between-Subjects Effects

Dependent Variable: nDCG

Source	Type III Sum of Squares	df	Mean Square	F	Sig.	Partial Eta Squared
Corrected Model	.282[a]	5	.056	2.382	.037	.017
Intercept	365.703	1	365.703	15457.985	.000	.956
algorithm	.131	2	.065	2.765	.064	.008
dataset	.089	1	.089	3.751	.053	.005
algorithm * dataset	.062	2	.031	1.315	.269	.004
Error	16.750	708	.024			
Total	382.735	714				
Corrected Total	17.032	713				

a. R Squared = .017 (Adjusted R Squared = .010)

Figure 17. The results of the two-way between-subjects ANOVA for Experiment 2, mean ratings of candidate experts top bookmarked resources.

Descriptive Statistics

Dependent Variable: nDCG

algorithm	dataset	Mean	Std. Deviation	N
HITS	main	.7128998363	.1303539875	119
	expert	.6817978473	.1265224267	119
	Total	.6973488418	.1291250427	238
SPEAR	main	.7182108503	.1915201165	119
	expert	.7218690768	.1880527945	119
	Total	.7200399635	.1894024099	238
EARL	main	.7493571327	.1446863741	119
	expert	.7099081991	.1263691708	119
	Total	.7296326659	.1369835690	238
Total	main	.7268226064	.1580771095	357
	expert	.7045250410	.1503455154	357
	Total	.7156738237	.1545547423	714

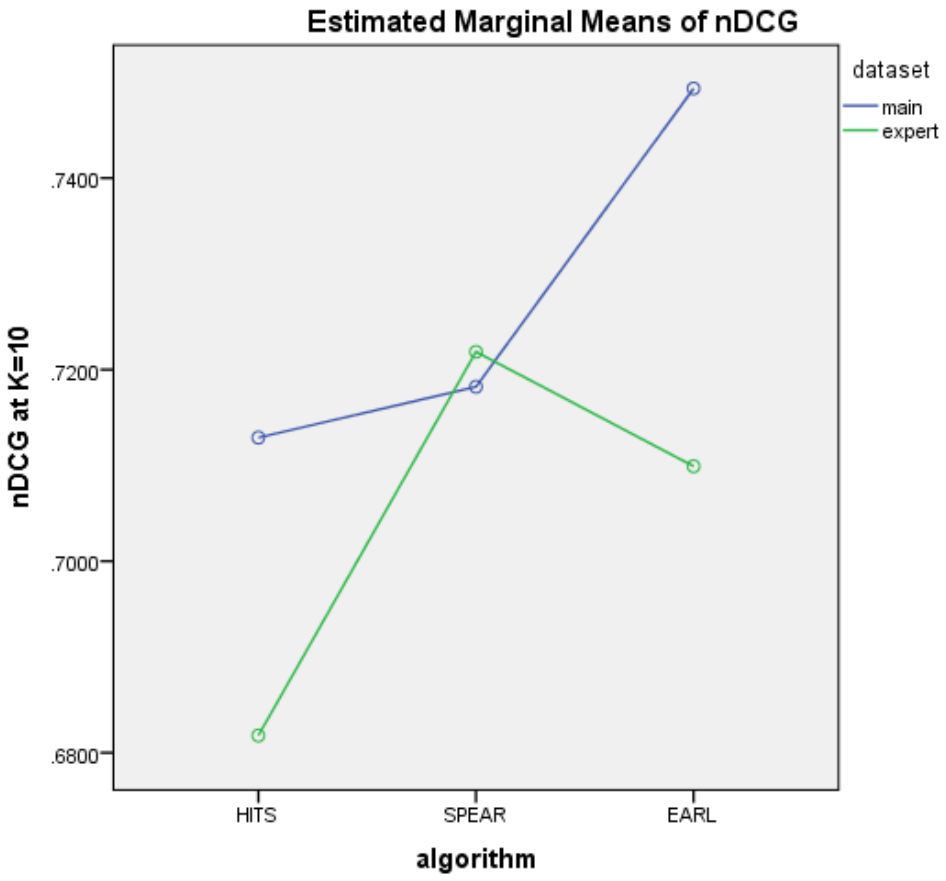


Figure 18. The means and standard deviations of $nDCG_{10}$ of candidate expert rankings for Experiment 2, mean ratings of candidate experts' top bookmarked resources.

5.4.2 Analysis of Domain Expert Rankings by Algorithm & Dataset: % of Highly-Rated Resources

The second analysis evaluates the composite candidate expert scores calculated from the percentage of high-quality resources (i.e., resources rated "mostly relevant" or higher by subjects, on average) bookmarked by each candidate expert for a given query, assuming a minimum of five resources. Two-way between-subjects Analysis of Variance (ANOVA) was used to test the two sets of hypothesis. Figures 19 and 20 present the results of the analysis of the candidate expert rankings based on the percentage of highly-rated resources bookmarked. Please note that one observation is missing from each condition (n=119) due to the lack of data for one task by one of the subjects.

Using the composite scores based on the percentage of highly-rated bookmarked resources for a given query, we reject both null hypotheses, $H_{1\text{-}0}$ and $H_{2\text{-}0}$. The means of the $nDCG_{10}$ of EARL, SPEAR, and HITS are significantly different ($\mu_{EARL} \neq \mu_{SPEAR} \neq \mu_{HITS}$) at the $\alpha = .05$ confidence level, as well as the means of the $nDCG_{10}$ of the main and expert datasets ($\mu_{MAIN} \neq \mu_{EXPERT}$), $F(5, 713) = 40.271$, $p < .001$, and $\eta^2 = .221$. Pairwise comparisons using the Scheffe procedure were then performed to determine the pattern of differences among the ranking algorithms. Because there are only two datasets, marginal comparisons were used in lieu of post-hoc comparisons with the Scheffe procedure to find the pattern of differences between the expert dataset (i.e., candidate filtering procedure applied) and main dataset (i.e., no filtering applied.) Figures 21 and 22 present the results of the comparisons of the ranking algorithms and filtering procedure, respectively.

Descriptive Statistics

Dependent Variable: nDCG

algorithm	dataset	Mean	Std. Deviation	N
HITS	main	.6360147097	.2545680161	119
	expert	.5676744241	.2338056264	119
	Total	.6018445669	.2462832756	238
SPEAR	main	.3978701732	.1996383364	119
	expert	.4150865320	.2144356864	119
	Total	.4064783526	.2069115386	238
EARL	main	.7239158564	.2513538711	119
	expert	.6262560722	.1752168848	119
	Total	.6750859643	.2216669192	238
Total	main	.5859335798	.2732344166	357
	expert	.5363390094	.2268969621	357
	Total	.5611362946	.2521843571	714

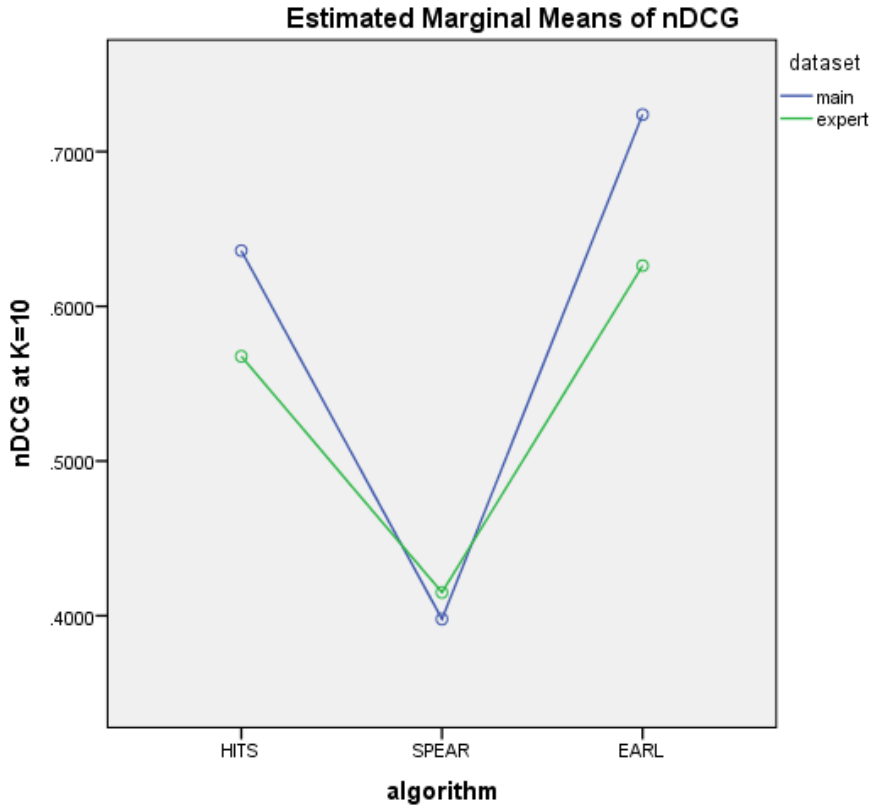


Figure 19. The means and standard deviations of the $nDCG_{10}$ of candidate expert rankings for Experiment 2, percentage of highly-rated resources bookmarked (*n*=714.)

Tests of Between-Subjects Effects

Dependent Variable: nDCG

Source	Type III Sum of Squares	df	Mean Square	F	Sig.
Corrected Model	10.040[a]	5	2.008	40.271	.000
Intercept	224.820	1	224.820	4508.606	.000
dataset	.439	1	.439	8.805	.003
algorithm	9.177	2	4.589	92.024	.000
dataset * algorithm	.424	2	.212	4.251	.015
Error	35.304	708	.050		
Total	270.165	714			
Corrected Total	45.345	713			

a. R Squared = .221 (Adjusted R Squared = .216)

Figure 20. The results of the two-way between-subjects ANOVA for Experiment 2, percentage of highly-rated resources bookmarked.

Among the ranking algorithms (Figure 21), the means of the $nDCG_{10}$ of EARL's candidate expert rankings were significantly higher than those of SPEAR ($p < .001$) and HITS ($p = .002$) across both datasets at the $\alpha = .05$ confidence level. We also note that the means of the $nDCG_{10}$ of SPEAR's candidate expert rankings were significantly lower than HITS' rankings ($p < .001$) across both datasets at the $\alpha = .05$ confidence level. The relatively poor performance of SPEAR in this analysis is largely due to the assumption that candidate experts are expected to bookmark at least five resources related to a given topic query, similar to the criteria used in Hilltop (Bharat and Mihaila, 2000.) SPEAR tends to rank highly users who are among the first to bookmark one or two very popular resources on a given topic, but have no other bookmarks related to that topic. Because the computation for this analysis used a five-resource minimum when calculating the percentage of high-quality resources bookmarked, many of SPEAR's top-ranked candidate experts actually had very few highly-relevant bookmarks in their collections.

Multiple Comparisons

Dependent Variable: nDCG
Scheffe

(I) algorithm	(J) algorithm	Mean Difference (I-J)	Std. Error	Sig.	95% Confidence Interval	
					Lower Bound	Upper Bound
HITS	SPEAR	.1953662143*	.0204702353	.000	.1451540680	.2455783606
	EARL	-.073241397*	.0204702353	.002	-.1234535437	-.0230292511
SPEAR	HITS	-.195366214*	.0204702353	.000	-.2455783606	-.1451540680
	EARL	-.268607612*	.0204702353	.000	-.3188197580	-.2183954654
EARL	HITS	.0732413974*	.0204702353	.002	.0230292511	.1234535437
	SPEAR	.2686076117*	.0204702353	.000	.2183954654	.3188197580

Based on observed means.
The error term is Mean Square(Error) = .050.

*. The mean difference is significant at the .05 level.

Figure 21. Comparisons to find significant differences in $nDCG_{10}$ of the candidate expert rankings among the ranking algorithms.

On the other hand, the results also suggest that EARL's additional expertise factors helped to improve candidate expert rankings.

For the comparisons of the use of the candidate expert filtering procedure (Figure 22) among the ranking algorithms, the means of the $nDCG_{10}$ of the candidate expert rankings using the filtering procedure (i.e., the expert dataset) were significantly lower than those of the main dataset for HITS, $F(1, 708) = 5.573$, $p = .019$, as well as for EARL, $F(1, 708) = 11.380$, $p = .001$. These results suggest that the candidate expert filtering procedure is likely removing users with domain expertise in the chosen topics, or at the very least, removing users who tend to selectively bookmark resources of higher quality.

Marginal Comparison: Main vs. Candidate Expert Dataset, HITS[a]

Contrast			Dependent Variable: nDCG
L1	Contrast Estimate		.068
	Hypothesized Value		0
	Difference (Estimate - Hypothesized)		.068
	Std. Error		.029
	Sig.		.019
	95% Confidence Interval for Difference	Lower Bound	.012
		Upper Bound	.125

Marginal Comparison: Main vs. Candidate Expert Dataset, SPEAR[a]

Contrast			Dependent Variable: nDCG
L1	Contrast Estimate		-.017
	Hypothesized Value		0
	Difference (Estimate - Hypothesized)		-.017
	Std. Error		.029
	Sig.		.552
	95% Confidence Interval for Difference	Lower Bound	-.074
		Upper Bound	.040

Marginal Comparison: Main vs. Candidate Expert Dataset, EARL[a]

Contrast			Dependent Variable: nDCG
L1	Contrast Estimate		.098
	Hypothesized Value		0
	Difference (Estimate - Hypothesized)		.098
	Std. Error		.029
	Sig.		.001
	95% Confidence Interval for Difference	Lower Bound	.041
		Upper Bound	.154

Figure 22. Comparisons to find significant differences in the $nDCG_{10}$ of the candidate expert rankings of the main dataset (no filtering procedure) versus the candidate expert dataset (filtering procedure used) for each ranking algorithm.

5.5 EXPERIMENT 3: CLASSIFICATION EXPERTISE AND RANKING OF TOPICS OF INTEREST

Experiment 3 evaluates a technique that filters candidate *classification* experts from Delicious, generates power sets of the candidate experts' tag sets, and selects frequently co-occurring terms shared by many candidate experts to classify resources. The goals of the third experiment are 1) to test the effectiveness of aggregating shared power sets among many users to find good classification terms, and 2) to test how well the candidate expert filtering procedure identifies Delicious users with classification expertise.

Twenty subjects were recruited to analyze twenty-five resources and rate the relevancy of a series of terms as keywords for each page. For each resource, terms were selected from corresponding ODP categories labels (ODP), the top ten tags by frequency in the main dataset (TOP10), the shared tag subsets derived from candidate experts' power sets (POWERSETS_EXPERTS), and the most-frequently occurring tag subsets derived from the resource's bookmark tag sets only (SUBSETS_RESOURCES.) The twenty subjects each rated the same twenty-five resources and corresponding sets of classification terms; however, the resources and the terms were presented in random order to each subject. Because the subjects are considered experts in this experiment, their term relevancy ratings are considered ideal.

The following hypotheses were tested in Experiment 3:

$H_{1\text{-}0}$: There is no statistically-significant difference among the means of the $nDCG_{10}$ of classification terms selected by candidate experts' power sets, top 10 tags from the main dataset, the ODP category terms, and the most-frequently co-occurring subsets of tags among a resource's tag sets. ($\mu_{POWERSETS\_EXPERT} = \mu_{TOP10} = \mu_{ODP} = \mu_{SUBSETS\_RESOURCE}$)

$H_{1\text{-}1}$: There is a statistically-significant difference among the means of the $nDCG_{10}$ of classification terms selected by candidate experts' power sets, top 10 tags by popularity from the main dataset, the ODP category terms, and the most-frequently co-occurring subsets of tags among a resource's tag sets. ($\mu_{POWERSETS\_EXPERT} \neq \mu_{TOP10} \neq \mu_{ODP} \neq \mu_{SUBSETS\_RESOURCE}$)

$H_{2\text{-}0}$: There is no statistically-significant difference between the mean percentages of candidate experts using high-quality tags (i.e., tags rated as "good" or "excellent" classification terms) on resources versus all users in the main dataset. ($\mu_{EXPERT\_RATINGS} = \mu_{MAIN\_RATINGS}$)

$H_{2\text{-}1}$: There is a statistically-significant difference between the mean percentages of candidate experts using high-quality tags on resources versus all users in the main dataset. ($\mu_{EXPERT\_RATINGS} \neq \mu_{MAIN\_RATINGS}$)

5.5.1 Analysis of Entry Questionnaire Response

Prior to the experiment, subjects were asked to fill out an entry questionnaire (Appendix C) similar to the one used by Syn (2010). Of the twenty subjects recruited to provide ratings of classification terms for Experiment 3, 65% were professional librarians; 25% were current Master of Library and Information Science (MLIS) students; one subject was an MLIS degree holder; and one subject was a current Ph.D. student. All current and former LIS students reported completing the Organization of Information course (LIS2005), while 43% reported completing the Introduction to Cataloging and Classification course (LIS2405).

