A Thesis for the Degree of Master

An Interactive Information Seeking Interface for Exploratory Search

Hogun Park
School of Engineering
Information and Communications University
2008
An Interactive Information Seeking Interface for Exploratory Search
An Interactive Information Seeking Interface for Exploratory Search

Hogun Park

We certify that this work has passed the scholastic standards requested by the Information and Communications University as a thesis for the degree of Master of Science

July 3, 2008

Approved by:

Chairman of the Committee
Sung-Hyon Myaeng, Professor
School of Engineering

__________________________
Committee Member
Geehuck Lee, Associate Professor
School of Engineering

__________________________
Committee Member
In-Young Ko, Assistant Professor
School of Engineering
Abstract

As the Web has become a commodity, it is used for a variety of purposes and tasks that may require a great deal of cognitive efforts. However, most search engines developed for the Web provide users with only searching and browsing capabilities, leaving all the burdens of manipulating information objects to the users. In this thesis, we focus on an exploratory search task and propose an underlying framework for human-Web interactions. Based on the framework, we designed and implemented a new information seeking interface that helps users reduce cognitive burden. The new human-Web interface provides a personal workspace that can be created and manipulated cooperatively with the system, which helps the user conceptualize his information seeking tasks and record their trails for future uses. This interaction tool has been tested for its efficacy as an aid for exploratory search.
# Table of Contents

Abstract ........................................................................................................................................ 1

List of Tables ................................................................................................................................ 1

List of Figures ................................................................................................................................. 1

List of Abbreviations ....................................................................................................................... 1

I Introduction ................................................................................................................................. 1

1.1 Motivation for the Proposed Approach .................................................................................. 1

1.2 Contribution ............................................................................................................................ 3

1.3 Thesis Organization .................................................................................................................. 6

II Background ................................................................................................................................. 7

2.1 Exploratory Search ................................................................................................................... 7

2.1.1 Finding Important Scenarios .............................................................................................. 8

2.1.2 Designing New Interfaces .................................................................................................. 9

2.1.3 Evaluating Exploratory Searching ..................................................................................... 9

2.2 Seamless Integration among Information Seeking Activities ............................................... 10

2.3 Reducing Cognitive Burdens in Information Seeking ........................................................... 12

2.4 Information Seeking Interface ............................................................................................... 14

III Underlying Model ..................................................................................................................... 17

3.1 Formalization .......................................................................................................................... 20

3.1.1 Operations for Knowledge Space Organization ................................................................. 22

3.1.2 Inter-Space Operations ....................................................................................................... 23

3.2 Examples .................................................................................................................................. 25

IV Interaction Framework ............................................................................................................. 28

4.1 Personal Workspace ............................................................................................................... 29
4.2 Interaction Mediation ................................................................. 30
  4.2.1 Path Recommendation .......................................................... 30
  4.2.2 Search and Navigation Trails ............................................... 38
  4.2.3 Topic Extraction and Association Recommendation ............... 39
  4.2.4 Other Components ............................................................. 43

V Implementation ........................................................................... 45
  5.1 Key Design Goal ...................................................................... 45
  5.2 Implementation Details ........................................................... 46
    5.2.1 Authoring Knowledge Objects in Workspace ....................... 46
    5.2.2 Saving and Loading Workspace .......................................... 48

VI Experiment ................................................................................. 50
  6.1 Guided Navigation .................................................................. 51
    6.1.1 Experiment Design and Result ............................................ 51
  6.2 Reducing Workload .................................................................. 52
  6.3 Information Reuse .................................................................... 54
    6.3.1 Experiment Design ............................................................... 55
    6.3.2 Result and Analysis ............................................................ 57
  6.3 Subjective Comments .............................................................. 61

VII Conclusions .............................................................................. 63
  7.1 Contributions .......................................................................... 63
  7.2 Future Work ............................................................................ 64

국문요약 ......................................................................................... 65

References ...................................................................................... 68

Acknowledgements ......................................................................... 74

Curriculum Vitae ............................................................................. 75
List of Tables

Table 1. A List of Operations in the Two-level .................................................. 18
Table 2. A Weight Matrix of Example Network ............................................... 36
Table 3. The Result of the First Experiment .................................................... 52
Table 4. The Result of SWAT ................................................................. 54
Table 5. ANOVA Result ............................................................................. 60
Table 6. Pairwise Comparisons ................................................................... 60
Table 7. Survey Results ............................................................................... 61
List of Figures

Figure 1. An Example Screen Shot of sketchBrain ............................................. 4
Figure 2. A Conceptual Overview of the Two-level Model .................................. 17
Figure 3. sketchBrain Interaction Framework ...................................................... 28
Figure 4. An Example Screen Shot of sketchBrain ............................................... 29
Figure 5. Classification of User Participation ....................................................... 34
Figure 6. An Example Network ............................................................................. 36
Figure 7. A Snapshot of the Navigation Trails ...................................................... 38
Figure 8. A Snapshot of the Query Trails .............................................................. 39
Figure 9. Accuracy of the Automatically Generated Association Names ............... 42
Figure 10. Adding a Topic on the Workspace-before .......................................... 46
Figure 11. Adding a Topic on the Workspace-after ............................................. 47
Figure 12. Associating Two Topics on the Workspace ......................................... 47
Figure 13. Making a New Session .......................................................... ............. 48
Figure 14. Saving a Workspace ............................................................................ 49
Figure 15. Experimental Design for Each Group ................................................. 56
Figure 16. Comparison for Task Completion Time .............................................. 57
Figure 17. Comparison by Different Tasks ............................................................ 58
List of Abbreviations

