Discovery Tool: A Framework for Accelerating Academic Collaborations

THESIS

Presented in Partial Fulfillment of the Requirements for the Degree Master of Science in the Graduate School of The Ohio State University

By

Mitesh Mukesh Kanjariya

Graduate Program in Computer Science and Engineering

The Ohio State University

2013

Master's Examination Committee:

Dr. Jay Ramanathan, Thesis Advisor

Dr. Rajiv Ramnath

Dr. Caroline Wagner
Copyright by
Mitesh Mukesh Kanjariya
2013
Abstract

Science and technology long ago passed from luxury to necessity. Yet, the mechanisms and processes by which researchers connect still resemble the days of the filing cabinet. The state of compartmentalized research persists even as studies show that innovation in science, technology, and the arts often emerge at the nexus of these fields. In the recent past, there has been an increase in collaborative research but interdisciplinary research remains small. Hence, in this thesis we intend to address the questions like: How can we accelerate the collaboration process among researchers? What are the patterns of academic collaboration that add value? Can we predict when collaborations will be useful? What are the factors that affect the establishment of such collaborations?

To address these questions, we propose a framework that provides the architecture for developing the features of a system which will help to accelerate the academic collaboration process. The framework includes intelligent techniques to visualize the vast amount of academic data. In order to build such a generalized framework and systematically understand the requirements of a collaboration system, we first start with a basic academic search tool implemented on top of published data assets of an institution. Using the feedback from the prototype, we create a visual framework for a system which will accelerate the academic collaboration process. This framework helps in the design
and implementation of collaboration features and enhances the knowledge discovery process across an institution. It also helps in connecting researchers together and enables the connection between researchers and actual or potential users.

In this thesis, we provide initial architecture for an academic search system. We design and implement Discovery Tool based on this architecture. Discovery Tool acts as an interface with which agents (professionals, academics and students) interact with comprehensive data assets (publications, experts and needs). It provides us with an interface that helps to understand the future needs and requirements of a generalized framework for accelerating academic collaborations. We use the feedback from Discovery Tool to develop an integrated framework for accelerating academic collaborations.

We address two research problems:

- A need for design and implementation of an academic search system
- To present a generalized framework for accelerating the design of an academic collaboration system
Dedication

This thesis is dedicated to my parents, Mukesh and Indu Kanjariya, and my brother, Neel.
Acknowledgments

I would like to thank my advisors Prof. Jay Ramanathan and Prof. Caroline Wagner for their continuous support. I would also like to thank Prof. Rajiv Ramnath for his advice and guidance throughout my graduate studies. I would like to thank the entire OCIO team, without their help the project would not have been realized. I would like to thank Vedu and John from the KMData team for their support, insights and resources which they provided during the course of the project.

I would like to thank Zhe and Derrick for working along with me on this project.

I would like to thank my roommates Kshitij, Aditya and Rohit who have been my family away from home. I will take this opportunity to thank all my friends Aishu, Ananth, Prathyusha, Ramya, Uzma, Abhishek, Swaroop, Shweta and Chirag for their unwavering support and belief in me.

Best of luck to Pallavi as she takes on my role at the Battelle Center for Science and Technology Policy.

This work was done under the aegis of the Center for Experimental Research in Computer Science at Georgia Tech and The Ohio State University, funded by the National Science Foundation's I/UCRC program (NSF IIP 0753710). Funding was also provided by The Ohio Board of Regents Computer Science Graduate Education program and by the National Science Foundation Course, Curriculum, and Laboratory
Improvement program (NSF CCLI Phase 0837555). We wish to acknowledge the helpful ideas by CETI students, colleagues, members, and sponsors.
Vita

2007 to 2011 ........................................... B. Tech Information Technology, Veermata Jijabai Technological Institute, University of Mumbai

2011 to present ................................. M.S. Computer Science and Engineering, The Ohio State University

Fields of Study

Major Field: Computer Science and Engineering
# Table of Contents

Abstract........................................................................................................................................... ii

Dedication .......................................................................................................................................... iv

Acknowledgments .............................................................................................................................. v

Vita ...................................................................................................................................................... vii

List of Tables ..................................................................................................................................... xiii

List of Figures ................................................................................................................................... xiv

Chapter 1: Introduction ....................................................................................................................... 1

   Terminology ...................................................................................................................................... 4

      Author .......................................................................................................................................... 4

      Publication .................................................................................................................................... 5

      Index Words ................................................................................................................................. 5

      Key Words .................................................................................................................................... 5

      Co-Author and Co-author Network ............................................................................................... 5

      Friend of a Friend .......................................................................................................................... 6

      Referenced Author ........................................................................................................................ 6

      Service Blueprint .......................................................................................................................... 6
H-index
G-index
PI
Scopus
KMDATA
IDEF
Framework
Service Oriented Architecture
Software Interaction Diagrams
CIP
Problem statement
Problem I
Problem II
Primary Contributions
Initial system design, model and analysis
Prototype system implementation
Usability Surveys and Feedback
Generalized framework for accelerating academic collaboration
Chapter 2: Related Work
Scenario 3: Incoming Student ................................................................. 27
System Architecture ............................................................................. 28
Profile Building ...................................................................................... 29
Expert Finding ....................................................................................... 30
Visualization .......................................................................................... 30
Data Sources ............................................................................................ 30

Chapter 5: Discovery Tool: Methods and Implementation ..................... 38
Profile Building Component ................................................................. 39
H-index calculator .................................................................................. 41
G-index calculator .................................................................................. 43
Author Network extractor ...................................................................... 44
Key areas .................................................................................................. 45
Expert finding component ...................................................................... 46
Visualization Component ....................................................................... 47
Co-Authors .............................................................................................. 49
Referenced Authors ................................................................................. 50
Friend-of- friend ...................................................................................... 51
Interactive tree/dendrogram view .......................................................... 53
Pack diagram .......................................................................................... 55
List of Tables

Table 1: Comparing Visualization Features of Discovery Tool with Existing Systems ...15
Table 2: Comparing Discovery Tool with Existing Systems...........................................16
Table 3: Ohio State Faculty .................................................................................................72
List of Figures