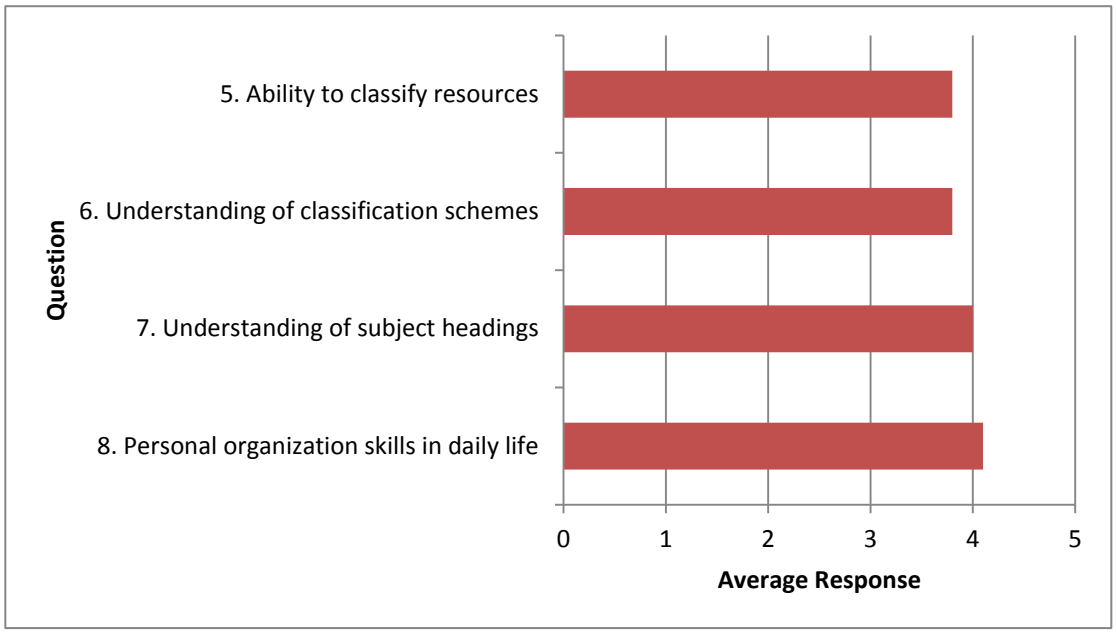


Figure 23. Average Ratings of Self-Assessment Questions (*n*=20.)

Figure 23 summarizes the responses to the self-assessment portion of the questionnaire. Subjects were asked to rate on a five-point scale – 1: Very Poor, 2: Poor, 3: Average, 4: Good, 5: Excellent – their own professional ability to classify resources ($\mu = 3.8$, $\sigma = 0.81$), understanding of the basics and concepts of classification schemes ($\mu = 3.8$, $\sigma = 0.81$), understanding of subjects headings ($\mu = 4.0$, $\sigma = 0.71$), and personal organization skills in daily life ($\mu = 4.1$, $\sigma = 1.03$.) Of the items subjects said they organize in their personal lives, the most commonly-cited items were personal computer files and folder (95%), personal documents (95%), and web pages (85%).

5.5.2 Analysis of Subjects' Ratings of Classificatory Terms: nDCG

The first analysis evaluates the performance of the four conditions (ODP, TOP10, POWERSETS_EXPERT, SUBSETS_RESOURCE) used to rank topics of interest based on

subjects' ratings of classificatory terms presented in random order. As described in section 4.6.5., composite ratings were computed for each POWERSETS_EXPERT and SUBSETS_RESOURCE item as the mean rating of the item's component terms. One-way Analysis of Variance (ANOVA) was used to test Experiment 3's first hypothesis of any significant differences in the performance of the four methods to rank topics of interest as measured by $nDCG_{10}$. Figure 24 present the results of the ANOVA. Figure 25 shows the means and standard deviations of the $nDCG_{10}$ of the four conditions' topics of interest rankings.

Based on the results of the ANOVA, we reject the null hypothesis $H_{1\text{-}0}$. There was a significant difference in the $nDCG_{10}$ means of the topics of interest rankings among ODP, TOP10, POWERSETS_EXPERT, SUBSETS_RESOURCE at the $\alpha = .05$ confidence level ($\mu_{POWERSETS\_EXPERT} \neq \mu_{TOP10} \neq \mu_{ODP} \neq \mu_{SUBSETS\_RESOURCE}$), $F(3, 99) = 11.975$, $p < .001$, and $\eta^2 = .272$. Post-hoc pairwise comparisons using the Scheffe procedure were then performed to determine the pattern of differences among the four conditions. As shown in Figure 26, the mean $nDCG_{10}$ of POWERSETS_EXPERT's topic of interest rankings was significantly lower than that of SUBSETS_RESOURCE, $p = .009$ at a confidence level of $\alpha = .05$. There were no significant

ANOVA

nDCG

	Sum of Squares	df	Mean Square	F	Sig.
Between Groups	.300	3	.100	11.975	.000
Within Groups	.801	96	.008		
Total	1.101	99			

Figure 24. The results of the one-way ANOVA for Experiment 3, means of the $nDCG_{10}$ of the four methods to rank topics of interest.

Descriptives

nDCG

	N	Mean	Std. Deviation	95% Confidence Interval for Mean	
				Lower Bound	Upper Bound
ODP	25	.7359650947	.1051700601	.6925530276	.7793771619
TOP10	25	.8299518422	.0504569555	.8091242346	.8507794498
POWERSETS_EXPERT	25	.7967743840	.1241557826	.7455253958	.8480233722
SUBSETS_RESOURCE	25	.8872178375	.0660283793	.8599626621	.9144730129
Total	100	.8124772896	.1054547820	.7915527730	.8334018062

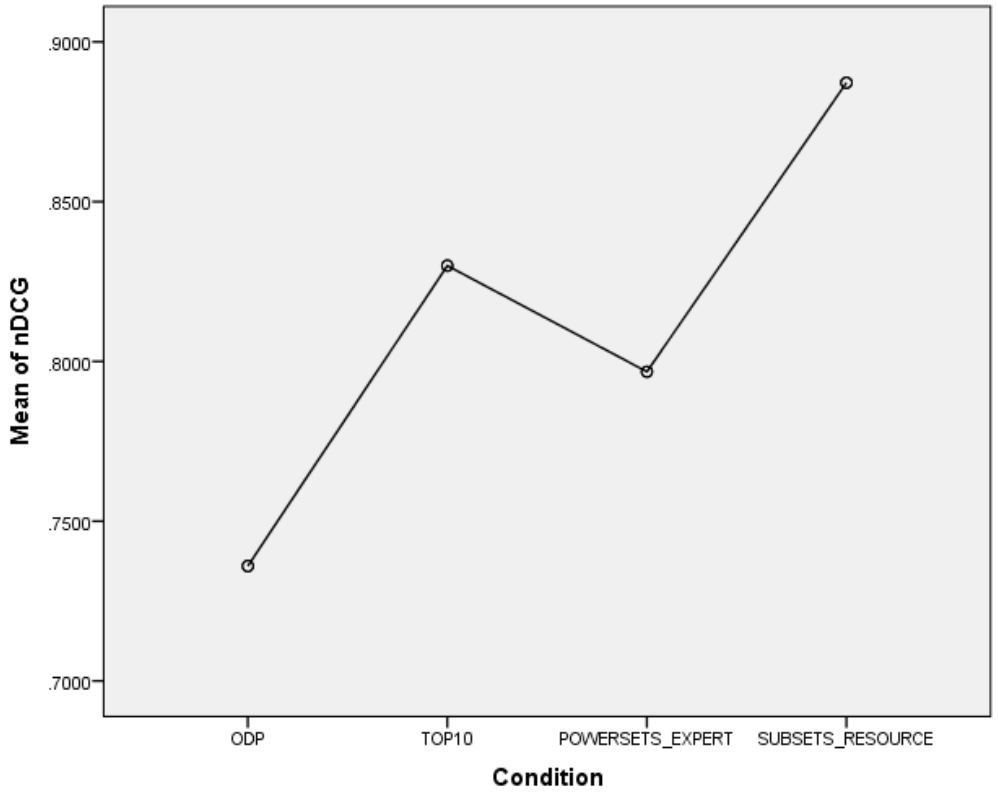


Figure 25. The means and standard deviations of the $nDCG_{10}$ of the topic of interest rankings, Experiment 3 (*n*=100.)

Multiple Comparisons

Dependent Variable: nDCG
Scheffe

(I) Condition	(J) Condition	Mean Difference (I-J)	Std. Error	Sig.	95% Confidence Interval	
					Lower Bound	Upper Bound
ODP	TOP10	-.093986747*	.0258383634	.006	-.1675157468	-.0204577482
	POWERSETS_EXPERT	-.0608092893	.0258383634	.144	-.1343382886	.0127197100
	SUBSETS_RESOURCE	-.151252743*	.0258383634	.000	-.2247817421	-.0777237435
TOP10	ODP	.0939867475*	.0258383634	.006	.0204577482	.1675157468
	POWERSETS_EXPERT	.0331774582	.0258383634	.650	-.0403515411	.1067064575
	SUBSETS_RESOURCE	-.0572659953	.0258383634	.186	-.1307949946	.0162630040
POWERSETS_EXPERT	ODP	.0608092893	.0258383634	.144	-.0127197100	.1343382886
	TOP10	-.0331774582	.0258383634	.650	-.1067064575	.0403515411
	SUBSETS_RESOURCE	-.090443453*	.0258383634	.009	-.1639724528	-.0169144542
SUBSETS_RESOURCE	ODP	.1512527428*	.0258383634	.000	.0777237435	.2247817421
	TOP10	.0572659953	.0258383634	.186	-.0162630040	.1307949946
	POWERSETS_EXPERT	.0904434535*	.0258383634	.009	.0169144542	.1639724528

*. The mean difference is significant at the 0.05 level.

Figure 26. Comparisons to find significant differences in $nDCG_{10}$ of the topic of interest rankings among the four conditions.

differences in the means of $nDCG_{10}$ of POWERSETS_EXPERT, TOP10, and ODP. These results suggest that although aggregating the shared, power-set-derived tag subsets of candidate experts identifies topics of interest comparable to Open Directory Project's category labels and the top ten tags in the main dataset for a given resource, the method's performance versus professionally-assigned metadata and individual Delicious tags does not justify the additional data processing.

On the other hand, the mean $nDCG_{10}$ of SUBSETS_RESOURCE's topic of interest rankings was significantly higher than that of ODP, $p < .001$ at a confidence level of $\alpha = .05$, but not significantly different from that of TOP10. These results suggest that using power sets to find frequently co-occurring subsets of tags on the resources identifies relevant classificatory terms than the Open Directory Project's category labels and the candidate experts' shared topics of interest. However, the resource tag subsets did not significantly outperform the top ten

individual resource tags from the main Delicious dataset, again suggesting that deriving frequently co-occurring tags subsets from power sets does not necessarily yield results that justify the processing expense.

5.5.3 Analysis of High-quality Tag Use by Candidate Experts vs. Average Delicious Users

The second analysis evaluates how well the candidate filtering procedure identifies classification experts in Delicious by comparing the mean percentages of high-quality tag use by candidate experts to those of average users in the main dataset. Using subjects' ratings on the twenty-five resources selected for Experiment 3, the tags with mean ratings of 4.00 or above (i.e., "good" to "excellent" classificatory terms) were identified for each resource. Then for each highly-relevant tag T_i on resource R_j, the percentages of candidate experts who used T_i on R_j were calculated and compared with the percentages of average users in the main dataset who used T_i on R_j. One-way Analysis of Variance (ANOVA) was used to test Experiment 3's second hypothesis of any significant differences in the percentages of high-quality tag use between candidate experts and average users. Figure 27 present the results of the ANOVA. Figure 28 presents the means and standard deviations of the percentages of high-quality tag use.

ANOVA

FreqPct

	Sum of Squares	df	Mean Square	F	Sig.
Between Groups	4.706	1	4.706	99.304	.000
Within Groups	11.278	238	.047		
Total	15.984	239			

Figure 27. The results of the one-way ANOVA for Experiment 3, percentages of high-quality tag use by candidate experts and average Delicious users.

Descriptives

FreqPct

		N	Mean	Std. Deviation	95% Confidence Interval for Mean	
					Lower Bound	Upper Bound
CANDIDATE_EXPERT		120	.5464359453	.2547320185	.5003912069	.5924806838
MAIN		120	.2663802204	.1728826266	.2351303784	.2976300624
Total		240	.4064080829	.2586116149	.3735232777	.4392928880
Model	Fixed Effects			.2176887731	.3787263951	.4340897706
	Random Effects				-1.372814606	2.185630772

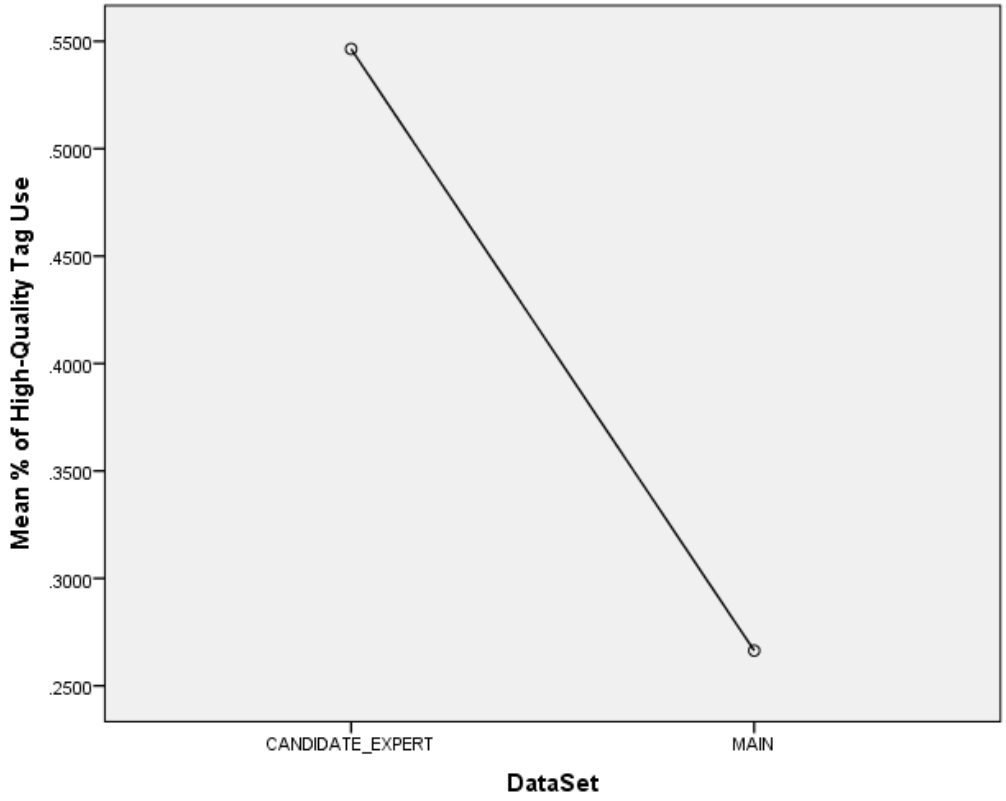


Figure 28. The means and standard deviations of the $nDCG_{10}$ of the topic of interest rankings, Experiment 3 (*n*=240.)