<table>
<thead>
<tr>
<th>Abbreviation</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>API</td>
<td>Application Program Interface</td>
</tr>
<tr>
<td>ICU</td>
<td>Information and Communications University</td>
</tr>
<tr>
<td>SIS</td>
<td>Stuff I’ve Seen (Microsoft Desktop Indexing Engine)</td>
</tr>
<tr>
<td>TMSA</td>
<td>Time-variant Spreading Activation</td>
</tr>
<tr>
<td>ANOVA</td>
<td>Analysis of Variance</td>
</tr>
<tr>
<td>SWAT</td>
<td>Subjective Workload Assessment Technique</td>
</tr>
</tbody>
</table>
I Introduction

1.1 Motivation for the Proposed Approach

For a traditional Web search engine, the process of querying and viewing the results is usually regarded as a single, isolated session that ends in itself. As the Web has become a commodity, however, it is used for a variety of tasks in many different ways, encouraging new paradigms in information seeking (e.g. berry-picking [2], information foraging [36], and sense-making [39]). However, most popular commercial search engines have taken a conservative position and adhered to the traditional model, leaving all the rest of the information seeking and related tasks to the user. More specifically, the user has all the burdens of manipulating the information objects that have come to his attention in a series of search activities.

An area in which this type of cognitive burden affects significantly is exploratory search. An exploratory search task ([28], [50]) is to investigate on the background information of a topic or gather information sufficient to make an informed decision. For example, assume that a user is considering purchasing a DMB (digital multimedia broadcasting) receiver. The user would want to learn more about the DMB technology and the manufacturers of various products related to it, so that he can select the provider and the products that best suit the needs.

We believe that most existing search engines and their interfaces are not satisfactory for exploratory tasks, because of the following.
1. Cognitive burden: Compared to the task of searching for specific or known items, an exploratory search task usually requires users to send a series of queries during a search session, visit more new domains, and revisit previously visited sites (especially branch pages) [50]. These activities together mean a significant amount of information and workload that traditional search engines have rarely attempted to reduce. The workload is associated with representing information needs [45], determining informativeness [46], and memorizing previously explored information ([7], [31], [44]). Without an explicit support from a search engine, the difficulties resulting from the workload are left as a cognitive burden to the user.

2. Narrow interaction channel for incorporating user interests: In an exploratory search, a user needs to build up background information on a topic gradually until she feels that a sufficient amount of information has been gathered for the given task. As such, it is important to incorporate the users’ interest and the information that has been found as the system processes the current query. However, current search systems rarely support the notion of “session” and interactions explicitly. While the one-time query/result model is simple and natural with HTTP, it ignores what has been done by the user in her attempt to change her anomalous state of knowledge [4]. Although there have been some attempts to infer user interest explicitly ([30], [16], [14]), implicitly ([8], [10], [40], [43]), or both [53], the problem remains challenging, especially within the context of user-system interactions.
Given the limitations of traditional search engines for an open-ended, exploratory search task, we propose a new interaction tool that can provide an interface between a user and a search engine, called sketchBrain. Our aim is to provide an effective interaction environment that facilitates the series of activities in an exploratory search of the Web.

1.2 Contribution

There are several novel features in this interaction environment. First of all, sketchBrain keeps track of query trails and post-query navigation trails (based on the click stream following the issued queries) and allows the users to conceptualize them. For an information seeking activity, a trail is sketched on the user’s workspace of sketchBrain. Over the trail, the user can associate user-defined topics and system-provided semantic associations between topics using the annotation facility in sketchBrain. These annotations together with the information items and queries are key objects in the underlying model. In essence, the workspace serves as a rich memory for the past and current search efforts, which can be accessed later.

Second, our interaction tool is equipped with operations on the objects created and manipulated in the workspace. In addition to the annotation facility, sketchBrain allows users to manipulate the objects for their information seeking tasks. Implicit operations such as project, select, and classification (to be described in Section III) can be utilized for the activities necessary for an exploratory search.
Third, *sketchBrain* has an intelligent path recommendation algorithm that can help users choose the most promising page to be explored at the next step in navigation. It assists users in determining informativeness of the pages that can be explored at the next step quickly. Fig. 1 shows all the suggested pages whose colours are on the spectrum between yellow and red. The user is currently visiting the page regarding “Microsoft Windows,” the blue node at the center, and about to choose one path from the available paths surrounding the node representing the current visit. The expected degree of relevance is determined by the algorithm and is shown in various colours (The degree of darkness stands for its relevance)

![Figure 1. An Example Screen Shot of sketchBrain](image-url)
This new interactive tool, sketchBrain, is based on an underlying interaction model, called two-level model. It is based on the recognition that two spaces are involved in human-Web interactions: information space and knowledge space. The information space is essentially the Web itself, containing the information objects (e.g. Web pages), whereas the knowledge space is superimposed on the information space to contain a user’s conceptual understanding of information objects and their relations. Concepts in the user’s mind, emerged by reading Web pages at the information level, are expressed as topics and their relationships (or associations) in the knowledge space. Topics are also connected to information objects, called occurrences, which can be represented by them. For convenience, an occurrence can be of any granularity, including a page, a figure, or a phrase in a document. The connection between a topic and an occurrence provides a way to establish a link between the two spaces.

The three terms, topics, associations, and occurrences, are borrowed from the Topic Maps framework (ISO/IEC 13250). In our information seeking interface, implicit knowledge-level operations performed in user’s mind have been explicated for the purpose of reducing user’s cognitive burdens. In essence, interactions between the user and the system enabled by the two-level model are at the knowledge level as well as at the information level. However, the types of interactions with traditional search engines, in the form of a query and a result, are at the information level.

In sketchBrain, our path recommendation algorithm supports a guided navigation for a successful completion of certain information seeking tasks. Previous approaches to guided navigation ([22], [24], [34]) keep track of and utilize
previous users’ browsing behaviors and queries. Although they propose good ways to suggest paths that match the inferred task on a specific point, they have utilized relatively noisy information such as all previous queries and pages visited. Our approach for supporting guided navigation utilizes both explicit and implicit feedback from the workspace in sketchBrain. All the supports are geared toward exploratory search tasks.