Figure 1: Current Situation – PI searching for collaborators.................................................2
Figure 2: Proposed Solution – PI submits basic search criteria and taps into accumulated
data within the system and it is visualized for them............................................................4
Figure 3: Example Datasets of Discovery Tool .........................................................................19
Figure 4: User Scenario 1- Professor (PI) using the Expert Finder system.........................25
Figure 5: User Scenario 2 – Industry Partner using the Expert Finder System to Find Top
Experts ....................................................................................................................................26
Figure 6: User Scenario 3- Prospective or Currently Enrolled Student is searching for an
Advisor ......................................................................................................................................27
Figure 7: Discovery Tool System Architecture ............................................................29
Figure 8: IDEF 0 model for Discovery Tool - Expert Finding System.................................32
Figure 9: IDEF 1 model for Discovery Tool - Expert Finding System.................................33
Figure 10: IDEF 0 Model for Profile Building Component ....................................................34
Figure 11: IDEF 1 Model for Profile Building Component ....................................................35
Figure 12: IDEF 0 Model for Expert Finding Component .....................................................36
Figure 13: IDEF 1 model for expert finding component..........................................................37
Figure 14: Software Interaction Diagram; Use Case 1- Search Author .................................38
Figure 15: Software Interaction Diagram; Use Case 2- Find Expert .................................39
Figure 16: Profile Building Component ................................................................. 40
Figure 17: H-Index Calculation...................................................................................... 42
Figure 18: Block Diagram for H-index Calculator ......................................................... 43
Figure 19: Block Diagram for G-index Calculator ......................................................... 44
Figure 20: Schema for Expert Finding Component ....................................................... 47
Figure 21: Schema for Visualization Component .......................................................... 48
Figure 22: Co-Authors Graph Visualization of a Computer Science Faculty ............... 50
Figure 23: Referenced Authors Network for a Computer Science Faculty .................. 51
Figure 24: Friends of Friend’s network for a Computer Science Faculty ..................... 53
Figure 25: Dendrogram View for a Computer Science Faculty .................................... 55
Figure 26: Example of a Pack Diagram for a Computer Science Faculty ..................... 56
Figure 27: Indented Tree Visualization ........................................................................ 57
Figure 28: Home Page of Discovery Tool ..................................................................... 58
Figure 29: Author Profile Page of Discovery Tool ......................................................... 59
Figure 30: Online Feedback service, Discovery Tool ..................................................... 60
Figure 31: Moments of truth ......................................................................................... 63
Figure 32: Total service offer ...................................................................................... 64
Figure 33: service Blueprint ........................................................................................ 65
Figure 34: Service Blueprint ....................................................................................... 66
Figure 35: Scores for Visualization Features ............................................................... 68
Figure 36: Ratings for Usefulness of Visualization Features in Collaboration ............ 69
Figure 37: Completeness of Author Profiles ............................................................... 70
Figure 38: User Friendliness .................................................................71
Figure 39: Proposed Framework .........................................................74
Figure 40: Framework Augmented with History Module ....................75
Figure 41: The History Module ..........................................................76
Figure 42: Augmenting the Visualizations .........................................77
Chapter 1: Introduction

Prior research has shown that inter- and intra-disciplinary collaboration has a large impact on the field of research. For example, the revolution in the field of biomedical informatics would have been impossible without the collaboration between the researchers of Computer Science and Biology. Moreover, research in certain fields such as Computer Science can have a substantial impact on research in other fields. It is due to such inter-disciplinary collaboration that many tasks have become affordable. Research collaboration often brings more point of views to the problem been addressed. Studies show that productivity and the extent of collaboration are correlated (Martin, 1997). It is therefore beneficial to have an end-to-end system which will address the problem of finding the potential collaborators.

In today’s world, academic institutions are constantly trying to foster multi-discipline research to solve increasingly complex issues. The logistics of those we choose to collaborate with can be a tiring process. In addition, considerable amount of time is spent in identifying the right resources and expertise across the enterprise. The user community has shown that the greater the amount of time wasted, the larger the cost for
the research activity, not to mention the overall time to complete the research is extended. These types of situations affect the bottom line in academic contexts. We need to find effective ways to foster scientific collaborations and ultimately improve the quality of output for related business deliverables. There is an emerging need for a framework which can serve as reference architecture for developing a system which can support academic collaboration.

We can see the problem in Figure 1.

![Figure 1: Current Situation – PI searching for collaborators](image)

A PI is trying many different methods with which to find potential collaborators. By rendering keyword based queries, the PI submits queries to different academic journals and aggregation services such as Google Scholar. The output returned is essentially raw and unstructured, consisting of paper abstracts, which have to be sifted through manually for contextually relevant material. This raw unprocessed information has to be processed by the PI, based on their individual motivation and experience. Results will vary considerably, and the cycle shown in the cloud emerges. As the ideas
come, reflection upon the results emerges and drives preparation for the next phase in the search for the perfect collaborator. Months pass by without result until finally a moment particularly suited for initiation occurs. This event is again, quite infrequent and is surely celebrated when it comes. However, there is a better way to proceed as we will see in the following sections of this document.

In order to reduce the time and effort spent in finding collaborators for research in organizations large and small, we propose a system to be implemented within the context of The Ohio State University. This system is targeted to accomplish the goal of fostering significant multi-discipline research by pairing up ideal candidates in a fraction of the time. The research here is to complement existing vendor technologies and open source tools and develop an interactive framework which will accelerate the academic collaboration process. The framework described here is implemented as a system with visualization, author profile management and a simple and intuitive user interface that enables different ways to search for field experts and find potential collaborators. The framework allows different criteria to be used for searching collaborators. Specifically, we provide a framework for extracting author networks such as the coauthor networks, referenced author network, friend of a friend network, research interests, and network browsing patterns of a researcher.

We designed an initial system that solves the problem of searching efficiently for potential collaborators. This system assists various types of PIs by searching and structuring the relevant data. Another side effect of restructuring the data for a query is
that we can start to draw out the relationships between authors and co-authors of papers. This information is invaluable for quickly producing results for the user PIs.

![Diagram](image)

**Figure 2: Proposed Solution** – PI submits basic search criteria and taps into accumulated data within the system and it is visualized for them.

As we can see in figure 2, the system is playing a buffering role for searching a seemingly random pool of collaborators. The visualization component takes the summarized results and puts them into immediately understandable formatting. This is one of the key features that drive down time spent on this activity. Thus, in summary, we gain months due to the usage of the Discovery Tool.

**Terminology**

**Author**

In the context of this thesis, an author or an academic author refers to a researcher or a scientist who has at least one publication.
**Publication**

In the context of this thesis, a publication refers to a paper published in a journal or a conference. A publication can have one or more authors associated with it. It can be cited by other publications or can cite other publications.

**Index Words**

In the context of this thesis, index words are the set of words which describe a publication.

**Key Words**

In the context of this thesis, key words are the set of words/phrases provided by an author to describe a publication.

**Co-Author and Co-author Network**

We define a co-author and co-author network of an author as follows:

An author ‘Y’ is said to be a coauthor of author ‘X’ if and only if author ‘X’ and author ‘Y’ have at least one joint publication.

Co-author network of author ‘X’ is the set of all co-authors of author ‘X’.
Friend of a Friend

We define a friend of friend and friends of friend’s network as follows:

An author ‘Z’ is said to be a friend of friend of author ‘X’ if and only if there exists no joint publication between authors ‘X’ and ‘Z’ and there exists at least one joint publication between author ‘Z’ and author ‘Y’ such that author ‘Y’ is a co-author of author ‘X’.

Friends of friend’s network of author ‘X’ are the set of all Friend of a friend of author ‘X’.

Referenced Author

In the context of this thesis, a referenced author and referenced author network is defined as follows:

Author ‘Y’ is said to be referred by author ‘X’ if there exists at least one publication of author ‘X’ which cites one or more publications of author ‘Y’.

Referenced author network of author ‘X’ is the set of all referenced author of author ‘X’.

Service Blueprint

The process of service blueprint (Shostack L. , 1984) helps us to devise meaningful questions for our user surveys.
**H-index**

An author is said to have an H-index of ‘h’, if he/she has received at least ‘h’ citations for ‘h’ of his publications. (JE, 2005)

**G-index**

Assuming a list of publications sorted in descending order of individual citations, an author has a unique g-index ‘g’ such that the first ‘g’ articles have a cumulative citation of ‘g^2’. (Egghe, 2006)

**PI**

In the context of this thesis, a PI (Principal Investigator) is the primary user of the system. A PI can refer to faculty members of Ohio State University, Industry Partners or Ohio State University and/or potential or current students of the Ohio State University.