We reject the null hypothesis $H_{2\text{-}0}$ of Experiment 3, as the mean percentages of high-quality tag use by candidate experts were significantly higher than those of average users ($\mu_{EXPERT\_RATINGS} \neq \mu_{MAIN\_RATINGS}$) at the $\alpha = 0.05$ confidence level, $F(1, 239) = 99.304$, $p < .001$, η^2

= .294. These results confirm the preliminary findings presented in Table 11 (Section 3.4.1.) that candidate experts are more likely to annotate resources with high-quality tags than the average Delicious user. As discussed in section 3.4.1., the higher tag-usage agreement among candidate experts may simply be a by-product of the initial candidate expert selection process that focuses on prolific annotators, immediately eliminating users whose bookmarks contain no tags. However, we conclude the similar rank-order lists and greater agreement support the notion that the candidate expert selection process helps isolate classification expertise. Not only do candidate experts consistently annotate their bookmarks with multiple tags, but they also choose (*and* are more likely to choose) tags that reflect the beliefs of the Delicious community and independent expert judges.

5.6 DISCUSSION OF RESULTS

5.6.1 Authoritative Resource Rankings

The first experiment evaluated the authoritative resource rankings of four ranking algorithms – EARL, SPEAR, HITS, and Google – as well as a candidate expert filtering procedure for reducing noise in the Delicious data graph. With the inclusion of the normalized expert agreement and sustained popularity factors to improve topic relevance and demote obsolete resources, respectively, EARL's authoritative resource rankings were expected to outperform those of HITS and SPEAR, and be at least on par with Google's resource rankings. Instead, there were no significant differences in ranking performance among the three graph-based algorithms providing rankings from Delicious data whether the candidate expert filtering

procedure was used or not, and all three algorithms performed significantly worse than Google in all conditions.

One reason for EARL's poor performance against Google is the use of an indexing approach that only considered Delicious tags, not the content of the resources themselves. The indexing approach used in Experiments 1 and 2 was similar to the "annotation-indexing" approach tested by Choochaiwattana (2008), an approach found to be inferior to an indexing method that combined social annotations with the content of resources. Despite these findings, the annotation-only indexing approach was chosen for this research to ensure that 1) the factors that were the focal points of this work – the query-dependent ranking algorithms and candidate expert filtering approach – could be tested properly without a third, confounding factor; and 2) that the indexing approach would not bias the results in favor of any particular algorithm.

A second reason for EARL's, SPEAR's, and HITS' weak results compared to Google's was the presence of dead links in the ranked Delicious resource results. The most recent bookmarks in the Delicious datasets were collected in August of 2010. Experimental sessions with subjects began in April, 2012. Of the 2,891 unique resource URL's presented to subjects during the experimental sessions, 354 resources (12.2%) were no longer available to rate. While we believe Google would have still produced better resource rankings than the three query-dependent algorithms had all resources been available, the mean $nDCG_{10}$ of EARL, SPEAR, and HITS would have been greater, possibly changing the significance of the results.

As for EARL's performance against SPEAR and HITS, EARL's mean rankings did not meet expectations, as they were not significantly different from those of the other two algorithms. Neither the normalized expert agreement nor the sustained popularity factors appeared to have improved EARL's authoritative resource rankings. Normalized expert

agreement itself was expected to improve EARL's rankings because of its relation to topical relevance: the greater the percentage of candidate experts who agree (i.e. tagged) a resource is about the given topic, the more EARL promotes the resource in the rankings. While it may be worth revisiting the relative weight of this factor for EARL's rankings, another possible solution is to look at how concentrated a resource is on a given topic. Similar to normalized expert agreement, if a greater percentage of a resource's *total* tag instances match a given query, the more EARL will promote the resource in its rankings. This factor reflects subjects' ratings of resources in Experiments 1 and 2, who tended to provide the highest ratings to resources whose content focused specifically on their search topic.

Finally, the results for the candidate expert filtering procedure in Experiment 1 are mixed. EARL's and SPEAR's resource rankings using the filtered candidate expert dataset were not significantly different from their rankings utilizing all Delicious data from the main dataset. Although the rankings did not improve with the filtering procedure, the results suggest we can use a smaller subset of the Delicious graph with EARL and SPEAR to provide ranked result lists of comparable quality more efficiently. However, the mean $nDCG_{10}$ of HITS' resource rankings was significantly lower with the candidate expert filtering procedure applied. The fact that the filtering procedure only affects HITS' ranking performance significantly reflects the impact of the long tail of average Delicious users on HITS' rankings. Although each user removed by the filtering procedure bookmarked few resources (i.e., no more than ten), these users tend to bookmark popular, highly-rated resources. Eliminating these users and their bookmarks on popular resources means fewer inlinks to these resources, reducing the resources' authority scores, and leading to the relatively poor performance of HITS' rankings with the candidate expert dataset.

5.6.2 Domain Expert Rankings

Experiment 2 evaluated the domain expert rankings of the three query-dependent, graph-based algorithms, and the ability of the candidate expert filtering procedure to isolate domain expertise and reduce noise in the Delicious data graph. Given the four criteria described in Section 3.3.2 to establish the weights for each link in EARL's adjacency lists, EARL's domain expertise rankings were expected to outperform those of HITS and SPEAR. When we computed the mean $nDCG_{10}$ of each algorithm's domain expert rankings using a composite score based on the mean rating of users' top ten resources by authority score, the ranking performance of the three algorithms showed no significant differences. However, when we calculated users' composite scores based on the percentage of high-quality resources bookmarked on the given topic with a five-resource minimum - as used by Bharat and Mihaila (2000) to define "expert" documents in Hilltop's index - EARL's domain expert rankings were significantly better than those of HITS' and SPEAR's, confirming our expectations. The second method for calculating the composite domain expert scores is more consistent with the expertise model presented in this research: domain experts in Delicious on a given topic should provide many bookmarks to highly-relevant resources consistently over time.

We note that SPEAR's ranking performance decreased dramatically when we utilized the percentage of high-quality resources as the composite score for calculating $nDCG_{10}$. This drop is due to the five-resource minimum used to calculate the composite score. Because temporal bookmarking sequence is the only additional weight factor, SPEAR tends to strongly promote users who bookmark only one or two very popular resources on a given topic, but do so early in each resource's history. Therefore, if a user bookmarked just two resources on a given topic, the highest composite score that user can receive in this scenario is 0.4 (2/5), assuming both

resources are highly relevant to the topic. We expect a domain expert on a given topic to consistently find and bookmark many highly-relevant resources over time, not just a few obvious resources.

Overall the results for the candidate expert filtering procedure in Experiment 2 suggest the procedure actually harms domain expert rankings, especially for those of EARL and HITS. EARL's and HITS' domain expert rankings generated from the filtered candidate expert dataset were lower than those produced from the main dataset under both methods of calculating users' composite scores, although the mean $nDCG_{10}$ of the scores based on the mean ratings of users' top ten resources by authority score were not significant at the 95% confidence level. The results suggest the filtering procedure removes domain experts who provide important resources on a given topic, but whose classification skills outside their area(s) of domain expertise were not strong enough for them to be considered classification experts.

5.6.3 Classification Expertise and Rankings of Topics of Interest

Experiment 3 analyzed the candidate expert filtering procedure's ability to filter classification experts from Delicious, as well as a technique that generates power sets of the candidate experts' tag sets, selecting frequently co-occurring terms shared by many candidate experts as topics of interest. Based on the preliminary analysis, we expected the candidate experts to annotate their bookmarks with highly-relevant tags more often than the average Delicious user. We also expected the candidate experts' shared topics of interest derived from power sets of tags used throughout all experts' bookmark collections would identify highly relevant classificatory terms for a given resource better than expert-generated ODP category labels or individual tags applied by average Delicious users. As expected, the candidate experts did, on average, apply highly-

relevant tags to a given resource with significantly greater frequency than the average Delicious user. This provides evidence that the candidate expert filtering procedure locates Delicious users with characteristics we expect of a classification expert: the consistent use of multiple terms to describe a resource's content, and the consistent application of terms that accurately reflect the topic(s) of a resource.

However, the mean $nDCG_{10}$ of the classificatory term rankings produced from the candidate experts' shared topics of interest were not significantly different from those of the ODP category labels and individual tags applied by average Delicious users. In fact, the mean rankings of candidate experts' shared topics of interest performed significantly worse than the rankings of frequently co-occurring subsets of tags identified by the same power set technique but used only on the bookmarks of individual resources. As described in section 4.6.1, the technique for selecting and ranking matching topics of interest on a particular resource chose topics based on how frequently they were shared by candidate experts throughout the *entire dataset*, not for the particular resource itself. The matching candidate experts' topics of interest tended to produce more general terms that were only moderately relevant to the resource due to the lack of focus on that particular resource. As the expert judges' ratings indicated, using the same power set technique on a more focused, per-resource basis identified and ranked frequently co-occurring subsets tags that better represented the topical nature of resources, on average. This suggests that 1) there is utility to using power sets to find good classificatory terms, and 2) that further exploration is necessary to see if the shared experts' topics of interest could be used to generate the higher levels of a classification scheme for resources in Delicious.

6.0 CONCLUSION

This chapter presents the conclusions of this research, including its contributions and implications. Plans for future work are also discussed.

6.1 CONTRIBUTIONS & IMPLICATIONS

This dissertation analyzes data from the Delicious social bookmarking system to find the most authoritative documents and expert users in Delicious for a given topic. Given the amount of noise in social bookmarking systems – irrelevant tags on resources, untagged resources, and users who abandon the system after little use – this research developed a novel algorithm, EARL, to better identify authoritative documents and expert users in these systems. The major questions addressed by this research include:

- Using a model to identify both domain and classification expertise, can a novel algorithm be developed to identify the best experts and most authoritative documents in Delicious on a given topic more *accurately* than existing algorithms?
- Can noise in the Delicious data graph be reduced, allowing an algorithm to better locate expert users and authoritative documents?

- Can extracting power sets from bookmark tag sets produce meaningful subsets of tags that represent users' topics of interest?

In the first phase of EARL, we reduce noise in the Delicious data by isolating a smaller sub-network of "candidate experts", users whose tagging behavior shows potential domain *and* classification expertise. In the second phase, a HITS-based graph analysis is performed on the candidate experts' data to rank the top experts and authoritative documents by topic. To identify topics of interest in Delicious, this research proposed and used a distributed method for finding the power sets of bookmark tag sets to identify subsets of frequently co-occurring tags shared among many candidate experts. Based on preliminary analyses of EARL and the method for finding topics of interest, the assumptions prior to the formal evaluations were that EARL's more explicit model of expertise and resource authoritativeness would produce superior rankings of authoritative resources and domain experts when compared to those of other HITS-based algorithms, as well as Google's ranking of resources. This research also assumed that the candidate filtering procedure would effectively reduce noise in the Delicious data graph, also contributing to comparable or superior ranking of domain experts and authoritative resources. Finally, the use of power sets to generate frequently co-occurring subsets of tags shared by many candidate experts would identify relevant topics of interest better than expert-generated metadata and individual Delicious tags.

Using human judges' relevancy ratings of resources related to a series of Java programming topics, the first evaluation found that EARL's rankings of authoritative documents were comparable to HITS and SPEAR, but significantly underperformed the rankings of Google. We note that this study, to the best of our knowledge, was the first to evaluate SPEAR's ability to rank resources. We also observed that the candidate expert filtering procedure had no effect

on EARL's or SPEAR's resource rankings, but led to significantly worse HITS rankings compared to those when the filtering procedure was not applied.

In the second evaluation focusing on domain expertise, there were no differences observed in the rankings of domain experts among the three HITS-based algorithms when composite candidate expert scores calculated from the mean ratings of each candidate expert's top-ten resources by authority score were used. At the same time, the candidate expert filtering procedure had no effect on domain expert rankings. However, when we calculated the composite candidate expert scores based on the percentage of high-quality human-rated resources bookmarked by each candidate expert, EARL produced the best domain expert rankings among the three algorithms. We also observed in this scenario that the candidate expert filtering procedure significantly decreased the performance of EARL and HITS (but not SPEAR), leading them to produce worse rankings compared to conditions without filtering.

The third evaluation analyzed the effectiveness of the distributed, power-set-based method to identify topics of interest that are highly relevant to a given resource, as well as the ability of the candidate expert filtering procedure to isolate Delicious users with classification expertise. Using human judges' relevancy of ratings of classificatory terms on a series of resources, we found that candidate experts' shared topics of interest identified high-quality classificatory terms no better than expert-generated ODP category labels and the top ten individual tags of each resource. We also observed that frequently co-occurring subsets of tags generated solely from the power sets of a resource's bookmarks identified and ranked high-quality topics of interest better than the candidate experts' globally-shared topics of interest and the expert-generated metadata. Finally, we observed that the candidate expert filtering procedure

does isolate Delicious users who, on average, use high-quality tags on their bookmarks with significantly greater frequency than the typical Delicious user.

While social bookmarking systems provide a way for people to bookmark and annotate useful resources on a given topic, the level of noise in these systems can prevent users from locating potentially useful, accurately-annotated information. To address this issue, this research contributes a model of expertise in the context of a social bookmarking system that helps reduce noise in the tag data's graph. The EARL algorithm that implements this model is another contribution to the small, but growing body of literature on expertise in social bookmarking systems. We also believe the evaluation framework for assessing the domain expertise and authoritative resource rankings of graph-based algorithms on social bookmarking data is a valuable tool for future research. Based on the evaluation, we conclude that EARL can identify domain experts in the Delicious social bookmarking systems better than existing methods, but more work remains to be done to improve resource and topics of interest rankings.

6.2 FUTURE WORK

Given the mixed results of the candidate expert filtering procedure, resource rankings, and the method for selecting classification terms based on shared topics of interest among users, we plan to pursue the following research directions to address more questions related to this work.

First, improved indexing methods beyond simple annotation-based indexing will be implemented and tested. Currently, EARL's index only considers the tags placed on bookmarks of a particular resource. Methods that incorporate the full text of the resource – e.g., document title, major headings and the subsequent text of the sections – will be evaluated to determine how

both resource and expert rankings are influenced. We will also modify EARL to rank resources based on query similarity beyond simply exact matches. We believe both annotation-based indexing and the reliance on exact matching to users' queries contributed to the poor performance of EARL's, SPEAR's, and HITS' resource rankings to Google's.

Second, we plan to further refine and tune the expertise model of EARL to improve authoritative document and domain expert ranking performance. Additional studies will be conducted to analyze the impact of each of EARL's four weighting criteria on the overall weights. Measurement of the effects of each criterion was not included in this study's design.

Third, new approaches to the candidate expert filtering procedure will be developed and evaluated. The current procedure had no effect on EARL's resource rankings, and significantly reduced the performance of EARL's expert rankings based on the percentage of high-quality resources bookmarked. Using the topics of interest generated for Experiment 3, development will focus on filtering techniques that identify users with domain and classification expertise within specific topics, not based solely on general, topic-independent statistics. Although such approaches will be more processing-intensive, they will be more likely to retain a greater percentage of domain experts who use high-quality tags on a particular topic without punishing them for less rigorous tag use on resources outside their areas of expertise.

Fourth, EARL and the distributed method for identifying topics of interest will be tested using data from the current version of Delicious, as well as data from other social bookmarking systems, such as CiteULike. The Delicious data collected for this research was gathered prior to Delicious' sale to AVOS systems in December, 2011. Many inactive users and their bookmarks were removed from Delicious during the transition, although the exact number is unknown. With these changes to the Delicious data graph and the website's design, we will evaluate EARL

and the distributed method for identifying topics of interest to test if they are general enough to apply to any social bookmarking system.