1.3 Thesis Organization

This thesis is organized as follows. In chapter II, a background is introduced. Exploratory search has become a topic of great interest to researchers. We introduce a basic description of it and challenges. Then, we introduce how researchers have attempted to avoid limitations of existing information seeking models. Especially, we note how researchers have tried to make a seamless transition between searching and browsing activities and reduce cognitive burden in information seeking.

Next, we define our underlying model for exploratory search. Based on an existing digital library theory for describing the environment surrounding information objects, we define a new underlying model in Chapter III. It, called two-level model, describes its formalization in connection with the 5s Theory [13] that provides fundamental abstractions in a digital library, so that our work can lead to an extension to the theory.

After that, the rest of the thesis is devoted to the interaction framework for supporting an exploratory search task (Chapter IV), the implementation of the framework (Chapter V), and empirical evaluation via user studies (Chapter VI.)
II Background

2.1 Exploratory Search

As the Web has become a commodity, it is used for a variety of purposes and tasks that may require a great deal of cognitive efforts, such as learning and investigating something, rather than just finding a piece of information. This type of cognitively demanding search is called exploratory search that has been studied in recent years (see 2006 SIGIR Workshop\(^1\) and 2007 SIGCHI Workshop\(^2\)). In exploratory search, it is required for users to spend a certain amount of time in scanning/viewing, analyzing, and making a qualitative/quantitative judgment.

For example, let’s consider a case where a user has a goal to acquire general knowledge about the ancient history of East Asia. During the process of learning, the user would not be interested in getting isolated facts but would want to collect pieces of information of higher granularity from different sources and integrate them to build up general knowledge (e.g., historical background, culture, etc.).

A question may arise as to what efforts people put in for searching tasks without an interaction mediation tool for searching and browsing. We can find interesting habits from a previous observation [60]. First, when people gather background information in a new domain, they usually gather possible resources that could contribute to their main task. As a technique, they can create a blank document and write down useful references on that. At that time, the

\(^1\) http://research.microsoft.com/~ryenw/eess
\(^2\) http://research.microsoft.com/~ryenw/esi/
users may annotate why the references are interacting and what subject it will be related to. After finishing the annotations, people start collecting and comparing the resources. During this process, they make a “meta-model” of the subject, which may contain both session-level grouping and subject-level grouping. Using this model, people finally identify answers in problem-solving sequences and make any conditions to reinforce/rule out the problem. During this work, if the user needs to re-find the same information later, she would need to rely on her memory or bookmarked pages, trying to reconfigure the state of mind she had before with a series of cognitive operations. These features should be employed in an information seeking interface and efficiently supported. However, traditional search engines or browsers are not amenable for assisting users for this type of tasks.

In order to support exploratory search, there are four key challenges: finding important scenarios, designing new interfaces, and evaluating exploratory search interfaces.

2.1.1 Finding Important Scenarios

In addition to research in traditional Information Retrieval (in other words, keyword-based search), one challenge for exploratory search is to further understand the search scenarios for the case when traditional retrieval is not sufficient. For example, assume that a user may not know much about classical music, but he will know what he likes when he hears it. At the time, how should we begin to find a piece that he might like? Nowadays, these kinds of concerns seem like to shift the focus of research toward understanding the behaviors and preferences
of users engaged in exploratory searching. Researchers started to observe their behaviors [50] and cognitive activities in a large-scale search engine and tried to model the searching activities [28][55]. The representative trend is to provide seamless a transition between searching and browsing activities. The attempts in some previous research will be introduced in section 2.2.

2.1.2 Designing New Interfaces

The second challenge for supporting exploratory search is to focus on identifying alternative user interfaces and interaction models that support users in different ways. A good example is faceted search [56][57] which allows selecting a proper perspective. The faceted search and the selection of proper facets (commonly, hierarchical metadata) leverage proper data attributes for search and contribute to find useful data that users may miss. Therefore, the design of interface can be effective for a large-scale search engine and their navigation. This kind of challenge is extended into many recent information retrieval interfaces including faceted browsers [56][57] and workspaces [5][12][14][16][17][30] for organizing search results. The detailed effort for designing new interfaces will be described in section 2.4.

2.1.3 Evaluating Exploratory Searching

In traditional retrieval, there exist many frameworks for evaluation, and especially, the annual NIST-sponsored TREC has provided a medium and corpus from well-defined analytic aspects in terms of Information Retrieval. However, exploratory search is different from the other tasks covered previously. Explora-
tory search tasks require supporting more complex behaviors (including regard-
ing efficient information use and gathering) beyond a simple lookup. Therefore, it is very hard to make a framework to evaluate it under the interactive environment. In addition, because there are many subjective variables such as cognitive burden and stress, most measures often used in Information Retrieval are not very helpful. Moreover, when we design tasks for experiments, a well-specified task may prevent the participants from showing real exploratory behaviors. Therefore, we need to explore further about objective evaluation methods.

2.2 Seamless Integration among Information Seeking Activities

In exploratory search, one of the most important tasks is to provide seamless integration among information seeking activities. Several researchers have tried to provide seamless integration of searching and browsing. Annotating hyperlinks is one method for associating searching and browsing. WBI project [3] initially proposed to annotate hyperlinks in a web page, and ScentTrails [34] tried to annotate the hyperlinks of web pages with graphical and textual representations. While WBI project focused on avoiding network latency, ScentTrails highlights links that match current search criteria. The links are selected by content that matches the search query. The value of the research lies in providing abundant searching cues.

In exploring the contents of object databases, PESTO [6]’s query-in-place was devised to integrate searching and browsing. Query-in-place provides an integrated query paradigm in which a complex object query can be formulated for filtered browsing. This idea affects several other works. For example, it was
adopted into a XML’s hierarchical formula that helps interleaved searching and browsing in the MIX project [32] and BBQ [33]. They showed an interactive information seeking method in the XML framework.