**Scopus**

Scopus or SciVerse Scopus (Scopus in detail: What does it cover?, 2013) provides publication datasets. The publication dataset consists of abstracts, title, citations and key words. Scopus acts as the primary dataset for the initial Discovery Tool implementation.

**KMData**

The Knowledge Management Data service (known as KMData) provides Ohio State University’s publication records.
IDEF

IDEF (Integration Definition) (IEEE Standard for Functional Modeling Language—Syntax and Semantics for IDEF0, 1998) is a modeling language extensively used to model the Discovery Tool.

Framework

A software framework provides an abstract view of all the functional components of a software system. It is a design on which fully functional application can be build.

In this thesis, we provide a software framework for building a collaboration system for The Ohio State University.

Service Oriented Architecture

Service Oriented Architecture (Velte, 2010) is a design which provides functional components of a large software system.

Software Interaction Diagrams

Software Interaction Diagrams (Types of modeling diagrams, 2008) are set of UML diagrams which are used for representing the data flow and interaction of the user with the system. They provide a simple way to model the complex interactions among the sub components of a system.
CIP

CIP (Classification of Instructional Programs) provides a standard taxonomy for the various departments of a university.

Problem statement

Problem I

Initial need for design and implementation of an academic search system for the Ohio State University

A system that fosters academic discovery within the context of The Ohio State University needs to be established. We begin with an initial need for the development of a prototype system which will help scientists to capture data required for understanding the factors that govern academic collaboration. The thesis proposes initial architecture for an academic search system. Discovery Tool is designed and implemented based on this architecture. Discovery Tool acts as an interface with which agents (professionals, academics and students) interact with comprehensive data assets (publications, experts and needs). Such an interface is essential to capture patterns of collaboration and will help us in understanding the future needs and requirements towards developing a generalized framework for accelerating inter and intra disciplinary scientific collaborations.
Problem II

Need for a generalized framework for new collaboration features

Feedback in which users provide usability input is essential for documenting the future needs and requirements which will help in enhancements of the system. We need to learn from the prototype system implementation and develop a generalized framework which will provide component services for future development. Based on the collected feedback and critiques on the prototype system, we need to develop an integrated end-to-end framework. Such a system with visualization, data management, and user interface will allow different feature options to be developed for collaborators.

Primary Contributions

This thesis has the following contributions:

Initial system design, model and analysis

We summarize the initial requirements of an academic search tool and provide a systematic model of the tool based on these early requirement analyses. Specifically, we list three high level initial user scenarios and implement a modular system architecture tailored for resources available within the Ohio State University.

For system modeling we leverage the widely accepted function modeling methodology, IDEF (Icam DEFinition for functional modeling, where ICAM stands for Integrated Computer Aided Manufacturing). In particular, we provide the IDEF 0 and IDEF 1 models for individual components of the prototype system.
Prototype system implementation

Discovery Tool is implemented as a spring MVC based web portal using the widely used D3.js visualization library and is accessible within and outside of the Ohio State University. Discovery Tool is a prototype system which will facilitate quick searches of research data. We also provide an interactive user friendly interface through which scientists can interact with the comprehensive data assets.

Usability Surveys and Feedback

The next aspect of this thesis is that the framework is adapted and modified based on the critiques and user feedback. We provide an analysis of the responses we gathered while surveying the primary users of Discovery Tool. These surveys help us in understanding what visualization of the underlying data facilitates collaboration and how can we modify the prototype architecture to a more generalized open architecture which favors the development of features for academic collaboration.

Generalized framework for accelerating academic collaboration

We build on the prototype of the Discovery Tool to formalize a generic framework which will serve as guidelines for future development. Specifically, we describe a framework which will foster the academic collaboration process. The proposed system is an end-to-end framework with visualization, data management, and simple user interface.
Chapter 2: Related Work

Related work is broadly classified and discussed below:

Structure of scientific collaboration network

Social networks of scientists such as the co-author network and friends of friend’s network have been studied extensively in the literature. There are various studies in the literature on co-authorship patterns (Newman, 2004) (Schubert, 2004) and co-citation patterns (Persson O. et al, 1995) (Crane D., 1972). Newman tried to formalize the relationships in academic collaborations through his extensive works (Newman, 2001) (Newman, 2000). Most of the existing research implies that studies in coauthoring patterns provide valuable insights in understanding the nature of scientific collaboration. It has been shown that the academic collaboration has characteristic such as the “small world hypothesis” and that two scientists who have a common collaborator are more likely to start joint research activities (Newman, 2001). Despite the fact that these studies gave valuable insights in understanding academic collaboration they were one time static studies based on past published assets of researchers. In order to accelerate collaboration and the expert finding process, we need an adaptive framework which can extract these networks and sits on top of published and unpublished data assets. We try to integrate
these findings in a generalized framework. We propose and implement Discovery Tool which provides efficient visual ways to analyze author networks on the fly.

**Visualization techniques**

An academic collaboration system has to work with large data sets containing millions of publication records. Listing the publication records as pure text no longer serves the need of researchers. We need efficient visualization methods to surface the underlying meaning of the data. Specifically, most relevant to our work is the visualization of co-authorship networks. Most of the current systems uses a force directed graph for visualization of author networks were nodes represent authors and links represent the collaboration between them (Huang, 2006) (Microsoft ©) (Conlon, 2009) (Devare, 2007)(AMiner©). While these representations gives insights about the researcher’s past collaboration, they fail to capture valuable information related to unique areas of researchers. Our work develops on these approaches and provides additional information with full utilization of the visualization factors such as the thickness of the links, the size of the nodes and the color of the nodes.

**Information Visualization Framework**

Variety of tools and libraries are available for visualizing the data in a meaningful way. There are sophisticated network visualization tools which are used widely (Starlight) (Pajek). In the proposed framework, the visualization module uses the recently developed framework D3.js (Michael Bostock, 2011). D3.js acts as a simple java
script library which can create efficient visualizations with the help of the underlying data. Discussions on the comparison of all these technologies are out of the scope of this document. We utilized the survey conducted by the Office of the Chief information Officer, Ohio State University and decided to use D3.js for the implementation of the visualization component.

**Existing collaboration systems**

Recently, academic search engines have gained interest among commercial companies like Google and Microsoft. Though these systems provide excellent visualization features and efficient search functionality, they hardly help with collaboration. These systems do not consider the collaboration aspects while displaying the data to the user. Also, the key difference between these systems and the proposed framework lies in the underlying data. While the existing systems cover prominent conference and journal articles, they fail to include the ongoing projects and various other publications. On the other hand, Discovery Tool framework provides data which is specific to The Ohio State University as an enterprise. As most of the implementation details of these approaches are not publicly available for examination, we only compare the higher level architecture and features of these systems with the proposed framework.

Researchers have also developed systems to cater to these needs. ResearchIQ (Omkar Lele, 2012), VIVO (Conlon, 2009), AMiner© tries to solve the problem. The main difference between these frameworks and our approach lies in the data management
and the intelligence extraction layer. Table 1 summarizes the main differences of Discovery Tool and these systems.