Finally, although topics of interest shared among many users did not identify relevant classificatory terms better than other expert-generated metadata or individual tags, we will investigate other potential uses for these frequently co-occurring subsets of tags. Classification schemes are one such use, as these globally-shared tag subsets tend to describe more general topics of interest. We will also test different values of the threshold for determining when the component terms of a topic of interest are deemed frequently co-occurring.

APPENDIX A

INTER-RATER RELIABILITY FOR EXPERIMENTS 1 & 2

Question	Resource URL	1	2	3	4	5	# of raters	P_i
A	http://en.w ikipedia.org/w iki/Sorting_algorithm	0	0	0	1	19	20	0.9
A	http://cg.scs.carleton.ca/~morin/misc/sortalg/	1	0	3	4	11	19	0.374269
A	http://people.cs.ubc.ca/~harrison/Java/sorting-dem	0	1	0	4	14	19	0.5672515
A	http://csunplugged.org/sorting-algorithms	2	4	3	8	2	19	0.2280702
A	http://w w w .cs.princeton.edu/~rs/strings/	6	5	3	1	1	16	0.2333333
A	http://w w w .unicode.org/reports/tr10/	7	5	1	2	1	16	0.2666667
A	http://googleresearch.blogspot.com/2006/06/extra-e	8	4	2	1	1	16	0.2916667
A	http://w w w .cprogramming.com/tutorial/computersc	0	1	3	6	9	19	0.3157895
A	http://lbrandy.com/blog/2008/10/algorithms-in-real-lif	3	7	5	1	3	19	0.2163743
A	http://david-royal-martin.blogspot.com/2008/11/sorti	16	0	0	0	0	16	1
A	http://en.w ikipedia.org/w iki/Category:Sorting_algorit	0	0	0	3	7	10	0.5333333
A	http://dukesoferl.blogspot.com/2009/07/osmos.html	6	6	3	1	0	16	0.275
A	http://corte.si/posts/code/timsort/	0	3	6	3	4	16	0.225
A	http://w w w .catonmat.net/blog/three-beautiful-quick	1	3	3	8	4	19	0.2339181
A	http://w w w .cs.ubc.ca/~harrison/Java/sorting-demo	0	1	2	4	12	19	0.4269006
A	http://corte.si/posts/code/visualisingsorting/index.ht	0	2	0	4	10	16	0.4333333
A	http://w w w .cs.princeton.edu/courses/archive/spr0	1	1	7	3	4	16	0.25
A	http://betterexplained.com/articles/sorting-algorithms	0	0	2	3	6	11	0.3454545
A	http://w iki.python.org/moin/How To/Sorting	1	4	6	1	4	16	0.225
A	http://w w w .cs.princeton.edu/~rs/AlgsDS07/04Sort	0	0	1	2	8	11	0.5272727
A	http://w w w .iti.fh-flensburg.de/lang/algorithmen/sort	1	0	4	2	9	16	0.3583333
A	http://w w w .evanmiller.org/how -not-to-sort-by-aver	8	4	5	1	1	19	0.2573099
A	http://w w w .sorting-algorithms.com/	0	1	1	5	12	19	0.4444444
A	http://w w w .codinghorror.com/blog/archives/00101	4	4	6	1	4	19	0.1929825
A	http://iaroslavski.narod.ru/quicksort/	8	7	2	0	2	19	0.2982456
A	http://sortvis.org/index.html	0	4	3	5	7	19	0.2339181
A	http://linux.w ku.edu/~lamonml/algor/sort/sort.html	16	1	0	1	1	19	0.7017544
A	http://w w w .algolist.net/Algorithms/	1	1	2	3	9	16	0.3333333
A	http://vision.bc.edu/~dmartin/teaching/sorting/anim-h	19	0	0	0	0	19	1
A	http://w w w .bitw iese.de/2007/06/highly-efficient-4-	3	3	3	5	5	19	0.1695906
A	http://w w w .dangermouse.net/esoteric/intelligentdes	3	9	2	1	1	16	0.3333333
A	http://epaperpress.com/sortsearch/dow nload/sorts	1	1	1	1	11	15	0.5238095
A	http://w w w .igvita.com/2009/03/26/ruby-algorithms-	1	6	6	0	6	19	0.2631579
A	http://w w w .youtube.com/w atch?v=JdXoUgY QebM	5	3	5	3	3	19	0.1695906

Question	Resource URL	1	2	3	4	5	# of raters	P_i
A	http://atschool.eduweb.co.uk/mbaker/sorts.html	0	0	0	4	15	19	0.6491228
A	http://news.ycombinator.com/item?id=478632	10	4	2	0	0	16	0.4333333
A	http://www.i-programmer.info/news/150-training-a	0	0	4	6	6	16	0.3
A	http://www.iti.fh-flensburg.de/lang/algorithmen/sort	6	4	5	0	1	16	0.2583333
A	http://www.cs.ubc.ca/spider/harrison/Java/sorting	0	0	2	3	11	16	0.4916667
A	http://c2.com/cgi/wiki?SortingAlgorithms	0	0	1	7	11	19	0.4444444
A	http://www.nihilogic.dk/labs/sorting_visualization/	1	5	5	3	5	19	0.1929825
A	http://home.westman.wave.ca/~rhenry/sort/	5	1	2	3	8	19	0.245614
A	http://www.davekoelle.com/alphanum.html	3	4	5	3	4	19	0.1637427
A	http://epaperpress.com/sortsearch/	1	1	1	2	5	10	0.2444444
A	http://www.hatfulofhollow.com/posts/code/timsort/	1	3	6	2	7	19	0.2339181
A	http://www.sorting-algorithms.com/?	0	0	1	7	11	19	0.4444444
A	http://www.cs.rit.edu/~atk/Java/Sorting/sorting.htm	0	0	1	10	8	19	0.4269006
A	http://www.hatfulofhollow.com/posts/code/visualis	0	1	2	6	10	19	0.3567251
A	http://www.concentric.net/~ttwang/sort/sort.htm	0	0	1	7	10	18	0.4313725
A	http://en.wikipedia.org/wiki/Trie	5	4	4	3	0	16	0.2083333
A	http://coderaptors.com/?Sorting_algorithms	1	0	2	4	9	16	0.3583333
A	http://www.cs.princeton.edu/~rs/	7	6	3	2	1	19	0.2339181
A	http://www.math.ucla.edu/~rcompton/musical_sorti	3	2	5	3	3	16	0.1666667
A	http://users.aims.ac.za/~mackay/sorting/sorting.htm	0	0	7	3	6	16	0.325
A	http://www.topcoder.com/tc?module=Static	2	4	6	1	3	16	0.2083333
B	http://www.openlaszlo.org/	6	5	3	1	2	17	0.2132353
B	http://tiny.spket.com/	2	5	6	2	3	18	0.1960784
B	http://sourceforge.net/projects/rubyeclipse	6	3	4	4	0	17	0.2205882
B	http://netbeans.org/features/java/profiler.html	0	0	2	2	6	10	0.3777778
B	http://www.jetbrains.com/	6	6	4	1	1	18	0.2352941
B	http://www.jedit.org/	3	11	4	1	0	19	0.374269
B	http://www.jformdesigner.com/	2	3	1	5	7	18	0.2287582
B	http://www.borland.com/jbuilder/	13	2	4	0	0	19	0.497076
B	http://www.myeclipseide.com/	2	2	5	3	6	18	0.1960784
B	http://netbeans.org/features/java/javase.html	0	1	1	4	6	12	0.3181818
B	http://www.phpeclipse.de/	16	1	1	0	0	18	0.7843137
B	http://eclipsesql.sourceforge.net/index.php	4	7	3	4	0	18	0.2352941
B	http://pollo.sourceforge.net/	11	5	2	0	0	18	0.4313725
B	http://www.wavemaker.com/	12	4	2	0	1	19	0.4269006
B	http://www.aquafold.com/	9	7	1	0	1	18	0.372549
B	https://abeille.dev.java.net/	15	1	1	0	0	17	0.7720588
B	http://tivohme.sourceforge.net/	8	3	6	1	0	18	0.3006536
B	http://www.netbeans.org/downloads/index.html	0	1	3	4	10	18	0.3529412
B	https://netbeans-opengl-pack.dev.java.net/	18	0	0	0	0	18	1
B	http://www.eclipse.org/buckminster/	8	3	4	2	1	18	0.248366
B	http://www.borland.com/	12	3	2	0	1	18	0.4575163
B	http://www.objectcentral.com/vide.htm	15	2	1	0	0	18	0.6928105
B	http://www.refactorit.com/	18	0	0	0	0	18	1
B	http://www-128.ibm.com/developerworks/opensou	2	4	6	2	4	18	0.1895425
B	http://www.aptana.com/	3	5	2	5	3	18	0.1764706
B	http://www.omnicore.com/	16	1	0	0	0	17	0.8823529
B	http://www.nbextras.org/	11	1	2	4	0	18	0.4052288
B	http://netbeans.org/kb/docs/ide/java-db.html	1	4	3	4	0	12	0.2272727
B	http://www.willryan.co.uk/WWWorkspace/	7	1	3	5	2	18	0.2287582
B	http://www.yourkit.com/index.jsp	8	4	3	3	0	18	0.2614379
B	http://www.yoxos.com/ondemand/	4	4	5	3	2	18	0.1699346
B	http://www.jetbrains.com/idea/features/ruby_deve	2	3	3	7	3	18	0.2026144
B	http://jvi.sourceforge.net/	2	5	4	6	1	18	0.2091503
B	http://alexdp.free.fr/violetumleditor/page.php?id=fr:u	14	3	1	0	0	18	0.6143791
B	http://www.eclipse.org/	3	8	4	3	2	20	0.2157895

Question	Resource URL	1	2	3	4	5	# of raters	P_i
B	http://mevenide.codehaus.org/	4	5	4	5	0	18	0.2091503
B	http://www.intelligentedu.com/blogs/post/Best_New	2	3	5	2	6	18	0.1960784
B	http://www.aspectprogrammer.org/blogs/adrian/20	2	1	5	6	4	18	0.2091503
B	http://www.axiomsol.com/	14	2	2	0	0	18	0.6078431
B	http://www.eclipse.org/pulsar/	7	5	5	1	1	19	0.2397661
B	http://aptana.com/	2	2	7	2	6	19	0.2280702
B	http://www.jcreator.com/	1	0	2	8	8	19	0.3333333
B	http://springide.org/project	19	0	0	0	0	19	1
B	http://www.greenfoot.org/	5	2	9	0	3	19	0.2923977
B	http://www.junit.org/news/article/index.htm	11	4	2	0	1	18	0.4052288
B	http://jdee.sourceforge.net/	1	1	4	6	7	19	0.245614
B	http://dmy999.com/article/29/using-eclipse-efficientl	2	2	5	6	3	18	0.1960784
B	http://www.eclipsezone.com/	6	6	6	0	1	19	0.2631579
B	http://www.plentyofcode.com/2007/07/most-usefu	16	1	1	0	0	18	0.7843137
B	http://blogs.sun.com/cwebster/entry/netbeans_6_w	16	0	1	0	0	17	0.8823529
B	http://www.eclipseplugincentral.com/displayarticle4	2	7	5	1	3	18	0.2287582
B	http://help.eclipse.org/galileo/index.jsp	2	2	5	4	5	18	0.1830065
B	http://www.eclipse.org/downloads/	0	1	6	2	9	18	0.3398693
B	http://www.easyeclipse.org/site/home/	1	5	3	5	5	19	0.1929825
B	http://www.slickedit.com/	6	3	6	1	2	18	0.2222222
B	http://www.jsurfer.org/	16	2	0	0	0	18	0.7908497
B	http://syntori.com/mochacode/	1	3	4	7	3	18	0.2156863
B	http://www.eclipse.org/downloads/moreinfo/jee.ph	0	2	3	1	5	11	0.2545455
B	http://www.cs.brown.edu/people/acb/codebubbles	15	0	2	0	1	18	0.6928105
B	http://www.mindview.net/WebLog/wiki-0047	18	0	0	0	0	18	1
B	http://www.jetbrains.org/display/IJOS/Home	2	2	5	5	3	17	0.1838235
B	http://www.springsource.com/products/sts	3	3	4	3	5	18	0.1633987
B	http://www.cs.brown.edu/people/acb/codebubbles	15	4	0	0	0	19	0.6491228
B	http://netbeans.dzone.com/	3	5	5	3	2	18	0.1764706
B	http://ejp.sourceforge.net/	7	5	2	3	1	18	0.2287582
B	http://ant.apache.org/	7	3	5	3	0	18	0.2418301
B	http://code.google.com/p/counterclockwise/	5	7	6	0	0	18	0.3006536
B	http://www.netbeans.org/kb/trails/java-se.html	2	3	4	5	4	18	0.1699346
B	http://www.eclipseplugincentral.com/	5	3	5	3	3	19	0.1695906
B	http://eclipsewiki.editme.com/	18	0	0	0	0	18	1
B	http://www.vogella.de/eclipse.html	0	3	4	6	5	18	0.2222222
B	http://netbeans.org/kb/docs/java/quickstart.html	0	0	1	4	12	17	0.5294118
B	http://mevenide.codehaus.org/mevenide-ui-eclipse/f	8	4	4	2	0	18	0.2679739
B	http://www.jgrasp.org/	3	3	5	5	2	18	0.1764706
B	http://www.xored.com/trustudio	18	0	1	0	0	19	0.8947368
B	http://maven.apache.org/guides/mini/guide-ide-eclip	1	5	6	6	1	19	0.2339181
B	http://www.gexperts.com/	10	4	4	0	0	18	0.372549
B	http://code.google.com/p/yamleditor/	6	7	4	1	0	18	0.2745098
B	http://blogs.sun.com/roller/page/toddfast/20041203#	17	0	0	0	0	17	1
B	http://www.planetnetbeans.org/	3	5	5	5	0	18	0.2156863
B	http://www.eclipseproject.de/	18	0	0	0	0	18	1
B	http://www.eclipse.org/downloads/packages/eclip	1	0	1	4	6	12	0.3181818
B	http://www.eclipse.org/webtools/	4	3	5	3	3	18	0.1633987
B	http://www.borland.com/us/products/jbuilder/index	12	4	0	2	0	18	0.4771242
B	http://www.netbeans.org/	0	0	2	2	16	20	0.6421053
B	http://marketplace.eclipse.org/	4	6	2	4	2	18	0.1895425
B	http://eclim.sourceforge.net/	2	4	7	4	2	19	0.2046784
B	http://jcsc.sourceforge.net/	5	2	7	3	1	18	0.2287582
B	http://www.eclipse.org/downloads/packages/eclip	0	2	5	6	5	18	0.2352941
B	http://www.omondo.com/	5	6	3	2	2	18	0.1960784
B	http://eclipse-plugins.2y.net/eclipse/index.jsp	14	2	1	0	0	17	0.6764706