The work known as Walden’s Path is most relevant to ours since it was also targeted at guiding user’s searching and browsing. Walden’s path is a tool that applies meta-data to superimpose a structure over unconnected documents to facilitate their reuse [41]. It provides some paths that help readers’ intuitive understanding of the organized information, resulting in saving of time and effort for the processes between searching and browsing. Although it is meaningful to reuse information of other authors, the author’s additional work is still far from achieving the power of our model of knowledge and information spaces.

There were other attempts to represent knowledge on top of information. In [23] a new representation helps to reduce wasting resources and enhance information integration with an ontology paradigm. They focused on generating topic maps automatically, but there is no interaction with users because their intention was not to guide users for searching and browsing.

Zizi and Beaudouin-Lafon [54] also tried to provide automatically-generated, abstract graphical views in Interactive Dynamic Maps using topic maps. They attempted to develop automatic techniques for building maps and provide an integrated graphical query language. However, it was not developed for a real environment containing text documents.

Leake, Maguitman, and Reichherzer [20] proposed two approaches: DISCERNER and EXTENDER. The former is helpful in organizing concept map libraries. It is made into a hierarchical structure of topics, so it is efficient to
access data in a map. The latter manufactures the topics of concept maps and extracts related topics from the web. They initialized automatic topic-gisting and constructed concept maps at knowledge level but it lacks in inter-space operations and doesn’t provide searching and browsing cues.

The work by Ma and Tanaka [26] tried to provide information retrieval within a content-based join model for data streams and web pages. They extracted topic-structures from the target and utilized them to find other candidates. Their idea is necessary for intelligent searching in the Topic Maps framework but it provides neither browsing cues nor inter-space management.

Most of the works described above can be seen as direct or indirect efforts to integrate searching and browsing by providing intelligent automation, but what seems to be lacking is management through knowledge space, information space, and inter-space operations. Our work’s unique contribution is to give an interactive environment for expressing users’ cognitive activity and to utilize the knowledge for later seamless searching and browsing through interaction mediation.

2.3 Reducing Cognitive Burdens in Information Seeking

Nowadays, the Web is used for a variety of purposes and tasks that may require a great deal of cognitive efforts. However, most information systems (especially, search engines) developed for the Web provide users with only searching and browsing capabilities, leaving all the burdens of manipulating information objects to the users. In order to solve this limitation, there were several approaches to helping cognitive processes in using information systems. The most
popular approach was using assistance such as contextual helps and recommendations, making the searching experience richer and the cognitive load lower.

For effective use of assistance, Bernard J. Jansen [18] categorized 26 patterns of interactions into 9 categories, such as View Results, Selection, View Document, Execute, Navigation, Browser, Relevance Action, View Assistance, and Implementation Assistance, and analyzed the use of each patterns. From that analysis, he developed an application that provides automated assistance.

Gary Marchionini [28] divided search tasks into 3 types: Lookup, Learn, and Investigate. Lookup is just searching the fact and discrete data. Performance of learning is usually affected by mathematical algorithms or machine power. Searching to learn or investigate, however, includes searcher’s cognitive process. He defined the last two types as exploratory search. According to him, more interactive interface makes searcher active, and active searcher’s feedback can improve exploratory search. Related to this work is the development and evaluation of a relation browser that shows the search result with some contextual data facets such as topic, time, space ([27], [28], [29]). With this information, searchers can understand the context of the retrieved lists and easily select what they are looking for.

While above mentioned approaches focus on reducing searcher’s cognitive load, they simply suggest feedback to the user in anticipation of the next steps. However, these assistances are limited in that users do not have their own space where various information or knowledge items can be manipulated for their conceptual understanding of the search results.
Several researchers attempted to help user’s cognitive efforts and comprehension by using Topic Maps. In [9], Topic Maps are used as a tool for helping Web-based education. It shows many advantages on learner’s side and author’s side. Topic Maps created by others can be retrieved easily and used for various learning and authoring purposes. While Topic Maps were created, retrieve, and used by different types of users, this work has nothing to do with general purpose information seeking, in particular exploratory search, which our research is centered around.

Van Hemel, Bert Paepen, and Jan Engelen [48] used Topic Maps for smart search in a news domain. Topics and associations are extracted automatically when a news article is added to the database. The system provides relational navigation and advanced query generation. However, automatic extraction cannot satisfy individual searcher’s specific view.

The research on Topic Maps shows the possibility of helping visualization of some cognitive processes. However, there is no freedom for users to create their own topics and associations, not to mention their own knowledge space. Our approach helps users to express their cognitive process explicitly in searching and browsing by providing an interaction framework with various operations.

2.4 Information Seeking Interface

Various information seeking interfaces also have been proposed to support complex information seeking activities. Sketchtrieve [17] employs Cognitive Dimension Framework to map out the design space and provides an unstructured canvas. In this canvas, searchers can freely represent queries and corresponding
search results with an intuitive interface by using typographic and layout cues that lie outside of a formal notation. Navique [12] and DLITE [5] also suggest new interfaces for information seeking. Navique proposed a zoom-able fractal result navigation system, and DLITE proposed a search GUI that keeps track users’ information seeking activities and provides a space allowing their organization. These three interfaces were new attempts to organize search results, but their effectiveness for reducing cognitive burdens was not clear. In addition, there was no proactive guide for users to explore and navigate the search space.

George et al. [14] introduces information seeking workspace called Garnet. They exploit implicit knowledge that can be discovered from the contents in the workspace and try to find direct connections between the workspace and digital libraries. They utilize spatial parsing to extract profiles of documents and use them to learn a lexical classifier. This classifier is to identify newly searched documents that are relevant to each parsed cluster. Martin and Jose [30] suggest a personal information retrieval tool that employs a folder-like structure, so that searchers can bundle search results into folders. In addition to the interface that searchers can freely organize results, it assists query formulation and recommends hot relevant documents to each folder. Harper and Kelly [16] employ a topical structure for relevance feedback. Their interface allows users to save documents in user-defined piles for similar documents, which could be used for relevance feedback. These approaches suggest new information seeking environments with some assistance. However, their design goals are not to support exploratory search explicitly, and the systems were not tested as such.
Recently, there are several approaches to propose an interactive information seeking interface for supporting exploratory search. The information seeking interface interactively guides users by presenting objects with useful views of information space and showing them the available options like additional query refinements. Because this research field is not mature, a combination of multiple search paradigms has been attempted.