<table>
<thead>
<tr>
<th>System</th>
<th>Visualization features</th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Co-authors</td>
<td>Referred</td>
<td>Friends of</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>authors</td>
<td>friends</td>
</tr>
<tr>
<td>Microsoft</td>
<td>✓</td>
<td></td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Academic Search</td>
<td>✓</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ResearchIQ</td>
<td>✓</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Arnetminer</td>
<td>✓</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>VIVO</td>
<td>✓</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Google Scholar</td>
<td>✓</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Discovery Tool</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
</tbody>
</table>

Table 1: Comparing Visualization Features of Discovery Tool with Existing Systems

While most of the existing systems provide visualization features for co-author networks, they fail to capture other important relationships such as the friends of friend’s network or the referenced author network. On the contrary, Discovery Tool provides efficient visualization techniques for visualizing the academic networks of an author. This networks form a key aspect for an academic collaboration system as proven by the faculty survey documented in the later sections of this thesis.
<table>
<thead>
<tr>
<th>System</th>
<th>License</th>
<th>Data</th>
<th>RDBMS/Semantic</th>
<th>Collaboration</th>
</tr>
</thead>
<tbody>
<tr>
<td>Microsoft Academic Search</td>
<td>Proprietary</td>
<td>Best for Computer Science</td>
<td>Proprietary</td>
<td>×</td>
</tr>
<tr>
<td>ResearchIQ</td>
<td>Open Source</td>
<td>Limited to the medical domain</td>
<td>Semantic</td>
<td>×</td>
</tr>
<tr>
<td>Arnetminer</td>
<td>Open Source</td>
<td>Decent coverage</td>
<td>RDBMS</td>
<td>×</td>
</tr>
<tr>
<td>VIVO</td>
<td>Open Source</td>
<td>Institution Specific</td>
<td>Semantic</td>
<td>×</td>
</tr>
<tr>
<td>Google Scholar</td>
<td>Proprietary</td>
<td>Almost Complete</td>
<td>Proprietary</td>
<td>×</td>
</tr>
<tr>
<td>Discovery Tool</td>
<td>Open Source</td>
<td>Complete for OSU</td>
<td>RDBMS</td>
<td>✓</td>
</tr>
</tbody>
</table>

Table 2: Comparing Discovery Tool with Existing Systems

Data model for a search tool can be implemented in two ways: semantic or relational database. Discussions on the comparison of these technologies are out of the
scope of this document. We utilize the survey conducted by the Office of the Chief information Officer, Ohio State University and decided to develop our system on top of an ontology driven relational data model. Most commercial system offers a good search system. However, the disadvantage of such system is that the details are proprietary. Discovery Tool provides an open source solution to the implementation of an academic search system.

**Search personalization**

A wide variety of personalized search techniques are suggested in the literature. A number of these approaches use previous interaction of the user with the search engine to describe that user. Most of these approaches can be classified in two categories: a session based personalization or a long term history based personalization (Micro Sperett, 2005) (S. Sriram, 2004). There are some techniques which have tried to incorporate more information about the user by storing full browsing history (J. Teevan, 2005). Other approaches have tried to capture additional information like e-mails and calendar information of an individual for personalizing the search (Radlinski, 2011). While the state of the art algorithms exists in the web search domain, there hardly exist any personalized systems which tackle the research collaboration domain. We leverage the previous work in search personalization and include in the proposed framework a module which provides personalization in the collaboration domain of scientists. Our module is unique in the sense that it not only builds a profile based on the publication records of the researcher but also takes into account the previous searches made by him.
Chapter 3: Introduction to Discovery Tool

In this section, we give a brief outline of the prototype implementation of Discovery Tool. Discovery Tool is designed to make The Ohio State University’s research and creative output transparent, linked, and functional. Discovery Tool acts as a Spring MVC based interactive web tool which helps the user to identify relevant and practical information within the community. Discovery Tool provides a simple and intuitive user interface which is designed by taking HCI factors into account. Discovery Tool provides users an easy way to explore quality data and is based upon published materials such as journal articles, patents, art work, and invention disclosures. The underlying data is drawn from a number of sources, including those data being compiled for use in Research In View, as well as by Academic Analytics and others shown in Figure 3. The Discovery Tool is available to The Ohio State website and the Battelle Memorial Institute. For the foreseeable future, full data access is limited to those users with URLs located at these three institutions.
Figure 3: Example Datasets of Discovery Tool

Goals of Discovery Tool

The main goals of the Discovery Tool are to provide:

- A simple interface through which users can query past published assets
- A system which will ease the expert finding problem
- A system which will provide efficient visualizations to visualize vast amount of academic data
- A modular and adaptable architecture for a collaboration search system
- A prototype interface which will help in understanding the future needs for developing a system with collaboration support
Challenges

There are certain inherent challenges in the design of an academic search system. Below we highlight the challenges while developing such a system.

What is an “expert”?

It seems to be a philosophical question of sorts, but as we set out in this project, how will we define the quintessential expert? The initial idea is to develop solid metrics based on measureable data within the environment, but also look at the organization’s social structure as well when making the recommendation.

Discovery Tool introduces the concept of profiles which acts as CV for researchers. A user profile consists of his publications and his reputation can be easily defined based on the citations he has received for his publications. We adopt the use of citation indices such as the h-index (JE, 2005) and the g-index (Egghe, 2006) to describe an individual researcher. Discovery Tool ranks experts in a field based on his or her citation indexes.

How will we define relationships?

Since most of the data is unstructured text, it might be quite difficult to process the data properly and succinctly. We will have to draw out logical linkages based on some simple rules in order to create the relationships that we want.
Discovery Tool uses the widely accepted concept of author networks to establish relationships in the data.

**How will we extrapolate meaningful connections from data?**

In addition to the above challenge, we need to further extrapolate the valuable connections from the data. In the initial step, we just wanted relationships, but in the end we need to find the relationships that will matter to the user in a real-world sense.

Discovery Tool provides an efficient way to extract author networks such as the co-author networks, friends of friend’s network and referenced author network. These three features combined provide an excellent way to document relationships in the publications database provided by datasets such as Scopus.

**Will this be in real-time?**

Another challenge is that will the system be accumulating this data and inferred understanding in real-time? It seems to point to a partial real-time/cached system from an initial viewpoint. We anticipate an absolute mountain of data that will have to be sifted through in order to construct profiles, so retaining information is a probable direction at this point.

Current architecture of Discovery Tool provides a mechanism for storing vast publications database from well-defined datasets such as Scopus. It has provisions for online querying of these datasets.
How to visualize the results succinctly?

Finally, we have the challenge of visualization. We will want to succinctly visualize data in a way that is meaningful, but not so fanciful that it gets in the way of rendering quickly or analyzing the content to be displayed. This is something that will be a significant ongoing technical challenge for us.

Discovery Tool provides efficient visualizations for visualizing the author’s academic networks and his research interests.

How do we tackle name ambiguities?

In a large bibliographic database, an author can have multiple names and many authors can have the same name. This ambiguity affects the quality of information retrieval and severely degrades the efficiency of academic search engines. It also leads to erroneous distribution of credits.

Since the primary goal of the Discovery Tool was to find experts across the institution, we assume that the dataset provided by Scopus contains minimum ambiguities.