Question	Resource URL	1	2	3	4	5	# of raters	P_i
B	http://en.wikipedia.org/wiki/Eclipse_(software)	0	0	3	2	13	18	0.5359477
B	http://quantum.sourceforge.net/	3	6	9	0	0	18	0.3529412
B	http://www.elixirtech.com/	12	3	3	0	0	18	0.4705882
B	http://code.google.com/p/q4e/	2	4	6	4	2	18	0.1895425
B	http://netbeans.org/kb/docs/java/profiler-intro.html	0	0	2	4	4	10	0.2888889
B	http://www.codegear.com/	6	7	2	1	2	18	0.248366
B	http://www.devx.com/Java/Article/34009/	8	0	3	3	4	18	0.2614379
B	http://www.gentleware.com/uml-software-commun	7	5	6	0	0	18	0.3006536
B	http://eclipse-plugins.info/eclipse/plugins.jsp	16	1	0	1	0	18	0.7843137
B	http://springide.org/blog/	19	0	0	0	0	19	1
B	http://www.mpsoftware.dk/phpdesigner.php	8	7	3	1	0	19	0.3040936
B	http://javaforge.com/project/HGE	2	4	8	1	3	18	0.248366
B	http://netbeans.org/kb/docs/java/editor-codereferer	0	0	1	5	5	11	0.3636364
B	http://www.bluej.org/index.html	3	2	7	3	4	19	0.1988304
B	http://www.gentleware.com/	12	4	1	1	0	18	0.4705882
B	http://www.netbeans.org/switch/	0	1	4	6	8	19	0.2865497
B	http://www.eclipse.org/downloads/moreinfo/java.p	0	0	4	3	12	19	0.4385965
B	http://www.apl.jhu.edu/~hall/java/IDEs.html	0	0	2	1	8	11	0.5272727
B	http://www.easyeclipse.org/site/distributions/index	1	2	4	7	5	19	0.2222222
B	http://www.jetbrains.com/idea/index.html	0	1	2	5	12	20	0.4052632
B	http://www.oracle.com/technology/products/jdev/ir	2	1	6	2	8	19	0.2631579
B	http://www.devdirect.com/ALL/CODEDEBUG_PCA	5	6	4	3	0	18	0.2222222
B	http://netbeans.org/features/web/java-ee.html	0	1	1	5	10	17	0.4044118
B	http://www.oracle.com/technology/products/enterp	2	4	3	6	3	18	0.1830065
C	http://rymden.nu/exceptions.html	1	0	1	3	11	16	0.4833333
C	http://www.octopull.demon.co.uk/java/ExceptionalJ	18	0	0	0	0	18	1
C	http://radio.weblogs.com/0122027/stories/2003/04/(	0	3	8	6	1	18	0.3006536
C	http://www.javabeginner.com/java-exceptions.htm	0	1	1	1	16	19	0.7017544
C	http://onjava.com/pub/a/onjava/2003/11/19/exceptio	3	0	2	8	5	18	0.2745098
C	http://today.java.net/pub/a/today/2006/04/06/except	0	1	2	6	11	20	0.3736842
C	http://www.subbu.org/weblogs/welcome/2005/07/(	16	0	0	0	0	16	1
C	http://docs.oracle.com/javase/6/docs/api/java/lang/E	0	0	1	5	5	11	0.3636364
C	http://docs.oracle.com/javase/tutorial/essential/exce	0	0	0	2	18	20	0.8105263
C	http://cafe.elharo.com/java/internal-and-external-ex	0	0	3	8	5	16	0.3416667
C	http://littletutorials.com/2008/04/27/exceptional-java-	0	1	0	5	10	16	0.4583333
C	http://forum.springsource.org/showthread.php?t=63	5	4	6	1	0	16	0.2583333
C	http://en.wikibooks.org/wiki/Java_Programming/Thro	1	0	1	3	9	14	0.4285714
C	http://www.javaworld.com/javaworld/javaqa/2003-	0	0	7	7	2	16	0.3583333
C	http://www-128.ibm.com/developerworks/java/libra	5	7	4	0	0	16	0.3083333
C	http://www.tutorialspoint.com/java/java_exceptions	0	0	2	4	13	19	0.497076
C	http://java.sun.com/docs/books/tutorial/essential/ex(	1	0	0	2	15	18	0.6928105
C	http://www.infoq.com/resource/presentations/effe(	2	4	5	3	2	16	0.175
C	http://www.hietavirta.net/blog/item/2007/06/do-not-	16	0	0	0	0	16	1
C	http://www.manageability.org/blog/stuff/exceptiona	1	0	6	4	7	18	0.2745098
C	http://www.javaworld.com/javaworld/jw-07-2005/j	1	3	2	7	3	16	0.2333333
C	http://www.roseindia.net/java/java-exception/index	0	0	2	2	12	16	0.5666667
C	http://www.javamex.com/tutorials/exceptions/exce	0	0	2	7	8	17	0.3676471
C	http://googletesting.blogspot.com/2009/09/checked-	9	3	3	2	1	18	0.2810458
C	http://java.sun.com/docs/books/tutorial/essential/ind	1	2	7	5	1	16	0.2666667
C	http://www.ibm.com/developerworks/java/library/j-j	1	0	7	8	2	18	0.3267974
C	http://java.sun.com/docs/books/tutorial/essential/ex(	0	0	8	5	5	18	0.3137255
C	http://blog.objectmentor.com/articles/2009/07/13/en(	13	2	1	0	0	16	0.6583333
C	http://www-128.ibm.com/developerworks/java/libra	0	2	6	6	2	16	0.2666667
C	http://dev2dev.bea.com/pub/a/2006/12/incremental-	13	0	1	2	0	16	0.6583333
C	http://blog.robwhelan.com/2008/10/05/an-approach	0	0	4	7	5	16	0.3083333
C	http://www.c2.com/cgi/wiki?CheckedException	0	1	9	4	2	16	0.3583333

Question	Resource URL	1	2	3	4	5	# of raters	P_i
C	http://www.onjava.com/pub/a/onjava/2006/01/11/e	1	2	7	6	2	18	0.248366
C	http://www.artima.com/intv/solid3.html	2	3	7	4	0	16	0.2583333
C	http://www.c2.com/cgi/wiki?RefineExceptions	0	7	7	2	0	16	0.3583333
C	http://www.artima.com/intv/solid.html	3	5	7	2	1	18	0.2287582
C	http://www.c2.com/cgi/wiki?HomogenizeException	1	3	5	5	2	16	0.2
C	http://www.artima.com/intv/handcuffs.html	2	2	8	5	1	18	0.2614379
C	http://www.c2.com/cgi/wiki?CheckedExceptionsAr	0	1	8	7	2	18	0.3267974
C	http://www.c2.com/cgi/wiki?ExceptionTunneling	0	3	7	4	2	16	0.2583333
C	http://jug.org.ua/wiki/display/JavaAlmanac/Handling	12	3	0	1	0	16	0.575
C	http://littletutorials.com/2008/05/23/exceptional-java-	0	0	8	3	5	16	0.3416667
C	http://davidvancouvering.blogspot.com/2008/09/cur	2	5	1	7	1	16	0.2666667
C	http://weblogs.goshaky.com/weblogs/alexkli/entry/e	17	0	0	0	0	17	1
C	http://blog.thinkrelevance.com/2008/2/4/layering-an	8	5	2	1	0	16	0.325
C	http://www.odi.ch/prog/design/newbies.php	6	4	6	0	0	16	0.3
C	http://www.blueskyline.com/ErrorPatterns/A2-Long	6	1	4	5	2	18	0.2091503
C	http://www.onjava.com/pub/a/onjava/2003/11/19/e	2	0	5	6	3	16	0.2416667
C	http://softarc.blogspot.com/2007/06/exception-hand	2	3	5	4	2	16	0.175
C	http://blogs.concedere.net:8080/blog/discipline/soft	16	0	0	0	0	16	1
C	http://www.onjava.com/pub/a/onjava/2006/01/11/e	1	2	6	5	2	16	0.225
C	http://dev2dev.bea.com/pub/a/2006/11/effective-ex	14	1	3	0	0	18	0.6143791
C	http://www.javaworld.com/javaworld/javatips/jw-ja	1	3	9	2	1	16	0.3333333
C	http://www.mindview.net/Etc/Discussions/Checked	0	0	11	5	4	20	0.3736842
C	http://wuhrr.wordpress.com/2007/11/22/java-excep	0	2	8	2	4	16	0.3
C	http://tutorials.jenkov.com/java-exception-handling/e	2	0	7	5	6	20	0.2473684
C	http://www.javapractices.com/topic/TopicAction.do	0	0	5	8	5	18	0.3137255
C	http://pages.cs.wisc.edu/~hasti/cs368/JavaTutorial	0	0	0	1	11	12	0.8333333
C	http://www.onjava.com/pub/a/onjava/2003/11/19/e	1	0	5	5	7	18	0.2679739
C	http://www.javaworld.com/javaworld/jw-11-2007/j	1	2	7	2	8	20	0.2684211
C	http://www.jroller.com/page/hackingarchitect?entry	1	1	7	7	2	18	0.2810458
C	http://www.javaworld.com/javaworld/jw-07-1998/j	0	0	3	6	10	19	0.3684211
C	http://www.oracle.com/technology/pub/articles/dev	14	1	2	1	0	18	0.6013072
C	http://www.wikijava.org/wiki/10_best_practices_w	0	1	1	7	9	18	0.372549
C	http://jakarta.apache.org/commons/lang/api/org/apa	16	0	0	0	0	16	1
C	http://www.javaworld.com/jw-07-1998/jw-07-exce	0	0	1	5	7	13	0.3974359
C	http://www-106.ibm.com/developerworks/java/libra	0	0	5	6	5	16	0.2916667
C	http://www.jenkov.com/training/trails.tmpl	14	2	0	0	0	16	0.7666667
C	http://tutorials.jenkov.com/java-exception-handling/e	0	1	5	6	6	18	0.2614379
C	http://www.javaworld.com/javaworld/jw-10-2003/j	1	4	5	4	2	16	0.1916667
C	http://nat.truemesh.com/archives/000698.html	6	3	5	1	1	16	0.2333333
C	http://www.cajoon.com/	16	0	0	0	0	16	1
C	http://tutorials.jenkov.com/java-exception-handling/in	1	0	2	4	13	20	0.4473684
C	http://accu.org/index.php/journals/236	0	1	7	5	3	16	0.2833333
C	http://www.javaspecialists.eu/archive/Issue162.htr	0	0	3	5	10	18	0.379085
C	http://today.java.net/pub/a/today/2003/12/04/except	0	1	4	5	10	20	0.3210526
C	http://www.oracle.com/technetwork/articles/java/ja	0	2	1	6	3	12	0.2878788
C	http://www.infoq.com/news/2008/01/presentation-	10	6	0	0	0	16	0.5
C	http://www.javaworld.com/javaworld/jw-03-2002/j	0	2	4	4	6	16	0.2333333
C	http://jakarta.apache.org/commons/lang/api/org/apa	16	0	0	0	0	16	1
C	http://www.codingthearchitecture.com/2008/01/14/	4	6	6	0	0	16	0.3
C	http://www.javaworld.com/javaworld/jw-11-2007/j	0	1	8	4	3	16	0.3083333
C	http://developingdeveloper.wordpress.com/2008/02	3	1	10	2	0	16	0.4083333
C	http://www.javaworld.com/javaworld/jw-08-2001/j	0	0	6	9	3	18	0.3529412
C	http://www-128.ibm.com/developerworks/library/j-	16	0	0	0	0	16	1
C	http://www.mortench.net/blog/2006/08/08/dos-and-	16	0	0	0	0	16	1
C	http://wiki.java.net/bin/view/Javapedia/Exception	7	2	2	2	3	16	0.225
C	http://www-06.ibm.com/jp/developerworks/java/04	16	0	0	0	0	16	1

Question	Resource URL	1	2	3	4	5	# of raters	P_i
D	http://www.superliminal.com/sources/JarLoader.jav	0	0	3	1	6	10	0.4
D	http://wikis.sun.com/display/code/Home#j2ee	10	0	0	0	0	10	1
D	http://www.java-examples.com/	0	0	2	6	10	18	0.3986928
D	http://codingbat.com/example.html	1	0	0	0	16	17	0.8823529
D	http://javaalmanac.com/	0	3	3	3	8	17	0.2720588
D	http://www.javacodeexamples.com/	0	0	2	6	6	14	0.3406593
D	http://www.exampledepot.com/egs/java.net/Post.ht	0	1	1	4	11	17	0.4485294
D	http://www.bejug.org/confluenceBeJUG/display/Be	0	1	2	0	7	10	0.4888889
D	http://snippets.dzone.com/tag/java	0	1	3	2	4	10	0.2222222
D	http://www.wickedcooljava.com/downloads.jsp	1	0	1	4	4	10	0.2666667
D	http://www.exampledepot.com/	0	1	6	5	6	18	0.2614379
D	http://www.javareference.com/	7	1	1	1	0	10	0.4666667
D	http://www.idevelopment.info/data/Programming/jav	13	0	0	0	0	13	1
D	http://www.pscode.com/vb/default.asp?lngWId=2#	3	1	2	3	1	10	0.1555556
D	http://littletutorials.com/2008/03/14/console-applicati	0	0	2	3	5	10	0.3111111
D	http://java.sun.com/docs/books/tutorial/uiswing/com	0	0	2	1	7	10	0.4888889
D	http://sujitpal.blogspot.com/	3	0	2	0	5	10	0.3111111
D	http://www.javacodegeeks.com/2012/01/java-7-pro	0	0	1	3	10	14	0.5274725
D	http://www.example-code.com/	4	4	4	2	2	16	0.1666667
D	http://www.roseindia.net/java/	2	1	6	2	7	18	0.248366
D	http://javafaq.nu/modules.php?name=Encyclopedia	10	0	0	0	0	10	1
D	http://javaalmanac.com/egs/java.lang/pkg.html	1	2	0	2	5	10	0.2666667
D	http://java-source.net/	2	0	4	4	0	10	0.2888889
D	https://filthyrichclients.dev.java.net/	9	1	0	0	0	10	0.8
D	http://www.kodejava.org/	1	0	0	5	12	18	0.496732
D	http://www.jexamples.com/	2	2	7	2	4	17	0.2205882
D	http://www.bigbold.com/snippets/	6	2	4	3	5	20	0.1842105
D	http://www.google.com/search?hl=en	5	3	2	0	0	10	0.3111111
D	http://netbeans.dzone.com/news/simple-mysql-integ	0	1	5	2	2	10	0.2666667
D	http://en.wikipedia.org/wiki/Category:Articles_with_	3	2	2	3	2	12	0.1363636
D	http://64.18.163.122/rgagnon/howto.html	5	0	2	1	2	10	0.2666667
D	http://www.uize.com/javascript-examples.html	2	2	2	2	2	10	0.1111111
D	http://kickjava.com/src/	6	2	3	0	1	12	0.2878788
D	http://lombok.demon.co.uk/tapestry5Demo/	3	1	4	0	2	10	0.2222222
D	http://labs.oreilly.com/code/	10	0	0	0	0	10	1
D	http://www.javapractices.com/index.cjp	0	2	1	1	6	10	0.3555556
D	http://www.makeuseof.com/tag/top-10-professiona	4	3	6	2	1	16	0.2083333
D	http://www.codefetch.com/	10	0	0	0	0	10	1
D	http://www.myhomepageindia.com/index.php/2009/	5	1	2	1	1	10	0.2444444
D	http://www.leepoint.net/notes-java/index.html	1	0	5	6	7	19	0.2690058
D	http://www.exampledepot.com/egs/index.html	0	1	5	1	3	10	0.2888889
D	http://www.oracle.com/technology/sample_code/te	10	0	0	0	0	10	1
D	http://forums.sun.com/thread.jspa?threadID=538656	10	0	0	0	0	10	1
D	http://www2.cs.uic.edu/~sloan/CLASSES/java/	0	0	2	6	10	18	0.3986928
D	http://oreilly.com/catalog/javanut/examples/	1	0	1	7	10	19	0.3859649
D	http://kickjava.com/	3	3	4	4	4	18	0.1568627
D	http://www.java2s.com/Code/Java/CatalogJava.htr	0	3	1	5	7	16	0.2833333
D	http://www.java2s.com/	4	4	4	3	5	20	0.1631579
D	http://pleac.sourceforge.net/	6	1	3	1	0	11	0.3272727
D	http://www.javabat.com/	0	3	2	3	2	10	0.1777778
D	http://www.movesinstitute.org/~mcgredo/mv3500/r	10	0	0	0	0	10	1
D	http://www.techfaq360.com/tutorial/hibernate.jsp	7	3	3	3	1	17	0.2205882
D	http://www.java2s.com/Code/Java/CatalogJava.htr	0	3	1	5	7	16	0.2833333
D	http://www.java2s.com/	4	4	4	3	5	20	0.1631579
D	http://pleac.sourceforge.net/	6	1	3	1	0	11	0.3272727
D	http://www.javabat.com/	0	3	2	3	2	10	0.1777778