TagSphere [58] was an approach to show a visual presentation of search results. Users make search queries interactively by selecting sample images from suggestions. Search results are also visually presented in the sets that are similar to the query. The similarity calculation was based on distance and overlap between the results of tag search and traditional information retrieval. The result provides users a better understanding with a comprehensive overview. However, their attempt was limited to image searching, and only a list of tags and images were used for querying.

Tvarožek et al. [59] proposed collaborative search to support exploratory search. They developed a facet browser via a combination of keyword-based, view-based, and content-based search paradigms using social collaboration. In the work, although it employs a general faceted browser layout, it provides different personalized views for different types of data and shows a graphical overview of search results based on a hierarchical clustering visualization. Over the visual interface, it allows users to further refine/filter their query by selecting a specific information subspace. They showed an interesting visual interface, but they lacking in supporting future searching/browsing, especially in long-term search. Our information seeking interface is beyond their scope, and it can be described in our underlying model.
III Underlying Model

Fig. 2 depicts a conceptual overview of the relationship between the information and knowledge spaces and the operations. The information space consists of collections of information object instances such as Web pages, documents, images, and so forth. Hyper links and public directories in the Web are also included. Traditional searching and browsing are defined to be operations on this space.

![Figure 2. A Conceptual Overview of the Two-level Model](image)

The knowledge space exists over the information space and contains various virtual links\(^3\) to the information objects on the information space. Although

---

\(^3\) In the terminology of Topic Maps, they are called occurrences.
our tool can make suggestions for topics and their associations from the content of the information objects, a knowledge space is basically created by an individual user based on her interpretation of the information objects and their relationships. It is composed of topics and associations that correspond to the meaning or usage of the information objects and their relationships, respectively. Topic types, association types, and ontology\(^4\) can be included for full semantic operations as in the Topic Maps framework. Table 1 is a brief description about operations in the two-level model.

Table 1. A List of Operations in the Two-level Knowledge Space

<table>
<thead>
<tr>
<th>Knowledge Space Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Creation</strong></td>
</tr>
<tr>
<td>Gen_Topic</td>
</tr>
<tr>
<td>Gen_Assoc</td>
</tr>
<tr>
<td>Gen_Occ</td>
</tr>
<tr>
<td><strong>Deletion</strong></td>
</tr>
<tr>
<td>Del_Topic</td>
</tr>
<tr>
<td>Del_Assoc</td>
</tr>
<tr>
<td>Del_Occ</td>
</tr>
<tr>
<td><strong>Modification</strong></td>
</tr>
<tr>
<td>Mod_Topic</td>
</tr>
<tr>
<td>Mod_Assoc</td>
</tr>
</tbody>
</table>

\(^4\) Ontology is an essential element in semantic Web, but it is not necessary in a personal knowledge space as the user is assumed to have ontology in her mind. Nonetheless, we have defined some operations using ontology.
<table>
<thead>
<tr>
<th>Mod_Occ</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Ontology</strong></td>
</tr>
<tr>
<td>Get_Super, Get_Sib, Get_Sub</td>
</tr>
<tr>
<td><strong>Search</strong></td>
</tr>
<tr>
<td>Knowledge Search</td>
</tr>
<tr>
<td><strong>Inter-space Operations</strong></td>
</tr>
<tr>
<td>Classify</td>
</tr>
<tr>
<td>Select</td>
</tr>
<tr>
<td>Project</td>
</tr>
<tr>
<td>Follow Association</td>
</tr>
<tr>
<td>Backward/Forward</td>
</tr>
<tr>
<td><strong>Information space Operations</strong></td>
</tr>
<tr>
<td>Search</td>
</tr>
</tbody>
</table>
The Select and Project operations are analogous to those in SQL. Select takes a condition such as a Boolean expression or manual selection over topics and returns a set of information objects connected to the selected topics, giving the effect of truly semantic searching. With the Project operation, users can specify the types of the information objects that satisfy other conditions, if any. The type information, such as format (e.g. PDF) or a genre, can be available as an occurrence type in Topic Maps. Note that the first four operation groups are purely at the knowledge level, but the rest (Classify, Select, Project, Follow-Link, Backward, and Forward) are across the knowledge and information spaces; they are conceptualized at the knowledge space, but affect the information space.

### 3.1 Formalization

Our two-level model is compatible with the 5s (Streams, Structures, Spaces, Scenarios, and Societies) theory of digital libraries [13]. At the same time, it can be seen as a way to extend the theory for personalized digital libraries with the notion of knowledge space and associated operations.

The 5s theory proposed the fundamental abstractions of a digital library. *Streams* are the sequence of any contents such as characters, images, video, etc. They can represent both a sequence of static elements and a dynamic information flow. *Structures* are the way of presenting arranged information. They are defined in a tuple of a directed graph and a set of labels, and a labeling function. *Spaces* are a set of various space (vector/metric/.. ) with corresponding opera-
tions. *Scenarios* consist of events or actions that modify states of a computation in order to accomplish a functional requirement. *Societies* are the highest component in a digital library and defined in a set of entities and relationships among them.

Based on these abstractions and the formal definitions, we describe our two-level model and the corresponding operations in a similar way. As described in [13], information space can have a set of collections, $C$, which is a set of digital objects, $do$, a tuple of universally unique handles, a set of *streams*, a set of structural metadata specification (see Definition 11 in [13]), and a set of *structuredStream* (see Definition 15).