Tools and Technologies used

The list below summarizes the vendor technologies used in the implementation of the Discovery Tool.

a) Vendor: SpringSource
Technology: SpringSource MVC

Value Provided: We use the spring source tool suite IDE for the development of the Discovery Tool. Spring provides a modular MVC framework for efficient development of a web based tool.

b) Vendor: Pentaho (Madan Sheina, 2010)

Technology: Kettle

Value Provided: Kettle is used for ETL purposes. The raw unstructured data from Scopus is converted to an RDBMS format using the open source ETL tool kettle.

c) Vendor: Red Hat

Technology: JBoss

Value Provided: It serves as an application server which hosts the Discovery Tool.

d) Vendor: OSU

Technology: KMdata

Value Provided: Provides a publication dataset complete for the Ohio State University.

e) Vendor: Scopus (Scopus in detail: What does it cover?, 2013)

Technology: Publication dataset

Value Provided: Scopus serves as the primary dataset to the Discovery Tool.
Chapter 4: Discovery Tool - System Model, Design and Analysis

In this section, we present the modeling, design and architecture details of the Discovery Tool.

User scenarios

In order to grasp the value to be gained by providing the innovation, we need to enumerate several scenarios which quantify the benefit in detail. At the high level, we list three key user scenarios of the system. These scenarios summarize the role of the three primary users as listed below:

Scenario 1: Ohio State University Faculty

Summary: In this case, a professor/researcher is searching for collaborators for his upcoming research project. He uses keywords and/or names and sends that data into the user interface portion of the software. It subsequently performs a profile search and/or keyword search and renders possible matches based on his reputation and current collaboration connections. Collaborators that are closer in scope are much better than ones that are totally unknown; therefore they come in higher in the search results. This is a feature of being “known” to the system. When one is known to the system, it can
provide a much higher quality output because the results are based upon pre-accumulated information.

![Diagram: Use Case 1: Professor PI Search]

**Figure 4: User Scenario 1 - Professor (PI) using the Expert Finder system**

The result is that the professor finds his possible matches and has his or her next steps and is ready to proceed unlike before where the Discovery Tool was not available. In the end, he saved months of time that he can now use for research, and this is where the key value of the newly implemented system lies.

**Summary of Operation:**

Input - Input a keyword / key phrase

Output - Names of people who are working in that area

Future Implementation - Profile based / Policy making for collaboration / Impact based search

Visualization Used: Keywords pack diagram/Coauthor Diagram / Tree diagram and Tree map diagram
Scenario 2: Industry partners of Ohio State University

**Summary:** In this case we have a situation in which industry partners are involved. This scenario is portrays these particular partners as being somewhat initially uninvolved in the academic sphere and therefore are unknown to the system. This is a typical assumption although not entirely thorough. Theoretically we could have authors identified with a corporation and retain profile data. However, in this instance let us assume the contrary. Thus, instead of finding collaborators based on research relationship the industry partners want to search based on raw quality of the collaborator.

![Figure 5: User Scenario 2 – Industry Partner using the Expert Finder System to Find Top Experts](image)

The reason for this is instead of having an academic reputation with which to match to another collaborator, industry partners have funding dollars which motivate any PI looking for the next significant project. Therefore, they will seek out the absolute best that the Discovery Tool can give them. This search is especially beneficial for industrial partners because they typically attempt to control costs quite strongly in order to best
utilize resources. Therefore, they want to spend the least amount of time possible on the process.

**Summary of Operation:**

To the system there is currently no distinction between an industry and a professor.

Input: Keyword or a key phrase and/or name of professor in interest

Output: Potential people working in that area

Future: create a profile for industry / policy making for industry profile creation / impact and metrics to be decided for industry profile

**Scenario 3: Incoming Student**

**Summary:** In this case, we have a student who is either interested in or already committed to the university. This student is searching in an unfamiliar organization and the system does not know them either.

![Figure 6: User Scenario 3- Prospective or Currently Enrolled Student is searching for an Advisor](image)

Just like the previous example that is somewhat outside the targeted range for the system (at least in the case of the industry partner not having a profile); the prospective student
case is one that is not of incredibly high value. It is also one where the system is not going to be able to fully flex its relational muscles because it has no idea who they are. However, it does provide a valuable service to the incoming student that is merely trying to get his bearings in a new environment.

**Summary of Operation:**

Input: Keywords / key phrases

Output: Professors and PhD students working in that area

Visualization: same as the above for each name given / no current support

**System Architecture**

Discovery Tool is designed based on a unique modular architecture which we describe in the coming sections.

Based on the user scenarios and preliminary requirements analysis, we present the system architecture for the Discovery Tool. The high level system architecture can be viewed as presented in the figure 7:
In the system, the functionality can be divided into three distinct parts.

**Profile Building**

Data is acquired and accumulated for each registered principal investigator that we find through several avenues of characterizing the data. As the profile is built automatically and stored, we will be able to reference this accumulated data when the user searches for a particular research area or individual. Think of this data as summarizing the raw unrefined data previously existing only in unstructured and untrusted data sources.
Expert Finding

This function will functionally accomplish the aggregation functions of the profile data. Essentially it will assist in summarizing it by using particular metrics created by our project team. Additionally, the component will draw out relationships between individuals’ data that are stored in the profiles.

Visualization

Finally, we need to have a functional piece that will facilitate rendering this complexly linked data to the user in a succinct way. This portion of our solution will essentially take a stream of simplified data and show it to the user in a number of ways that they specify. Of course, the option will always be there to see more detailed data, but we assume a high level starting point.

Data Sources

For all practical purposes, Scopus serves as the primary source of publication dataset which is used by the Discovery Tool. However, the modular MVC structure of the Discovery Tool makes it easy to seamlessly transfer to other data publication data sources such as research in view dataset. The publications dataset which we extract from the Scopus is then transferred to a RDBMS format using the open source ETL tool, Kettle. The primary fields of the dataset are:
In order for better modeling of the system, we created the widely used functional modeling technique, IDEF. Specifically, we create the IDEF 0 (Systems Engineering Fundamentals, 2001) and IDEF 1 (Integration Definition for Information Modeling (IDEFIX), 1993) model for the system and its individual components.

Below we present the IDEF 0 and IDEF 1 models of the system components and the system as a whole.
Figure 8: IDEF 0 model for Discovery Tool - Expert Finding System
Figure 9: IDEF 1 model for Discovery Tool - Expert Finding System
Figure 10: IDEF 0 Model for Profile Building Component
Figure 11: IDEF 1 Model for Profile Building Component
Figure 12: IDEF 0 Model for Expert Finding Component
Figure 13: IDEF 1 model for expert finding component
Chapter 5: Discovery Tool: Methods and Implementation

In order to understand the interaction among the system components of the Discovery Tool, we provide the software interaction diagrams below.

Figure 14: Software Interaction Diagram; Use Case 1- Search Author
We now present the implementation details of individual system components.

**Profile Building Component**

A researcher’s academic profile can be accurately summarized by analyzing his publication records. Past literature suggests the use of citation indices such as the H and
the G index to quantify the impact of a particular researcher in his field (JE, 2005) (Egghe, 2006). Though these indices give us a measure of a researcher’s impact in the field, it hardly serves our collaboration needs. Hence, to cater to the collaboration requirements, we introduce the concept of author profiles. In Discovery Tool architecture, the author profile is a combination of the author’s citation indices, his publication records, his academic networks such as the co-author network, referenced author network, friends of friend’s network and his research interests. In the Discovery Tool architecture, the profile building component is responsible for author profile management and is one of the major components of the system. It consists of smaller modules as shown in the figure 16.