Question	Resource URL	1	2	3	4	5	# of raters	P_i
D	http://www.movesinstitute.org/~mcgredo/mv3500/r	10	0	0	0	0	10	1
D	http://www.techfaq360.com/tutorial/hibernate.jsp	7	3	3	3	1	17	0.2205882
D	http://www.java-tips.org/index.html	1	2	3	1	3	10	0.1555556
D	http://wikis.sun.com/display/code/Home	10	0	0	0	0	10	1
D	http://www.eclipse.org/swt/snippets/	2	0	3	4	1	10	0.2222222
D	http://nicolaslecoz.blogspot.com/2007/05/how-to-fir	9	1	0	0	0	10	0.8
D	http://www.makeuseof.com/tag/top-5-websites-for	4	0	5	4	4	17	0.2058824
D	http://www.javapractices.com/TableOfContents.cjp	1	0	2	2	5	10	0.2666667
D	http://sites.google.com/a/pintailconsultingllc.com/jav	2	4	1	0	3	10	0.2222222
D	http://www.zvon.org/xxl/XPathTutorial/General/exa	12	2	2	0	0	16	0.5666667
D	http://snippets.dzone.com/	2	3	2	2	4	13	0.1538462
D	http://www.javadb.com/	3	1	3	1	5	13	0.2051282
D	http://www.springbyexample.org/	5	4	4	1	3	17	0.1838235
D	http://examples.oreilly.com/jswing2/code/	0	1	0	1	8	10	0.6222222
D	http://java.sun.com/docs/books/tutorial/uiswing/com	10	0	0	0	0	10	1
D	http://snobol.cs.berkeley.edu/prospector/	10	0	0	0	0	10	1
D	http://www.javapassion.com/	4	2	2	2	0	10	0.2
D	http://snipplr.com/	3	2	3	2	1	11	0.1454545
D	http://ajaxtags.sourceforge.net/	3	1	6	0	0	10	0.4
E	http://www.roseindia.net/java/jdk6/introduction-coll	0	3	2	4	8	17	0.2794118
E	http://trove4j.sourceforge.net/	5	5	7	1	1	19	0.2397661
E	http://people.csail.mit.edu/milch/blog/apidocs/commo	2	1	7	2	2	14	0.2637363
E	http://www.ibm.com/developerworks/java/library/j-	1	1	5	3	2	12	0.2121212
E	http://bitworking.org/news/358/restful-json	10	3	2	0	0	15	0.4666667
E	http://www.ociweb.com/jnb/jnbApr2008.html	1	2	4	2	3	12	0.1666667
E	http://stackoverflow.com/questions/629804?sort=o	1	0	5	6	1	13	0.3205128
E	http://github.com/jorgeortiz85/scala-javautils	4	6	3	1	0	14	0.2637363
E	http://publicobject.com/glazedlists/	13	1	3	0	1	18	0.5294118
E	http://www.xylax.net/hibernate/index.html	13	1	0	0	0	14	0.8571429
E	http://www.exampledepot.com/egs/java.util/coll_Ma	0	2	7	2	1	12	0.3484848
E	http://www.odi.ch/prog/design/newbies.php	4	6	3	0	0	13	0.3076923
E	http://rickyclarkson.blogspot.com/2007/09/point-free	5	3	3	1	0	12	0.2424242
E	http://www.infoq.com/news/2007/10/collections-ap	0	0	4	5	4	13	0.2820513
E	http://josql.sourceforge.net/index.html	7	1	3	0	1	12	0.3636364
E	http://blog.jayway.com/2009/10/22/google-collection	0	4	5	3	1	13	0.2435897
E	http://codemunchies.com/2009/10/diving-into-the-go	3	3	5	2	1	14	0.1868132
E	http://www.javamex.com/tutorials/collections/	0	1	1	4	12	18	0.4705882
E	http://www.youtube.com/watch?v=ZeO_J2OcHYM	5	3	2	3	0	13	0.2179487
E	http://weblogs.java.net/blog/jhook/archive/2006/12/(	3	5	3	1	1	13	0.2051282
E	http://www.kellyrob99.com/blog/2010/05/15/achiev	0	1	5	7	1	14	0.3406593
E	http://www.hazelcast.com/	11	4	0	1	1	17	0.4485294
E	http://www.infoq.com/articles/in-depth-look-clojure-	5	6	2	0	0	13	0.3333333
E	http://tutorials.jenkov.com/java-collections/index.htm	0	0	0	5	9	14	0.5054945
E	http://www.ibm.com/developerworks/java/library/j-	0	0	3	3	9	15	0.4
E	http://xircles.codehaus.org/projects/quaere	12	1	1	0	0	14	0.7252747
E	http://smallwig.blogspot.com/2007/12/why-does-se	4	5	3	1	1	14	0.2087912
E	http://www.rgagnon.com/javadetails/java-0633.htm	1	2	3	5	2	13	0.1923077
E	http://www.ibm.com/developerworks/java/library/j-	0	1	3	4	7	15	0.2857143
E	http://joda-primitives.sourceforge.net/	4	0	4	4	2	14	0.2087912
E	http://www.javaworld.com/javaworld/jw-11-2004/j	6	6	1	0	0	13	0.3846154
E	http://www.infoq.com/news/2010/01/google_collec	0	4	8	4	1	17	0.2941176
E	http://code.google.com/p/google-collections/	13	2	3	2	0	20	0.4368421
E	http://marxsoftware.blogspot.com/	5	7	1	0	0	13	0.3974359
E	http://docs.oracle.com/javase/tutorial/essential/conc	0	0	4	3	3	10	0.2666667
E	http://labs.carrotsearch.com/hppc.html	0	2	5	3	1	11	0.2545455
E	http://sourceforge.net/projects/high-scale-lib	3	8	2	1	0	14	0.3516484

Question	Resource URL	1	2	3	4	5	# of raters	P_i
E	http://codemunchies.com/2009/10/beautiful-code-w	4	3	2	2	2	13	0.1538462
E	http://code.google.com/p/concurrentlinkedhashmap/	5	1	4	2	2	14	0.1978022
E	http://codemunchies.com/2009/11/functional-java-fi	2	3	4	4	0	13	0.2051282
E	http://fastutil.dsi.unimi.it/	0	3	7	2	5	17	0.2573529
E	http://jtheque.developpez.com/	10	3	1	0	0	14	0.5274725
E	http://crazybob.org/2008/01/in-hot-seat.html	7	4	1	0	0	12	0.4090909
E	http://people.cs.aau.dk/~torp/Teaching/E01/Oop/har	0	0	0	0	11	11	1
E	http://javolution.org/	9	5	5	0	0	19	0.3274854
E	http://www.developer.com/java/other/article.php/37	5	4	2	1	1	13	0.2179487
E	http://weblogs.java.net/blog/van_riper/archive/2008	1	5	4	3	0	13	0.2435897
E	http://www.javabeginner.com/java-collections-fram	1	0	0	0	11	12	0.8333333
E	http://gleichmann.wordpress.com/2008/01/13/buildi	0	1	3	3	6	13	0.2692308
E	http://en.wikipedia.org/wiki/Java_collections_frame	0	0	0	2	16	18	0.7908497
E	http://code.google.com/p/guava-libraries/	4	6	4	0	0	14	0.2967033
E	http://www.jot.fm/issues/issue_2004_09/column1/	2	3	5	3	1	14	0.1868132
E	http://publicobject.com/2007/09/series-recap-coding	4	4	3	4	1	16	0.175
E	http://www.theserverside.com/tt/blogs/showblog.ts	5	7	1	0	0	13	0.3974359
E	http://www.caughtbyjava.com/new-java-6-collectio	9	1	3	0	0	13	0.5
E	https://www.sdn.sap.com/irj/sdn/weblogs?blog=/p	5	2	3	3	1	14	0.1868132
E	http://larvalabs.com/collections/	3	1	6	1	3	14	0.2307692
E	http://java.sun.com/j2se/1.4.2/docs/guide/collections	0	0	0	6	5	11	0.4545455
E	http://codemunchies.com/2009/11/preconditions-mu	3	2	3	3	2	13	0.1410256
E	http://www.recursionsw.com/Products/jgl.html	13	0	0	0	0	13	1
E	http://commons.apache.org/primitives/	8	3	1	2	0	14	0.3516484
E	http://www.onjava.com/pub/a/onjava/2002/06/12/tr	3	3	4	2	0	12	0.1969697
E	http://www.artima.com/intv/bloch.html	6	4	4	1	0	15	0.2571429
E	http://blogs.azulsystems.com/cliff/2008/01/adding-t	13	0	1	0	0	14	0.8571429
E	http://github.com/scalaj/scalaj-collection	2	1	9	1	1	14	0.4065934
E	http://www.fromdev.com/2008/05/java-collections-	0	0	4	3	11	18	0.4183007
E	http://pcj.sourceforge.net/	0	4	2	5	6	17	0.2352941
E	http://docs.oracle.com/javase/1.4.2/docs/api/java/ut	0	0	2	4	10	16	0.4333333
E	http://java.sun.com/developer/onlineTraining/collecti	1	0	0	1	15	17	0.7720588
E	http://locut.us/SimpleBloomFilter/	13	0	0	0	0	13	1
E	http://jnb.ociweb.com/jnb/jnbApr2010.html	3	6	3	1	1	14	0.2307692
E	http://www.tutorialspoint.com/java/java_collections	0	0	0	1	15	16	0.875
E	http://java.sun.com/docs/books/tutorial/collections/ir	0	0	1	4	8	13	0.4358974
E	http://www.javamex.com/tutorials/collections/using	0	0	1	2	7	10	0.4888889
E	http://gee.cs.oswego.edu/cgi-bin/viewcvs.cgi/jsr16	6	2	5	0	0	13	0.3333333
E	http://tutorials.jenkov.com/	2	2	5	1	4	14	0.1978022
E	http://www.javaworld.com/javaworld/jw-10-2004/j	4	2	3	3	1	13	0.1666667
E	http://tobega.blogspot.com/2008/05/beautiful-enums	6	3	4	1	0	14	0.2637363
E	http://docs.oracle.com/javase/tutorial/collections/intr	0	0	1	1	8	10	0.6222222
E	http://users.mafr.de/~matthias/articles/google-collec	1	2	5	7	3	18	0.2287582
E	http://jakarta.apache.org/commons/collections/	2	0	3	4	6	15	0.2380952
E	http://www.onjava.com/lpt/a/3286	3	3	3	3	1	13	0.1538462
E	http://jakarta.apache.org/commons/jxpath/	5	5	3	1	0	14	0.2527473
E	http://code.google.com/p/lambdaj/	10	3	6	0	0	19	0.3684211
E	http://www.space4j.org/	5	5	2	1	0	13	0.2692308
E	http://www.angelikalanger.com/GenericsFAQ/Java	7	4	3	0	0	14	0.3296703
E	http://spin.atomicobject.com/2010/02/23/better-java-	2	5	3	3	1	14	0.1868132
E	http://java.sun.com/javase/6/docs/technotes/guides	0	0	1	1	9	11	0.6545455
E	http://snehaprashant.blogspot.com/2008/10/quick-re	0	0	0	1	13	14	0.8571429
E	http://commons.apache.org/collections/	1	2	8	2	3	16	0.275
E	http://code.google.com/p/pcollections/	0	4	3	3	1	11	0.2181818
E	http://binterberg.blogspot.com/2009/09/introductio	1	4	5	3	2	15	0.1904762
E	http://www.javalobby.org/articles/google-collection	6	4	3	2	2	17	0.1911765

Question	Resource URL	1	2	3	4	5	# of raters	P_i
E	http://java.sun.com/docs/books/tutorial/collections/ir	0	0	4	5	10	19	0.3567251
E	http://today.java.net/pub/a/today/2006/11/07/nuance	5	3	3	1	0	12	0.2424242
E	http://docs.oracle.com/javase/1.5.0/docs/api/java/ut	1	0	1	5	3	10	0.2888889
E	http://www.op4j.org/	6	2	3	6	1	18	0.2222222
E	http://jaggregate.sourceforge.net/	4	3	6	4	0	17	0.2205882
E	http://docs.oracle.com/javase/tutorial/collections/ind	0	0	4	7	8	19	0.3216374
F	http://weblogs.java.net/blog/claudio/archive/nb-reus	4	2	0	3	3	12	0.1969697
F	http://java.sun.com/developer/onlineTraining/collecti	2	3	5	3	0	13	0.2179487
F	http://www.apl.jhu.edu/~hall/java/Swing-Tutorial/Sv	1	2	0	6	5	14	0.2857143
F	http://today.java.net/pub/a/today/2007/05/17/uispec	1	5	4	1	2	13	0.2179487
F	http://java.sun.com/docs/books/tutorial/uiswing/com	2	1	5	4	3	15	0.1904762
F	http://java.sun.com/docs/books/tutorial/uiswing/misc	13	0	0	0	0	13	1
F	http://www.ibm.com/developerworks/views/web/li	7	4	1	1	0	13	0.3461538
F	http://java.sun.com/products/jlf/at/book/Idioms5.html	5	3	5	0	0	13	0.2948718
F	http://java.sun.com/docs/books/tutorial/uiswing/look	0	2	2	6	5	15	0.2571429
F	http://java.sun.com/docs/books/tutorial/uiswing/look	1	2	5	3	2	13	0.1923077
F	http://www.apl.jhu.edu/~hall/java/Swing-Tutorial/	0	0	0	4	10	14	0.5604396
F	http://netbeans.org/kb/docs/java/gui-binding.html	4	1	5	3	0	13	0.2435897
F	http://docs.oracle.com/javase/tutorial/uiswing/comp	0	0	0	6	11	17	0.5147059
F	http://junit.sourceforge.net/doc/testinfected/testing.	5	2	5	2	0	14	0.2417582
F	http://zetcode.com/tutorials/javaswingtutorial/	0	0	0	1	18	19	0.8947368
F	http://www.java2s.com/Code/Java/Swing-Compon	0	1	2	6	4	13	0.2820513
F	http://java.sun.com/products/jfc/tsc/articles/threads	2	1	2	4	4	13	0.1794872
F	http://www.cise.ufl.edu/~amyles/tcpchat/	0	0	1	6	7	14	0.3956044
F	http://java.sun.com/developer/technicalArticles/java	2	2	5	4	3	16	0.175
F	http://www.informit.com/articles/article.aspx?p=10	7	4	2	0	0	13	0.3589744
F	http://java.sun.com/j2se/1.5.0/docs/api/javax/swing	1	2	3	7	0	13	0.3205128
F	http://www.netbeans.org/kb/articles/matisse.html	15	0	0	0	0	15	1
F	http://www.swingwiki.org/	1	0	7	4	6	18	0.2745098
F	http://www.jroller.com/gfx/date/20050214	3	1	7	1	2	14	0.2747253
F	http://www.netbeans.org/kb/articles/gui-functionali	2	1	3	2	6	14	0.2197802
F	http://weblogs.java.net/blog/tpavek/archive/2006/02	12	1	0	0	0	13	0.8461538
F	http://java.sun.com/docs/books/tutorial/uiswing/pair	0	1	3	7	2	13	0.3205128
F	http://java.sun.com/docs/books/tutorial/uiswing/com	0	0	1	7	9	17	0.4191176
F	http://www.java2s.com/Tutorial/Java/0240__Swing	0	0	0	2	12	14	0.7362637
F	http://www.tutorialized.com/tutorial/SWT-Tutorial/7	13	0	0	0	0	13	1
F	https://openjfx.dev.java.net/JavaFX_Programming_L	14	0	0	0	0	14	1
F	http://www.guj.com.br/java.tutorial.artigo.147.1.guj	3	1	4	4	1	13	0.1923077
F	http://www.netbeans.org/kb/60/java/gui-db.html	17	0	0	0	0	17	1
F	http://www.daltonfilho.com/articles/swingx/	4	2	2	4	1	13	0.1794872
F	http://www.exampledepot.com/egs/index.html	1	4	5	2	1	13	0.2179487
F	http://www.ibm.com/developerworks/java/library/j-	3	3	3	5	0	14	0.2087912
F	http://java.sun.com/developer/technicalArticles/J2S	8	3	2	3	0	16	0.2916667
F	https://appframework.dev.java.net/intro/index.html	15	0	0	0	0	15	1
F	http://www.netbeans.org/kb/docs/java/gui-db.html	13	0	0	0	0	13	1
F	http://www.swingwiki.org/table_of_contents	0	0	2	4	9	15	0.4095238
F	http://java.sun.com/docs/books/tutorial/uiswing/layo	0	0	5	5	3	13	0.2948718
F	http://www.netbeans.org/kb/60/java/quickstart-gui.	0	0	2	8	7	17	0.3676471
F	http://today.java.net/pub/a/today/2006/03/30/introdu	2	4	2	2	3	13	0.1538462
F	http://www.netbeans.org/kb/trails/matisse.html	0	2	3	4	8	17	0.2794118
F	http://cs.nyu.edu/~yap/classes/visual/03s/lect/l7/	0	0	0	0	17	17	1
F	http://www.javabeginner.com/java-swing/java-swi	0	0	0	2	17	19	0.8011696
F	http://www.swingwiki.org/table_of_contents#best_	0	0	2	3	8	13	0.4102564
F	http://java.sun.com/products/jfc/tsc/articles/painting	1	0	2	6	4	13	0.2820513
F	http://www.anyang-window.com.cn/construction-	6	5	1	0	0	12	0.3787879
F	http://java.sun.com/docs/books/tutorial/uiswing/lear	0	0	2	4	8	14	0.3846154