In addition to information space available in a traditional digital library, we create the notion of *knowledge space* and define a collection of knowledge spaces, $K$. Following are some primitive notations for knowledge space in conjunction with information space.

- **Definition**: Let $KM = (KS, Contents, P)$ be a knowledge map between the two spaces, where

1) $KS = \{ (T_{KS}, A_{KS}, L_{KS}, F_{KS}) \}$ is a knowledge structure. $T_{KS}$ and $A_{KS}$ represent topics and associations among topics, respectively. Each $A_{KS}$ or $T_{KS}$ has a label, $L_{KS}$, and there is a labeling function $F_{KS}$ on the structure.

2) *Contents* includes a set of digital objects, $do$, in information space, all of their (sub)streams, and all possible *StructuredStream* which associates nodes of a structure with segments of a stream.
$P$ is a matching function with $KS$ and $Contents$. Thus, this function connects Knowledge space with Information space. It is called “Occurrence” in Topic Maps.

3.1.1 Operations for Knowledge Space Organization

• Creation (Gen_Topic, Gen_Assoc, Gen_Occ)

Creating a knowledge space is to make a knowledge map, $KM = (KS, Contents, P)$. $KM$’s core part is to construct topics, associations, and occurrences. Anything that can be conceptualized in a user’s mind can be topics and associations. An occurrence is defined in the mapping function $P$ that associates topics with $Contents$.

• Deletion & Modification

These operations can be managed in a repository control module (see Definition 19 in [13]) that has the functions get, store, del, and modification to manage and access the collections.

• Knowledge_Search

$OP_{knowledge\_search}: Q \times K \rightarrow 2^K$, where $Q$ is a query (see Definition 21 in [13]) and $K$ is knowledge collection. This function associates a subset of $K$ with $q \in Q$ and knowledge object, $ko \in K$. It gets a query and then finds appropriate knowledge objects.
• Ontology-Related Operations

\[ OP_{\text{get\_super\_type}}: T_K \times O \rightarrow O, \text{ where } T_K \text{ is a topic and } O \text{ is an ontology object.} \]
This function associates an ontology object with \( t \in T_K \) to give a topic that is the super type of the given topic.

\[ OP_{\text{get\_sibling}}: T_K \times O \rightarrow O, \text{ where } T_K \text{ is a topic and } O \text{ is an ontology object.} \]
This function associates an ontology object with \( t \in T_K \) to give topics that are the sibling of the given topic.

\[ OP_{\text{get\_sub\_type}}: T_K \times O \rightarrow O, \text{ where } T_K \text{ is a topic and } O \text{ is an ontology object.} \]
This function associates an ontology object with \( t \in T_K \) to give topics that are the sub type of the given topic.

3.1.2 Inter-Space Operations

• Project

\[ OP_{\text{project}}: T_{KS} \times P \rightarrow 2^{\text{Contents}}, \text{ where } T_{KS} \text{ is a topic.} \]
This function associates contents with \( t \in T_{KS} \). It gets a topic from the user and finds the corresponding contents using the designated pointer \( P \).
• **Select**

\[ OP_{select} : 2^{KS} \times P \rightarrow 2^{Contents} \], where \( 2^{KS} \) is a subset of knowledge structure.

This function associates contents with \( t \in T_{KS} \). It gets a topic expression defined by the user, and finds the corresponding contents using all the pointers \( P \) emanating from the topics covered by the topic expression.

• **Classify**

\[ OP_{classify} : T_{KS} \times 2^{Contents} \rightarrow P \], where \( T_{KS} \) forms a classification schema with the topic types.

This function associates a subset of contents, or a set of digital objects, with \( t \in T_{KS} \) and returns a mapping function \( P \) that effectively generates occurrences.

• **Follow_Association**

\[ OP_{follow\_association} : T_{KS} \times A_{KS} \rightarrow 2^{Contents} \], where \( T_{KS} \) is a set of topics and \( A_{KS} \) is an association.

This function returns a subset of contents that are occurrences of the topics \( s \in T_{KS} \) that has an association \( a \in A_{KS} \) with \( t \in T_{KS} \) designated by the user.
• **Forward**

Let \( \mathcal{I} \) be a sequence of KS’s, \( \{KS_0, KS_1, \ldots, KS_n\} \) that have been constructed and saved after each retrieval followed by a series of knowledge-level operations.

\[ OP_{\text{forward}} : \text{succ} (KS_i) \text{ where succ(.) is a successor function that returns the} \]
\[ KS_{i+1} \text{ next to } KS_i \text{ in } \mathcal{I}. \text{ Succ}(KS_n) \text{ is not defined.} \]

• **Backward**

\[ OP_{\text{backward}} : \text{pred} (KS_i) \text{ where pred(.) is a predecessor function that returns} \]
\[ KS_{i-1} \text{ prior to } KS_i \text{ in } \mathcal{I}. \text{ Pred}(KS_0) \text{ is not defined.} \]

### 3.2 Examples

The following two scenarios contain examples of how various operations provided by the tool are used to accomplish a series of information seeking tasks.

**Scenario 1**: In a traditional Web search situation, a user issues a query, “laser ink printer” to find pages about laser printers (Search). Realizing that laser ink printers are different from laser printers (Browse), he modifies the query (Search). He reads through the first few lists of returned results (Browse) and marks relevant pages with the topic “laser printer” (Gen_Topic) and save them (Gen_Occ).
He clicks on an item to read the full page (Browse) and follows a link while reading it to see the picture of a particular printer model (Browse). In order to collect all the pictures linked from the saved pages, he issues the Project operation provided by the tool to get JPEG files only (with the occurrence type being JPEG). Having realized that he likes a company X’s products, he issues a new query (Search) and go through a similar procedure (Browse) as above to save relevant pages, including those about digital cameras, under a new topic, “X”. To narrow down to the pages about laser printers manufactured by X, he uses an operation (Select with the intersection of the two topics) and saves the result under a new topic name “laser printer by X” (Get_Sub & Gen_Occ). Finally he creates an association between the two topics with “manufactured by” (Gen_Assoc).