![Profile Building Component](image)

**Figure 16: Profile Building Component**

Below we list the details of individual sub modules of the profile building component.
H-index calculator

One way of quantifying the author’s impact in his field of research is to systematically analyze the citations he or she has received for his or her publications. Hirsch proposed one such index called as the h-index (JE, 2005). The H-index or the Hirsch index tries to quantitatively measure the quality of research and impact of a scientist. It is based on the citations of the publications of the author. It is calculated based on the most cited publication of the author along with citations he has received for his other publications. More formally, an author is said to have an H-index of ‘h’, if he/she has received at least ‘h’ citations for ‘h’ of his publications. (JE, 2005)

Figure below shows how the h-index is computed. X-axis shows the papers of a fictional researcher ‘A’ and the Y-axis shows the citations he has received per paper.
We leverage the h-index as one of the metric to measure the quality of research and quantifying the productivity of the researcher. The use of h-index to capture the impact of a scientist is not new and has been adopted widely in the literature. We follow a similar approach. The h-index sub module is implemented as a spring MVC based service in the Discovery Tool architecture. This module is responsible for calculating the h-index of each individual author. The block diagram of h-index calculator is as shown in the figure 18.

![h-index calculation diagram](image)

Figure 17: H-Index Calculation
**G-index calculator**

H-index serves as a decent metric for measuring the impact. However, it was shown by Egghe that it has limitation (Egghe, 2006). Egghe proposed a new citation based metric called as the g-index (Egghe, 2006) which was an improvement over the h-index. In most systems h-index and g-index serves as default indices for quantifying scientific productivity.

Similar to the h-index the g-index is based on author’s total citations. More formally, assuming a list of publications sorted in descending order of individual citations, an author has a g-index ‘g’ such that the first ‘g’ articles have a cumulative citation of ‘g²’. (Egghe, 2006)

The g-index sub module is implemented as a spring MVC based service in the Discovery Tool architecture. This module is responsible for calculating the g-index of each author. The block diagram for G-index calculator is as shown in the figure 19.

![Block Diagram for H-index Calculator]

Figure 18: Block Diagram for H-index Calculator
**Author Network extractor**

Author network extractor is responsible for the extraction of the coauthors, referenced authors and friends of friend’s networks of the queried author. Two authors are said to be coauthors if and only if there exists at least one joint publication between them. An author ‘X’ is said to be referenced by author ‘Y’ if and only if there exists at least one publication of author ‘Y’ which refers a publication of author ‘X’. An author ‘X’ is considered as a friend of a friend of author ‘Y’ if and only if there exists no joint publication between author ‘X’ and author ‘Y’ and there exists at least one joint publication between author ‘X’ and a co-author of ‘Y’.

Co-author network is defined as the set of all coauthors of the queried author. Similarly, referenced author network and friends of friend’s network is defined as the set of all referenced authors and all friend of a friend of the queried author respectively.

The author network extractor module contains simple routines for extracting the co-author network, the referenced author network and friends of friend’s network of the
queried author. The networks thus formed by this module are then passed to the visualization module.

**Key areas**

The Key Area sub module is responsible for extracting the key areas of an author. In the Discovery Tool architecture, we define the key areas of the author as the most frequent key phrases and key words used in his/her publications. The key areas extractor module first extract all the unigram related to the queried author from the Scopus dataset. These unigram are then sorted based on the the term frequency count. Second, this module extracts all the bigram, trigrams and key phrases of the author from the Scopus dataset and order them based on term frequency analysis. It then maps the key phrases to the unigram. To better understand this process let’s take an example. Assume that an author ‘X’ has two publications ‘p1’ and ‘p2’. The publication ‘p1’ has three key phrases “wireless networks”, “wireless sensor networks” and “wireless networks”. The publication ‘p2’ has three key phrases “distributed algorithms”, “distributed networks” and “distributed systems”. We first extract the unigram from these key phrases and order them as per frequency of occurrence. The two most frequent unigram are “distributed” and “wireless”. This serves as the high level area of interest of the author. One can safely say that the author works in the area of “wireless”. The key phrases are then matched to the unigram, thus the first three key phases map to “wireless” and later three phrases map to “distributed”. All together combined serves as the author’s area of research.
Specifically, in the Discovery Tool architecture the keywords are extracted from the author keywords and index keywords field of the publication record. Sophisticated keywords algorithms can be used for extracting the key words from abstract, and we leave this as future work. For all practical purposes, the key words extracted from the above mentioned fields are sufficient to describe the general area of interest of a researcher.

**Expert finding component**

The figure below shows the conceptual framework for the expert finding component of Discovery Tool. The expert finding component acts as one of the major component of the Discovery Tool architecture. As the name suggest, the primary responsibility of the Expert Finding Component is to efficiently search for field experts within the datasets. It relies on the created profiles from the Profile Building Component and searches across profiles to find potential experts in the queried field. The result of the Expert Finding Component is then passed to the visualization component for rendering. The results are ranked as per citation count of the expert.

In the current implementation of the Discovery Tool, we adopt two mechanisms for searching the expert from the dataset.

- Expert based on key areas
- Expert based on Author network
The Visualization Component is one of the high level components of the Discovery Tool architecture. The main responsibility of this component is to effectively visualize the data from the Profile Building Component and the Expert Finding Component. The Profile Building Component and the Expert Finding Component query...
the database and pass it to the Visualization component as shown in the figure 21. Visualization component then converts this data in Java Script Object Notation (Json) and passes it to the D3.js library which in turn is responsible for manipulating the data to render the visualizations.

Figure 21: Schema for Visualization Component

Discovery Tool provides various visualization features for visualizing the profile of an author. Specifically, we provide efficient visualization for the scientific social networks such as the co-Author network, friends of friend’s network and referenced
author network. We also provide visualizations for the author’s research area. Below we list the details of individual visualizations provided by the Discovery Tool.

**Co-Authors**

We adopt the force directed graph representation for visualizing the Co-Authors of an author. In this visualization, the authors are represented as nodes in the graph and the links between the authors represents a documented form of past collaboration (for example, a joint publication between the two authors). D3.js provides efficient ways to fully utilize various features of the graph for instance, the size of the node, the thickness of the links, the color of the nodes and the length of the links. In the current version of the Discovery Tool, we utilize the size of the nodes and the thickness of the length to further convey information to the users. The thickness of the links is directly proportional to the strength of the collaboration between the two authors; the higher the number of joint publications between the two the thicker the link. The size of the nodes is directly proportional to the impact of the author in his field; higher the citation counts of the author the larger the size of his corresponding node. Figure 22 shows an extracted Co-Author graph of an author from the Computer Science Department of the Ohio State University. The center node corresponds to the author and has a fixed size radius, the nodes surrounding the center node corresponds to his Co-Authors.
Figure 22: Co-Authors Graph Visualization of a Computer Science Faculty

Referenced Authors

We adopt the static graph with nodes and links to represent the referenced author network of the author. This visualization helps to analyze the references of the author. Figure 23 shows an example of referenced author network graph obtained from the Discovery Tool for a Computer Science faculty member. As the co-author graph, we can utilize the visualization features such as the size of the node to convey more information.
to the user. In the Discovery Tool implementation, we limit to a static graph and leave the further improvements for the future.