Question	Resource URL	1	2	3	4	5	# of raters	P_i
F	http://java.sun.com/docs/books/tutorial/uisw ing/con	0	3	4	5	3	15	0.2095238
F	http://w w w .javalobby.org/articles/jtable/	2	1	2	8	4	17	0.2647059
F	http://w eblogs.java.net/blog/kirillcool/archive/2005/0	2	2	4	3	2	13	0.1538462
F	http://w w w .jroller.com/santhosh/date/20050610#jtr	1	2	5	5	1	14	0.2307692
F	http://w w w .javaw orld.com/javaw orld/jw -07-2007/j	3	4	7	1	0	15	0.2857143
F	http://today.java.net/pub/a/today/2007/02/22/how -to	0	0	2	7	10	19	0.3918129
F	http://java.sun.com/docs/books/tutorial/uisw ing/exa	0	0	1	1	14	16	0.7583333
F	http://developerlife.com/tutorials/?p=15	3	1	4	3	2	13	0.1666667
F	http://java.sun.com/docs/books/tutorial/uisw ing/com	0	2	3	3	5	13	0.2179487
F	http://java.sun.com/docs/books/tutorial/ui/index.html	0	1	4	5	3	13	0.2435897
F	http://java.sun.com/products/jfc/tsc/articles/actions/	1	0	4	4	4	13	0.2307692
F	http://java.sun.com/docs/books/tutorial/extra/fullscr	3	2	8	0	1	14	0.3516484
F	http://java.sun.com/docs/books/tutorial/uisw ing/com	0	2	1	3	8	14	0.3516484
F	http://w w w .javatutorialhub.com/java-sw ing-gui.htm	0	0	0	3	13	16	0.675
F	http://java.sun.com/products/jfc/tsc/articles/cardpar	2	3	7	1	0	13	0.3205128
F	http://java.sun.com/docs/books/tutorial/2d/index.htm	1	7	3	3	0	14	0.2967033
F	http://homepage.mac.com/svc/	12	1	0	0	0	13	0.8461538
F	http://w eblogs.java.net/blog/g_s_m/archive/2007/09	0	0	4	6	3	13	0.3076923
F	http://w w w .javabeginner.com/java-sw ing-tutorial.h	0	0	0	3	11	14	0.6373626
F	http://w w w .roseindia.net/java/example/java/sw ing/	1	0	1	2	12	16	0.5583333
F	http://java.sun.com/docs/books/tutorial/uisw ing/exa	0	0	1	2	10	13	0.5897436
F	http://w w w .javafree.org/content/view .jf?idContent=	4	5	2	1	1	13	0.2179487
F	http://java.sun.com/docs/books/tutorial/uisw ing/dnd	4	2	3	2	2	13	0.1538462
F	http://java.sun.com/docs/books/tutorial/index.html	1	2	8	2	3	16	0.275
F	http://w w w .netbeans.org/kb/60/java/gui-saf.html	13	0	0	0	0	13	1
F	http://java.sun.com/docs/books/tutorial/uisw ing/	0	0	0	7	12	19	0.5087719
F	http://today.java.net/pub/a/today/2006/02/21/building	0	3	5	5	2	15	0.2285714
F	http://netbeans.org/kb/docs/java/quickstart-gui.html	0	0	1	5	5	11	0.3636364
F	http://w w w .javasw ingtutorial.com/	0	0	2	3	6	11	0.3454545
F	http://java.sun.com/products/jfc/tsc/articles/sw ing2	1	3	2	3	4	13	0.1666667
F	http://java.sun.com/docs/books/tutorial/uisw ing/com	1	3	3	4	2	13	0.1666667
F	http://w w w .ociw eb.com/jnb/jnbOct2005.html	3	4	3	3	0	13	0.1923077
F	http://w w w .new t.com/java/sw ing.html	0	0	1	8	5	14	0.4175824
F	http://w w w .netbeans.org/community/magazine/htm	1	4	2	5	1	13	0.2179487
F	http://java.sun.com/docs/books/tutorial/uisw ing/exa	0	0	0	2	11	13	0.7179487
F	http://w w w .jgoodies.com/	5	7	5	0	0	17	0.3014706
F	http://java.sun.com/developer/technicalArticles/java	2	0	2	6	6	16	0.2666667
F	http://w w w .javasw ing.net/	8	0	3	1	1	13	0.3974359
F	http://java.sun.com/docs/books/tutorial/uisw ing/look	2	1	3	5	3	14	0.1868132
F	http://today.java.net/pub/a/today/2004/01/05/sw ing.	0	3	6	3	1	13	0.2692308
F	http://w w w .javalobby.org/articles/miglayout/	10	4	1	0	0	15	0.4857143
F	http://jug.org.ua/w iki/display/JavaAlmanac/Inserting	9	4	0	0	0	13	0.5384615
F	http://java.sun.com/j2se/1.5.0/docs/api/javax/sw ing	6	5	2	1	0	14	0.2857143
F	http://java.sun.com/docs/books/tutorial/uisw ing/pair	12	0	1	0	0	13	0.8461538
F	http://java.sun.com/docs/books/tutorial/uisw ing/TOC	0	0	1	1	17	19	0.7953216
F	http://docs.oracle.com/javase/tutorial/uisw ing/TOC.l	0	0	0	2	10	12	0.6969697
F	http://w iki.netbeans.org/NBDemoFlickr	5	3	3	2	0	13	0.2179487
F	http://w w w .guj.com.br/java.tutorial.artigo.140.1.guj	3	0	2	7	1	13	0.3205128
F	http://w w w .javafaq.nu/java-allbooks.html	3	2	3	1	5	14	0.1868132

.

APPENDIX B

ENTRY QUESTIONNAIRE FOR EXPERIMENTS 1 & 2 (JAVA PROGRAMMING STUDY)

The purpose of this research study is to improve methods for locating expert users in social bookmarking systems. For this purpose, we will give participants a series of search tasks for finding information on topics related to the Java programming language. Participants will enter queries into an experimental system, and then judge how relevant the returned web resources are to the given topics. Participants will be asked to complete one session lasting approximately 2 hours. The session includes training on the experimental system, answering the pre-questionnaire below, and performing the actual experiment.

Prior to the research experiment, please provide answers to following questions.

1. What is your major of study?
   _____ Computer Science
   _____ Information Science
   _____ Other (Please specify ……)

2. How would you rate your knowledge of the **Java programming language**?
   _____ Expert
   _____ Intermediate
   _____ Novice

_____ I have no knowledge of Java.

3. How did you learn Java? (please check all that apply)
   _____ Self-study
   _____ Programming courses that used Java
   _____ On-the-job training
   _____ Other (Please specify …….)

4. How long have you used Java?
   _____ less than 1 year
   _____ 1-3 years
   _____ 4 years or more

5. In how many projects have you used Java as a development tool?
   _____ 3 or less
   _____ 4 – 6 projects
   _____ 7 – 9 projects
   _____ 10 or more

6. Describe those projects:
   _____ All academic assignments
   _____ Some academic assignments and some non-academic projects
   _____ All non-academic projects
   _____ Other (Please specify ……..)

7. Other than Java, what programming languages do you use?

……

……

8. How many **hours per day** do you use a computer?

_____ less than 3 hours

_____ between 3 and 6 hours

_____ between 6 and 9 hours

_____ between 9 and 12 hours

_____ more than 12 hours

9. How many **times per day** do you search for information on the web?

_____ 5 or less times per day

_____ 6 – 10 times per day

_____ 11- 15 times per day

_____ 15 or more times per day

10. When searching for information on the web, how many terms do you use *on average* in your search queries?

_____ 1 term

_____ 2 terms

_____ 3 terms

_____ 4 terms

_____ 5 or more terms

11. How would you describe your ability to find information on the web using a search engine?

_____ I always find what I want

_____ Most of the time I find what I want

_____ Half of the time I find what I want

_____ Rarely do I find what I want

_____ I never find what I want

APPENDIX C

ENTRY QUESTIONNAIRE FOR EXPERIMENT 3 (CLASSIFICATION STUDY)

The purpose of this research study is to improve methods for locating expert users in social bookmarking systems. For this purpose, we ask participants to judge how relevant a series of terms represent the topics of web resources. Participants will be asked to complete one session lasting approximately 1-2 hours. The session includes training on the experimental system, answering the pre-questionnaire below, and performing the actual experiment.

Prior to the research experiment, please provide answers to following questions.

1. I am a… (Please check all that apply)
   _____ librarian at _____
   _____ MLIS degree holder
   _____ MLIS student
   _____ PhD Student in LIS

2. If you are a librarian, what is your specialty (i.e., major tasks) in your library?

3. If you are a graduate student, what is your specialty (track or research interest)?

4. If you are a graduate student in LIS, please check all of the course(s) you have taken.

   _____ Organizing & Retrieving Information (LIS2005)

   _____ Introduction to Cataloging and Classification (LIS2405)

   _____ Advanced Cataloging and Classification (LIS2406)

   _____ Metadata (LIS2407)

   _____ Indexing and Abstracting (LIS2452)

   _____ Thesaurus Construction (LIS2453)

5. How would you rate yourself as a professional in resource classification?

Very Poor	Poor	Fairly Good	Good	Excellent
1	2	3	4	5

6. How well do you understand the basics and concept of **classification schemes**?

Very Poor	Poor	Fairly Good	Good	Excellent
1	2	3	4	5

7. How well do you understand the basics and concept of **subject headings**?

Very Poor	Poor	Fairly Good	Good	Excellent
1	2	3	4	5

8. How would you rate your **organization skills** in your day-to-day life?

Very Poor	Poor	Fairly Good	Good	Excellent
1	2	3	4	5

9. What do you organize for yourself in ordinary life? (Please check all that apply)

   _____ Personal Library (i.e., books)

   _____ Personal Pictures (i.e., photo albums)

   _____ Personal Computer Folders and Files

_____ Web Pages (e.g. favorites, bookmarks)

_____ Emails/Mails (e.g. folders)

_____ Important Documents (e.g. contracts, receipts, etc.)

_____ Other:

BIBLIOGRAPHY

Abrams, D., Baecker, R., and Chignell, M. (1998). Information archiving with bookmarks: personal Web space construction and organization. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (Los Angeles, California, United States, April 18 - 23, 1998). 41-48.

Adler, M. How to Mark a Book. *Saturday Review of Literature*, July 6, 1941 [Online]. Available: http://www.maebrussell.com/Articles%20and%20Notes/How%20To%20Mark%20A%20Book.html

Bailey, K. (1994) *Typologies and Taxonomies - An Introduction to Classification Techniques*. Thousand Oaks, CA, USA.

Bao, S., Wu, X., Fei, B., Xue, G., Su, Z., and Yu, Y. (2007). Optimizing web search using social annotations. *In Proceedings of the 16th International World Wide Web Conference* (Banff, Alberta, Canada May 8-12, 2007). WWW '07. 501-510.

Bates, M. J. (1998). Indexing and access for digital libraries and the internet: human, database, and domain factors. *Journal of the American Society for Information Science.* 49(13): 1185-1205.

Begelman, G., Keller, P., and Smadja, F. (2006). Automated tag clustering: improving search and exploration in the tag space. In *Proceedings of the Collaborative Web Tagging Workshop at WWW 2006*.

Berners-Lee, T., Hendler, J., Lassila, O. (2001). The Semantic Web. *Scientific American*. 294(5): 28-37.

Bharat, K. and Mihaila, G (2000). Hilltop: A search engine based on expert documents. In *Proceedings of the 9th International WWW Conference (Poster)* (Vol. 10).

Bischoff, K., Firan, C. S., Nejdl, W., and Paiu, R. (2008). Can all tags be used for search? In *Proceedings of the 17th ACM conference on Information and knowledge management*. (Napa Valley, CA, USA, October 26-30, 2008). CIKM '08. 193-202.

Bloom, B. (1985). Generalizations about talent development. In B.S. Bloom (Ed.), *Developing talent in young people*. New York: Ballantine Books. 507-549.

Brin, S. and Page, L. (1998). The anatomy of a large-scale hypertextual web search engine. *In Proceedings of the 7th World Wide Web Conference / Computer Networks 30*(1-7): 107-117.

Brooks, C. H. and Montanez, N. (2006). Improved annotation of the blogosphere via autotagging and hierarchical clustering. In *Proceedings of the 15th international Conference on World Wide Web* (Edinburgh, Scotland, May 23-26, 2006). WWW '06. 625-632.

Bryan, W. and Harter, N. (1899). Studies on the telegraphic language: The acquisition of a hierarchy of habits. *Pyschological Review*. 6: 345-375.

Bush, V. (1945). As we may think. *The Atlantic Monthly*. 176(1): 101-108.

Camerer, C. F. and Johnson, E. J. (1991). The process-performance paradox in expert judgment: how can the experts know so much and predict so badly? In K.A. Ericsson and J. Smith (Eds.), *Towards a general theory of expertise: Prospects and limits*. Cambridge: Cambridge University Press. 195-217.

Campbell, C., Maglio, P., Cozzi, A. and Dom, B. (2003). Expertise identification using email communications. In *Proceedings of the 12th international conference on Information and knowledge management* (CIKM '03). 528-531.

Chan, L. and Hodges, T. (2000). Entering the millennium: a new century for LCSH. *Cataloging & Classification Quarterly*, 29 (1-2): 225-234.

Chi, E. and Mytkowicz, T. (2007). Understanding navigability of social tagging systems. In *Proceedings of the SIGCHI conference on Human Factors in Computing Systems*.

Chi. M. (2006). Two approaches to the study of experts' characteristics. In Ericsson, K. A., Charness, N., Feltovich, P. J. & Hoffman, R. R. (Eds.) *The Cambridge handbook of expertise and expert performance*. Cambridge, UK: Cambridge University Press. 21-30.