This scenario illustrates how searching and browsing operations are connected with the operations in the proposed framework, which would otherwise be done manually in isolation, perhaps giving cognitive burdens to the user. In essence, the knowledge level operations bridged the gap between information level operations in a systematic way.

**Scenario 2**: A week later, the user wants to place an order for a laser printer by making the final decision. He goes to his knowledge space and realizes that all the search results were saved and organized by various company names (assuming that he repeated the similar process in the Scenario 1). He chooses the topic “laser printer by X” to see the associated pages one by one. Realizing that the prices are pretty high, he recalls that he searched on “laser ink printer” and pulls out one of the search sessions in the past (Backward).
This scenario shows how the information searched in the past can be reused using the proposed framework. Since the search results are organized at the knowledge level that is more compatible with user’s cognition than sets of information objects, users should be able to pull out old search results for more efficient perusal. The Backward and Forward operations, when implemented appropriately as in the Walden’s Paths [38], provide an additional help. When a reasonably large network of topics through associations is available after many search sessions, a large portion of navigations would be within the information sub-space guided by the knowledge space created by the user.

With the notion of knowledge space, we can provide an information access environment where users can keep track of the searching and browsing results for both short-term and long-term purposes. Since the space would be semantically tight with related topics and associations, it would be a useful guide for efficient searching and browsing that would otherwise be unbound.
IV Interaction Framework

We have designed an interaction framework for sketchBrain and implemented a prototype system that includes a search engine and the interaction tool, capturing the key ideas of the two-level model described below. sketchBrain is implemented with an open source graphics library (http://www.jgraph.com/) in Java, which we extended for our purposes.

As in Fig. 3, the framework connects users with the Web through Interaction Mediation. While a user has a virtual workspace, the Web side is assumed to have a conventional search engine and browsing facilities. When the user searches/navigates the Web and attempts to make informed decisions based on the information found, Interaction Mediation provides a support with the goal of relieving his cognitive burden in the information seeking process. It consists of various tools that facilitate users’ information seeking activities in terms of searching and browsing and work space creation/manipulation. Inter-space Man-
ager associates trails and user’s reification of them with raw information in the Web and provides facilities to manipulate them. A detailed description of the components for Interaction Mediation is given in section 4.2.

4.1 Personal Workspace

![Figure 4. An Example Screen Shot of sketchBrain](image)

A screenshot containing the user interface of *sketchBrain* is shown in Fig. 4. On the right is Web browser (as in (5)) for the exploration of the Web, and on the left is a visualization of the user workspace where three workflows are sketched as indicated by (1). Using this tool, a sequence of queries and search results and their relationships can be recorded as much as the user wishes to re-
member for future use. In effect, the network of topics and associations expresses her own conceptual understanding of the search results (e.g. creating a user-defined topic using (2) or associating topic nodes using a semantic association using (3)). In other words, our interactive workspace keeps track of previous interactions such as queries, browsed pages, and their relations to help users create own knowledge space. Using this knowledge space, the system can show what the user previously has seen and accessed and the reasons why she approached to them. In addition to this feature, our system can provide the relevant context of a specific page (like the one pointed by (4)) through time-variant multiple spreading activations, which can be used as a guidance for further navigation.

4.2 Interaction Mediation

4.2.1 Path Recommendation

Spreading Activation ([1]) is a well-known information access technique in associative networks. It was motivated from an analogy to the propagation of activation in a human brain. Algorithms using spreading activation start from source nodes and spread activations via weighted links to other nodes in the network. In this thesis, we utilize Time-variant Multiple Spreading Activation (TMSA) to recommend relevant paths from a certain page. In order to analyze the current interest of an exploratory searcher, our algorithm introduces new constraints and a procedure.
Let us define

Target Network = \((N, E)\)

where \(N\) is a set of nodes with \(N \subset \{N_{\text{personal workspace}} \cup N_{\text{web topology}}\}\) and

\[ E \text{ is a set of edges, } E \subset \{ E_{\text{personal workspace}} \cup E_{\text{web topology}}\}\]

A target of TMSA is an integration of Web topology and personal workspace. Web topology is the same as existing Web, and personal workspace consists of user created nodes and edges. Although the user-created nodes and edges can explain information in a different way, they have very important information about how accessed information is associated each other under users’ own information seeking criteria. Therefore, we need to regard it as a single network, and spreading activation will be executed over the network. Given the network, we define the iterative activation procedure as follows:

\[
A_i^{(t+1)} = \frac{1}{\sum_j \sum_j w_{ji}} \cdot \sum_j \alpha w_{ji} S_j^{(t)} \frac{1}{\sum_k w_{kj}}
\]

\[
S_i^{(t+1)} = T_i \cdot A_i^{(1)} + A_i^{(t)}
\]

where

\(A_i^{(t)}\) = activation level on node \(i\) at time \(t\),

\(S_i^{(t)}\) = spreading energy on node \(i\) at time \(t\),

\(\alpha\) = decaying factor, and

\(w_{ji}\) = association strength between node \(j\) and node \(i\)
In our network, each node $i$ has an activation level $A_i^{(t)}$ and a spreading energy $S_i^{(t)}$ at time $t$. Each activation level computed as in Equation (1) is determined by the spreading energy of adjacent nodes. When adjacent spreading energy is summed, the association strength between node $j$ and node $i$, $w_{ji}$, is multiplied. This specifies how much adjacent spreading energy influences node $i$. The amount of spreading energy is determined by Equation (2) where $A_i^{(1)}$ represents the initial activation level on node $i$, and has a value when the user previously has interacted with the node. In other words, it has a value when it is from the personal workspace. $T_i$ is a time decaying function and relieves its effects from the time when it first interacted. Other constraints will be described in next subsection. During spreading activation, spreading energy, $S_i^{(t)}$, is accumulated at each node, and finally the accumulated spreading energy is used for recommendation.