Figure 23: Referenced Authors Network for a Computer Science Faculty

Friend-of-friend

We adopt the multi-level n-ary graph representation for the visualization of the friends of friend’s network. The center node corresponds to the author and the nodes at level ‘i’ represent the author’s friends; the nodes which have a direct connection to the
author have joint publications and form the co-authors of the author. The author is
directly related to the authors at this level by some joint activity in the past (at least one
joint publication). The second level nodes represents that there is a form of indirect
collaboration between the two authors. Figure 24 shows an example of the friends of
friend’s graph for a Computer Science faculty member.

In the current implementation of the Discovery Tool, we limit the graph to two levels.
However, the graph can be extended to any desirable level. We leave the implementation
of the n-level foaf graph for the future.
Figure 24: Friends of Friend’s network for a Computer Science Faculty

*Interactive tree/dendrogram view*

The dendrogram view is used to represent the key areas of the author. The root node of the dendrogram represents the author. The data from the keyword extractor module of the Profile Building component is passed to the dendrogram visualization module. The keyword extractor module extracts all the unigram, bigram, trigrams and the
key phrases related to the queried author from the author keywords and index keywords field of the dataset. The higher level bigram, trigrams and phrases are mapped against the unigram. This unigram to phrase mapping is then passed to the visualization module which renders the dendrogram view for the author keywords. The words at level 1 of the tree represent the unigram and the words at the second level represent the key phrases of the author.

Figure 25 shows an instant of the dendrogram view for a Computer Science Faculty. The words at the \( n^{th} \) level are ordered as per their citation counts that the author received for those words.
Figure 25: Interactive Tree View for a Computer Science Faculty

Pack diagram

Pack diagram serves as an important visualization for the key areas of the author. The inner most circles represent the key phrases ordered based on the citation count. The circles at the level 1 represent the unigram and the circle at the level 0 represent the...
This visualization gives a brief overview of the author’s research area. Since the unigram and phrases are extracted from the publications of the author, it is safe to say that the key areas visualization provides an overview of the researcher’s area of interest. Figure 26 shows an example of a pack diagram for a faculty member at Ohio State University.

---

**Indented tree**

Another key area visualization based on a similar concept as the pack diagram and the tree diagram. Figure 27 shows an example of indented tree visualization for a Computer Science Faculty member.
Similar to the author network visualizations D3.js provides a simple and efficient way to utilize the various visualization features like the color the radius and the link thickness for conveying greater information to the user. We leave this for future implementation of the Discovery Tool.
User Interface

Discovery Tool provides a simple user interface. The complete discussions of the HCI factors to be considered while designing an interface are out of the scope of this thesis. However, we show the screenshots of the Discovery Tool below.

Figure 28: Home Page of Discovery Tool
Figure 29: Author Profile Page of Discovery Tool
Thus, Discovery Tool provides a modular architecture for an efficient academic search tool. We use this tool to gather feedback and build on the collected experiences to propose a framework for collaboration.
Chapter 6: Service Oriented Aspects

Discovery Tool acts as a system with which our primary users access the published datasets. Since the primary users directly interact with the system, they are the source for valuable feedback for the system. In our approach, we use the technique of Service Blueprint (Shostack G. L., 1984) to document the user-system interactions. The technique of Service Blueprint has been proven to be highly effective in the past for a variety of systems and is one of the techniques which are used for service innovation. It is a technique through which we show the various processes within the Discovery Tool. It helps to divide these processes of into various smaller components. Through this technique, we successfully describe the services provided by the Discovery Tool and record the interactions of the primary users (Scientist, Industry partners and students) with the system. Documenting these interactions through the concept of service blueprinting will allow us an easy way to draft usability survey questions.

The process of developing a service blueprint can be divided into 6 steps (Wilson Alan/Zeithaml, 2008):

1. We first identify the process which has to be to blueprinted
2. Second, we identify the list of users/customers who will be accessing or interacting with the particular service
3. Third, we analyze the service as seen from the customer’s perspective
4. Fourth, we analyze the interactions and actions of the system/user-system interaction points
5. Fifth, we systematically link the interaction points with the appropriate support activities
6. Finally, we document the service for each customer interaction step

In this section, we first outline the moments of truth for the prototype Discovery Tool and then give a total service offer of the system. We then outline the service blueprint for the Discovery Tool following the steps mentioned above. This will help us to understand the interaction of our users with the system.

**Moments of Truth**

Moment of truth is defined as the point where the user interacts with the system. In the case of the Discovery Tool, our primary users will first interact with the Discovery Tool home page. The home page serves as the starting point of contact with the system. The user will also come in contact with other pages as and when he is surfing through the system. All these points then become the moments of truth for the Discovery Tool website. Moments of truth for the Discovery Tool are as shown in the figure 31:
Figure 31: Moments of truth

**Total Service Offer**

Discovery Tool provides a variety of services to its primary users. A user can search for an expert; he can track a competitor and provide feedback. A user can also use all the visualization services provided by the Discovery Tool. It is beneficial to list down all the services. This will help us to draft an efficient survey system for the Tool. In the service domain, the total service offer provides an easy way to list all the core, support and related services of the system. We adopt this technique for listing the key services provided by the Discovery Tool. The total service offer of the Discovery Tool is as shown in the figure 32:
Service Blueprints

In this section, we provide the service blueprints for the core services of the Discovery Tool. In Figure 33, we present the service blueprints of the three primary services of the Discovery Tool: Track author, Find Expert and View Results.
Figure 33: service Blueprint

Figure 34 provides the service blueprints for the additional features seek recommendations and view results included in the proposed framework of Discovery Tool.
The concept of service blueprint provides keen insights in the formulation of the user surveys and evaluation criteria of the system. We provide the details of the survey results in the following section.
Chapter 7: Usability Survey

Primary users of the Discovery Tool, Professors, industry partners and students form an excellent pool of customers of the system. Based on the Line of interaction, we formulated a variety of questions for rating the effectiveness of the system in terms of collaboration needs, the user friendliness of the system, the quality of the output and completeness of the system. The exact questions are provided at the end of the document. We summarize the main results from the survey below.

Primary User feedback

User feedback was acquired from 18 primary users of the Discovery Tool. The users were asked to rate various aspects of the system on a score of 1 to 10. The results are summarized below, Y-axis represents the score and the X-axis represents the users. Figure 35 shows that almost all the visualization features were well received by the user with an average score of more than 7. Pack diagram was rated the highest by the users. Pack diagram provides a quick overview of the author’s research area and is a unique feature of the Discovery Tool. Hence the highest ratings were earned by this feature.
The author network graph gives the user a documented form of past collaboration. As per our survey, most authors follow the friend of friend approach while looking for collaborators. Hence, in figure 36 the co-author network and friends of friend’s graph were rated the most useful for collaboration needs.
Figure 36: Ratings for Usefulness of Visualization Features in Collaboration

We introduce the concept of author profiles while describing the Discovery Tool architecture. In order to enhance the author profiles, we need to ask the primary users for the missing components in the profiles of an author. To this effect, we asked the primary users to rate the completeness of the author profile. Figure 37 shows that almost all the users rated the completeness to be very high indicating that the author profiles provided by the Discovery Tool indeed represents the complete academic profiles.
The users were also asked to rate the user friendliness of the Tool. In figure 38, we see that almost all the users rated it high. The average score was higher than 8. This indicated that the User Interface of the Discovery Tool is simple and intuitive.
The above-mentioned survey questions serve as a good form of user validation of the Discovery Tool system.