Choochaiwattana, W. (2008). Using Social Annotations to Improve Web Search. Doctoral Thesis. University of Pittsburgh.

Choochaiwattana, W. and Spring, M. (2009). Applying social annotations to retrieve and re-rank web resources. In *Proceedings of the International Conference on Information Management and Engineering, 2009*. (ICIME'09). 215-219.

Choy, S. and Lui, A. K. (2006). Web information retrieval in collaborative tagging systems. In *Proceedings of the 2006 IEEE/WIC/ACM international Conference on Web intelligence* (December 18 - 22, 2006). 352-355.

Cohen, J. (1988). Statistical Power Analysis for the Behavioral Sciences. (2nd Ed.). Hillsdale, New Jersey, USA: Lawrence Erlbaum Associates.

Corter, J., and Gluck, M. (1992). Explaining basic categories: feature predictability and information. *Psychological Bulletin*, 111: 291-303.

Dellschaft, K. and Staab, S. (2008). An epistemic dynamic model for tagging systems. In *Proceedings of the 19th Conference on Hypertext and Hypermedia 2008* (Pittsburgh, Pennsylvania, USA, June 19-21, 2008), 71-80.

Djakow, Petrowski, & Rudik (1927). *Psychologie des Schachspiels [Psychology of chess]*. Berlin: Walter de Gruyter.

Dyrholm, M. (2009). A recursive algorithm for forming the constrained elemental cardinality power set. Retrieved on January 28, 2010 from http://www.machlea.com/mads/papers/powerset.pdf.

Ericsson, K. (2006). An introduction to *Cambridge Handbook of Expertise and Expert Performance*: Its development, organization, and content. In K.A. Ericsson, N. Charness, P. J. Feltovich, and R. R. Hoffman (Eds.), *The Cambridge handbook of expertise and expert performance*. Cambridge, UK: Cambridge University Press. 3-19.

Ericsson, K., Krampe, R., and Tesch-Römer, C. (1993). The role of deliberate practice in the acquisition of expert performance. *Psychological Review*. 100: 363-406.

Ericsson, K. and Lehmann, C. (1996). Expert and exceptional performance: Evidence on maximal adaptations on task constraints. *Annual Review of Psychology*. 47: 273-305.

Ericsson, K. and Smith, J. (1991). Prospects and limits in the empirical study of expertise: an introduction. In K. A. Ericsson and J. Smith (Eds.), *Towards a general theory of expertise: prospects and limits*. Cambridge: Cambridge University Press. 1-38.

Feinberg, M. (2006). An examination of authority in social classification systems. In *Proceedings of the 17th SIG Classification Research Workshop, 2006*.

Becerra-Fernandez, I. (2006). Searching for experts on the Web: A review of contemporary expertise locator systems. *ACM Transactions on Internet Technology*. 6(4): 333-355.

Fisher, D. H. (1987). Knowledge acquisition via incremental conceptual clustering. *Machine Learning*, 2(2):139-172.

Fleiss, J. L. (1971). Measuring nominal scale agreement among many raters. *Psychological bulletin*, 76(5), 378.

Garshol, L. M. (2004). Metadata? thesauri? taxonomies? topic maps! making sense of it all. *Journal of Information Science*, 20(4):378-391.

Gaus W (1995) *Dokumentations- und ordnungslehre - theorie und praxis des information retrieval*. 2nd ed., Berlin et al.

Gemmell, J., Shepitsen, A., Mobasher, M., and Burke, R. (2008). Personalization in folksonomies based on tag clustering. In *Proceedings of the 6th Workshop on Intelligent Techniques for Web Personalization and Recommender Systems*. 37–48.

Gluck, M. A., & Corter, J. E. (1985). Information, uncertainty, and the utility of categories. In *Proceedings of the Seventh Annual Conference of the Cognitive Science Society*. Irvine, CA: Lawrence Erlbaum Associates. 283-287.

Gobet, F., & Simon, H. A. (1996). Recall of rapidly presented random chess positions is a function of skill. *Psychonomic Bulletin and Reviews*. 3 , 159–163.

Golder, S. and Huberman, B. (2006). Usage patterns of collaborative tagging systems. *Journal of Information Science*. 32(2), 198-208.

Gruber, T. (1993). A translation approach to portable ontology specifications. *Knowledge Acquisition*. 5(2), 199-220.

Guy, M., Tonkin, E. (2006), "Folksonomies: tidying up tags?", *D-Lib Magazine*, available at: www.dlib.org/dlib/january06/guy/01guy.html (accessed April 23, 2010). 12(1).

Halpin, H., Robu, V., and Shepherd, H. (2007). The complex dynamics of collaborative tagging. In *Proceedings of the 16th international conference on World Wide Web*, (Banff, Alberta, Canada, May 8-12, 2007). WWW '07. 211-220.

Hammond, T., Hannay, T., Lund, B., and Scott, J. (2005). Social bookmarking tools (I). *D-Lib Magazine, 11(4),* Retrieved March 8, 2010, from http://www.dlib.org/dlib/april05/hammond/04hammond.html.

Hassan-Montero, Y. and Herrero-Solana, V. (2006). Improving tag-clouds as a visual information retrieval interfaces. In *Proceedings of International Conference on Multidisciplinary Information Sciences and Technologies*. (Medira, Spain, October 25-28, 2006). InSciT2006.

Heymann, P. and Garcia-Molina, H. (2006). Collaborative creation of communal hierarchical taxonomies in social tagging systems. Technical Report 2006-10, Stanford University.

Heymann, P., Koutrika, G., and Garcia-Molina, H. (2008). Can social bookmarking improve web search?. In *Proceedings of the international Conference on Web Search and Web Data Mining* (Palo Alto, California, USA, February 11-12, 2008). WSDM '08. 195-206.

Heymann, P., Ramage, D., and Garcia-Molina, H. (2008). Social tag prediction. In *Proceedings of the 31st annual international ACM SIGIR conference on Research and development in information retrieval*. SIGIR ’08. 531-538.

Hood, S. (2008). Delicious is 5! [Web Posting]. November 6, 2008. Retrieved from http://blog.delicious.com/blog/2008/11/delicious-is-5.html on March 31, 2010.

Jarvelin, K. and Kekalainen, J. (2002). Cumulated gain-based evaluation of IR techniques. *ACM Trans. Information Systems*, 20(4): 422-446.

John, A. and Seligmann, D. (2006). Collaborative tagging and expertise in the enterprise. In *Proceedings of the WWW 2006 Collaborative Web Tagging Workshop, 2006*.

Kalfoglou, Y. and Schorlemmer, M. (2003) Ontology mapping: The state of the art. *The Knowledge Engineering Review*. Cambridge, UK: Cambridge University Press. 18, 1-31.

Keller, R., Wolfe, S., Chen, J., Rabinowitz, J., and Mathe, N. (1997). A bookmarking service for organizing and sharing URLs. *Computer Networks and ISDN Systems*. 29(8-13): 1103-1114.

Kipp, M. E. and Campbell, D. G. (2006), Patterns and inconsistencies in collaborative tagging systems: An examination of tagging practices. *Proceedings of the American Society for Information Science and Technology*, 43:1–18.

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

Kome, S.H. (2005) Hierarchical subject relationships in folksonomies. Master's thesis, University of North Carolina at Chapel Hill, Chapel Hill, NC, USA.

Koutrika, G., Effendi, A., Gyongyi, Z., Heymann, P., and Garcia-Molina, H. (2008). Combating spam in tagging systems: An evaluation. *ACM Trans. Web*. 2(4): 1-34.

Krestel, R., Fankhauser, R., and Nejdl, W. (2009). Latent dirichlet allocation for tag recommendation. In *Proceedings of the third ACM conference on Recommender systems*. RecSys '09. 61-68.

Kroski, E. (2005). The hive mind: folksonomies and user-based tagging. *InfoTangle [blog]*. Retrieved February 8, 2010 from http://infotangle.blogsome.com/2005/12/07/the-hive-mind-folksonomies-and-user-based-tagging/

Lakoff, G. (1987). *Women, fire, and dangerous things: what categories reveal about the mind*. Chicago: University of Chicago Press.

Langridge, D. (1989). *Subject analysis: principles and procedures*. London: Bowker-Sauer.

Lansdale, M. (1983). The psychology of personal information management. *Applied Ergonomics*. 19:55-66.

Lee, S.H., Kim, P.J., and Jeong. H. (2006). Statistical properties of sampled networks. *Physical Review E*, 73:016102.

Leskovec, J. and Faloutsos. C. (2006). Sampling from large graphs. In *Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining.* (Philadelphia, Pennsylvania, USA, August 20-23, 2006), 631–636.

Li, X., Guo, L., and Zhao, Y. E. 2008. Tag-based social interest discovery. In *Proceedings of the 17th international Conference on World Wide Web* (Beijing, China, April 21-25, 2008). WWW '08. 675-684.

Macdonald, C., Hannah, D. and Ounis, I. (2008). High quality expertise evidence for expert search. In *Proceedings of the 30th European Conference on IR Research*, UK, 283-295.

Marlow, C., Naaman, M., Boyd, D., and Davis, M. (2006). Ht06, tagging paper, taxonomy, flickr, academic article, to read. In *HYPERTEXT '06: Proceedings of the seventeenth conference on Hypertext and hypermedia* (New York, NY, USA. ACM.) 31-40.

Marshall, C. (1997). Annotation: From Paper Books to Digital Library. In *Proceedings of DL'97*. (Philadelphia, PA, USA.) 131-140.

Marshall, C. (1998). Toward an Ecology of Hypertext Annotation. In *Proceedings of ACM Hypertext'98*. (Pittsburgh, PA, USA.) 40-49.

Michalski, R. S. and Stepp, R. (1983). Learning from observation: conceptual clustering," *In Machine Learning: An Artificial Intelligence Approach*, R. S. Michalski, T. J. Carbonell and T. M. Mitchell (Eds.), Palo Alto: TIOGA Publishing Co. 331-363.

Michalski, R. and Stepp, R. (1983). Automated construction of classifications: Conceptual clustering versus numerical taxonomy. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 5(4):396-409.

Mika, P. (2007). Ontologies are us: A unified model of social networks and semantics. *Journal of Web Semantics, 5*(1), 5-15.

Millen, D., Feinberg, J., and Kerr, B. (2005). Social bookmarking in the enterprise. *Queue*, 3(9):28-35.

Millen, D. R., Feinberg, J., and Kerr, B. (2006). Dogear: Social bookmarking in the enterprise. In *CHI '06: Proceedings of the SIGCHI conference on Human Factors in computing systems*. 111-120.

Noll, M. G., Au Yeung, C., Gibbins, N., Meinel, C., and Shadbolt, N. (2009). Telling experts from spammers: expertise ranking in folksonomies. In *Proceedings of the 32nd international ACM SIGIR conference on Research and development in information retrieval* (SIGIR '09). 612-619.

Petkova, D., Croft, W.B. (2006). Hierarchical language models for expert finding in enterprise corpora. *18th IEEE International Conference on Tools with Artificial Intelligence (ICTAI'06)*. 599-608.

Petkovic, D., Lank, E., Ramirez, F. A., Raghavendra, S., Chen, F., Pekiner, C., Fregoso, A., and Marquez, A. (2005). Asynchronous multimedia annotations for web-based collaboration in biology education. In Proceedings of SPIE Conference on Multimedia Storage and Retrieval. 5682, 108-113.

Ramage, D., Heymann, P., Manning, C. D., and Garcia-Molina, H. (2009). Clustering the tagged web. In *Proceedings of the Second ACM international Conference on Web Search and Data Mining* (Barcelona, Spain, February 09-12, 2009). WSDM '09. 54-63.

Ranganathan, S. (1963). *Colon classification. Basic classification (6th ed.)* Bombay, India: Asia Publishing House.

Rosch, E. (1973). Natural categories. *Cognitive Psychology*, 4(3):328-350.

Santos-Neto, E., Ripeanu, M., and Iamnitchi, A. (2007). Tracking user attention in collaborative tagging communities. In *Proceedings of International ACMIEEE Workshop on Contextualized Attention Metadata: personalized access to digital resources*.

Sauperl, A. (2002). *Subject determination during the cataloging process*. Lanham, MD: Scarecrow Press, Inc.

Shannon, C. (1949). Communication in the presence of noise. *Proceedings of the IRE*, 37(1):10-21.

Shi, X., Bonner, M., Adamic, L., Gilbert, A. (2008). The very small world of the well-connected. In *Proceedings of the 19th Conference on Hypertext and Hypermedia 2008* (Pittsburgh, Pennsylvania, USA, June 19-21, 2008), 61-70.

Shirky, C. (2005). Ontology is overrated: categories, links, and tags. *Clay Shirky's Writings About the Internet: Economics & Culture, Media & Community [blog]*. Retrieved February 8, 2010 from http://www.shirky.com/writings/ontology_overrated.html

Simon, H. and Chase, W. (1973). Skill in chess. *American Scientist*. 61: 394-403.

Smith, T. (2007). Cataloging and You: Measuring the Efficacy of a Folksonomy for Subject Analysis. In J. Lussky (Ed.) *18th Workshop of the American Society for Information Science and Technology Special Interest Group in Classification Research*. Available at http://arizona.openrepository.com/arizona/handle/10150/106434.

Specia, L. and Motta, E. (2007). Integrating folksonomies with the semantic web. In *The Semantic Web: Research and Applications*. Springer Berlin (Heidelberg), 4519: 624-639.

Speller, E. (2007). Collaborative tagging, folksonomies, distributed classification or ethnoclassification: a literature review. *Library Student Journal*. February, 2007.

Syn, S., & Spring, M. (2009). Tags as keywords – comparison of the relative quality of tags and keywords. *Proceedings of the American Society for Information Science and Technology*, 46(1), 1-19.

Syn, S. (2010). Generation of classificatory metadata for web resources using social tags. Doctoral Thesis. University of Pittsburgh.

Taylor, A. (1999). *The organization of information*. Englewood, CO: Libraries Unlimited.

Tonkin, E. (2006). Searching the long tail: Hidden structure in social tagging. In *Proceedings of the 17th ASIS&T SIG/CR Classification Research Workshop*. (Austin, TX, USA, November 4, 2006).

Trant, J. (2009). Studying social tagging and folksonomy: a review and framework. *Journal of Digital Information. 10(1).* Retrieved on January 21, 2010 from https://journals.tdl.org/jodi/article/viewArticle/269.

Udell, J. (2005). Jon Udell: language evolution in del.icio.us. Retrieved April 21, 2010 from http://jonudell.net/udell/gems/delicious/delicious.html.

Van Setten, M., Brussee, R., Van Vliet, H., Gazendam, L., van Houten, Y., & Veenstra, M. (2006). On the importance of "Who Tagged What". *Workshop on the Social Navigation and Community-Based Adaptation Technologies, In Conjunction with Adaptive Hypermedia and Adaptive Web-Based Systems (AH'06)*, Dublin, Ireland.

Vander Wal, T. (2005) Explaining and Showing broad and narrow folksonomies. *Off the Top [blog]*. Retrieved on April 22, 2010 from http://www.personalinfocloud.com/2005/02/explaining_and_.html.

Vander Wal, T. (2007), "Folksonomy coinage and definition", 2 February, available at: http:///vanderwal.net/folksonomy.html (accessed April 23, 2010).

Vatton, I., Quint, V., Kahan, J., Cramer, K., Nylander, K., Rosen, K., Spinella, M., and LeDoux, L. (2004) *Amaya User Manual*. Retrieved on March 8, 2010 from http://www.w3.org/Amaya/User/Doc/Manual.html

Voss, J. (2007), Tagging, folksonomy and co-renaissance of manual indexing?, In *Proceedings of the International Symposium of Information Science*, pp. 234-54.

Zhang, J., Ackerman, M., and Adamic. L. (2007). Expertise networks in online communities: structure and algorithms. In *Proceedings of the 16th international conference on World Wide Web* (WWW '07). 221-230.