**Ohio State Faculty**

One common critique of the Discovery Tool suggested that though the Discovery Tool provided an expert finding mechanism, it performed poorly when the users were in need of a collaborator. In order to add collaboration support, we surveyed the faculty members from the Ohio State University to get a direct documentation of the current collaboration techniques. Table 3 summarizes the results of the survey. We interviewed 5 members across the university to get an overview of the factors that affect the formation of an academic collaboration.
Various criteria such as the author networks, research area and the impact of the researcher in the field were used in the survey. Almost all the faculty members suggested that the most preferable method of collaboration was to search for a field expert among their friends network. To our surprise the citation index of the faculty was not a major criterion for collaboration among faculty members at the Ohio State University.

Based on this survey, we propose our final framework for accelerating collaboration in the next section.

<table>
<thead>
<tr>
<th>Feature</th>
<th>Member 1</th>
<th>Member 2</th>
<th>Member 3</th>
<th>Member 4</th>
<th>Member 5</th>
</tr>
</thead>
<tbody>
<tr>
<td>Citation Index</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
</tr>
<tr>
<td>People that cite you</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>May be</td>
</tr>
<tr>
<td>People whom you cite</td>
<td>Yes</td>
<td>Yes</td>
<td>No</td>
<td>Yes</td>
<td>May be</td>
</tr>
<tr>
<td>Friends</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
</tbody>
</table>

Table 3: Ohio State Faculty
Chapter 8: Proposed Framework for Accelerating Collaboration

Based on the collected feedback and critiques of the Discovery Tool, we present the framework which will foster academic collaboration. The surveys of the Discovery Tool showed that major components of the Discovery Tool provided value and were consistently rated high in terms of usefulness and ease of use. Thus, we keep most portions of the Discovery Tool intact providing minor modifications to the base architecture and add extra modules which will help in the collaboration process. Specifically, we replace the Expert Finding Component with the Active Recommendations Component as shown in the figure below.

Expert Finding Component served the prime purpose of searching for expert and provided to be of excellent value to our industry partners. However, it hardly served our use case of finding potential collaborators. The expert finding component ranks the expert as per citation counts of the authors, which was proven to be little value. The faculty survey clearly indicated that the citation count does not affect the collaborator search. Thus, though the expert finding component served two out of three user scenarios, it hardly was of value for our first user scenario. Thus, here we propose to fix the problem of finding potential collaborators by replacing the expert finding component with the active recommendations component. Active
recommendations component will be contain a list of algorithms which are tailored for solving the collaborator search problem. These algorithms can be devised from the faculty survey documented in the previous section. We propose rule based collaborator search to be implemented using the active recommendations component. These rules will be extracted from faculty interviews. Expert finding tool will serve as the most basic algorithm were result is ranks based on citation count. Figure 39 presents the modified architecture for a collaboration friendly Discovery Tool.

Figure 39: Proposed Framework

User interaction with the system provides great insights about his or her needs. For example, we safely assume that a person who has search for the key words “sensor networks” repeatedly is interested in that field and is looking for some potential collaborators in the field of “sensor networks”. We use this to further enhance the Discovery Tool architecture, and add a component for tracking the user’s network
browsing patterns. We call this module as the history module. The learning from the history module is loop back to the profile building component to further enhance the user profiles. The modified architecture with the History module is presented in figure 40.

Figure 40: Framework Augmented with History Module

The history module is responsible for tracking the previous searches and browsing patterns of each individual. Specifically, we keep a track of previously searched authors, keywords, publications and citation counts. As shown by the faculty survey, these four elements suffice to form the users search profile. The history module is shown in the figure 41.
Figure 41: The History Module

In the survey, the visualization features were proven to be of significant value addition in the collaborator search process. However, we did not utilize the full potential of the all the visual features. Full utilization of all the features will convey more information to the user and will provide better value. One of the main improvements will be to use color codes in the author network visualization.

A researchers’ area of interest is defined by the keywords in his publications and the keywords in the publications that he has referenced. The most straightforward way of defining the researcher’s span will be analyzing all the theses keywords. However, this is an expensive approach. A researcher can also be defined in a succinct manner by his and his co-authors affiliations. In order to define and visualize the span area of a particular researcher, we need to map the researcher affiliation with a unique code which accepted widely in the nation. We also need to map the researcher’s co-authors affiliations to the same code.
This in effect boils down to mapping every individual researcher’s affiliation in the University to a unique standard code. Classification of Instructional Programs (CIP) provides one such list of standard codes. We can use algorithms which uses several text similarity metrics to match the affiliations extracted from Scopus dataset to a unique code in CIP. These codes will directly map to a pre-defined color code which can used in the author network visualizations. The approach can be summarized in the figure 42.

Figure 42: Augmenting the Visualizations
Chapter 9: Conclusion and Future Work

This thesis presents Discovery Tool, a generalized architecture which helps in the development of features of an academic collaboration system. Through a prototype implementation and user feedback analysis, we show that a simple search system is not enough for solving the problem of finding collaborators across an institution. Further analyzing the user feedback and user system interaction, we show what architectural modules are essential for developing a system that will accelerate the academic collaboration process. We show that author networks such as co-author network, friends of friend’s network and referenced author network and research interest of authors are the primary factors which affect the formation of academic collaborations. We provide architecture that helps to extract and analyze these networks.
Bibliography


Types of modeling diagrams. (2008).


Sciences, A. L. (n.d.).


Appendix: Survey Questions

Survey Questions asked to primary users of Discovery Tool

1. On a scale of 1-10 how would you rate the following features (how useful do you think the features are)

   a. Co-author graph
   b. Referenced author graph
   c. Friend of a friend graph
   d. Indented tree
   e. Dendrogram view
   f. Pack diagram

   What is the best visualization for the research area?

   g. Indented tree
   h. Tree interactive
   i. Pack diagram

2. How would you improve the following feature?

   a. Co-author graph
   b. Referenced author graph
   c. Friend of a friend graph
3. What features did you like the most?
4. On a scale of 1-10 how would you rate the usefulness of the visualization feature for searching collaborators?
   a. Co-author graph
   b. Referenced author graph
   c. Friend of a friend graph
   d. Indented tree
   e. Interactive tree view
   f. Pack diagram
5. On a scale from 1-10 rate the completeness of the profile page?
6. On a scale from 1-10 rate the user friendliness of the front end?
7. What was the most striking feature that you think was missing in the system?
8. What was the most striking feature that you think was useful for finding collaborators?

Survey Questions asked to Ohio State Faculty Members

1. What algorithm do you follow to find potential collaborations?
2. How successful were your previous collaborations?
3. List the reasons where the collaboration did not produce the intended results

4. List the reasons were the collaborations were useful

5. How often do you look for collaborators?

6. Do you prefer to work with people having high citation index?

7. Would you like to work with people who you usually cite? Why? Why not?

8. Would you like to work with people who commonly cite you? Why? Why not?

9. Would you like to work with friends of friends? Till what level of the graph would you prefer to search?

10. What selection criteria do you use while selecting collaborators?

11. List the following criteria in ascending order of importance

   a. Co-authors as collaborators
   b. Authors you cite as collaborators
   c. Authors that cite you as collaborators
   d. Friends of friends as collaborators
   e. Geographically far versus near
   f. Authors citation index
   g. Research area of the potential collaborators (highly reputed in that field/new comers)

12. Would you like to change the answer to the first question?