A. Constraints

For reasonable performance, we selected Wikipedia as a test environment and set constraints of TMSA to best suit the characteristics of Wikipedia. We defined the association strength, $w_{ji}$ as:

$$w_{ji} = \tau_i \cdot \varepsilon_i \cdot p_{(j \rightarrow i)}$$

(3)

where

- $\varepsilon_i =$ user-participation constraint
- $\tau_i =$ fan-out constraint
- $p_{(j \rightarrow i)} =$ path constraint from node $j$ to node $i$
These three constraints are dependent on the characteristics of the network. The detailed descriptions for the constraints are given below. These constraints ($\tau_i, \epsilon_i$, and $p_{(j \rightarrow i)}$), are restricted to Wikipedia and are used in experiment 1 (evaluating guided navigation) and experiment 2 (evaluating workload reduction).

- **Fan-out Constraint**

  A node with a large branching factor, which is connected to many others, may be bypassed or have a penalty in the activation process. Since Wikipedia we used as the testing environment has the properties of a scale-free network where the nodes follow power-law degree distribution, $p(k) \sim k^{-\gamma}$ like other scale-free networks, we employ this distribution as a fan-out constraint. Scale-free network is noteworthy kind of complex networks and most real-world networks like social network are classified into this category. The power-law degree distribution is the most important feature of scale-free networks. In power-law degree distribution, $p(k)$ is the probability that a node in the network connects with $k$ other nodes.

- **Path Constraint**

  There may exist several different kinds of links in a network that spread activations. This constraint discriminates preferred paths from others. In a semantic network, it gives preference to meaningful links. In our algorithm, we treated user-defined paths as more important.
- **User Participation Constraint**

An online social network like Wikipedia collectively facilitates the spread of ideas. As a result, it is critical to record how the network has evolved over time and which users have participated in the spread of ideas. As a way to be sensitive to this nature of Wikipedia and take advantage of it for our tool, we decided to utilize user participation as a constraint to be considered for spreading activation, with two variables, ‘fad’ and ‘stickiness’. *Fad* refers to a fashion that becomes popular in a culture relatively quickly but loses popularity dramatically. Stickiness means the likelihood of having steady popularity. Sticky topics usually were struggle with many problems, and topics that have no solution at that time usually have lots of stickiness. For example, gossips and temporary news have low stickiness and high fad, but neural science currently have high stickiness and low fad. Fig. 5 shows how the user classification can be classified.

![Figure 5. Classification of User Participation](image)

34
We assume that temporal interest of users is classified into one of them. Therefore, we design the variables, and a vector of them is used for determining users’ intent.

\[ v_i = (\text{stickiness}_i, \text{fad}_i) \]  \hspace{1cm} (4)

where

\[ \text{stickiness} = \frac{1}{N_{\text{user}}} \sum_{n_{\text{user}}} \frac{t_{\text{last-edited}} - t_{\text{first-edited}}}{t_{\text{current}} - t_{\text{first-edited}}}, \]

\[ \text{fad} = \text{the number of in-links}, \]

\[ N_{\text{user}} = \text{the number of users}, \]

\[ t_{\text{current}} = \text{current time}, \]

\[ t_{\text{first-edited}} = \text{time when it was first edited}, \] and

\[ t_{\text{last-edited}} = \text{time when it was last edited}. \]

The value for the user-participation constraint on node \( i \), \( \varepsilon_i \), is computed by summation of cosine similarity values between recently visited articles’ vector \( v_j \) and itself.

\[ \varepsilon_i = \sum_j \text{cosine.similarity}(v_i, v_j) \] \hspace{1cm} (5)

Each vector \( v \) has two dimensions with the two components, \text{stickiness} and \text{fad}. 

35
B. Example

Figure 6. An Example Network

Here is a detailed example of the TMSA inferencing algorithm using an example Web collection. Fig. 6 shows an example target network that is the result of integrating Web topology and personal workspace. In the network, P4 is a user created topic, and P1 is a page that the user is browsing. For calculating activation levels and spreading energy, we assume that we have the following weights in Table 2.

Table 2. A Weight Matrix of Example Network

<table>
<thead>
<tr>
<th></th>
<th>P0</th>
<th>P1</th>
<th>P2</th>
<th>P3</th>
<th>P4</th>
<th>P5</th>
</tr>
</thead>
<tbody>
<tr>
<td>P0</td>
<td>0.0</td>
<td>0.2</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
</tr>
</tbody>
</table>
When $t = 1$, 
\[ S(0) = 0, \quad S(1) = 1, \quad S(2) = 0, \quad S(3) = 0, \quad S(4) = 1, \quad \text{and} \quad S(5) = 0 \]

If we ignore the time variant function and assume that the decaying factor is 0.5, we have the following situation when $t = 2$:
\[ S(0) = 0, \quad S(1) = 1, \quad S(2) = \frac{1}{0.2+0.2+0.2+0.2+0.4} \cdot (0.5 \cdot 0.2 \cdot 1) \approx 0.0667 \]
\[ S(3) = \frac{1}{0.2+0.2+0.3+0.2+0.2+0.4} \cdot (0.5 \cdot 0.3 \cdot 1) \approx 0.1 \]
\[ S(4) = 1, \quad S(5) = \frac{1}{0.2+0.2+0.3+0.2+0.2+0.4} \cdot \frac{0.5(0.2+1+0.4+1)}{0.2+0.4} \approx 0.3333 \]

If we stop the activation at $t=2$, we can sort out-links (P2, P3, and P5) of P1 and suggest paths in descending order. In this example, the first is P5, and the second is P3.
4.2.2 Search and Navigation Trails

Some past applications ([15], [52], [35]) attempted to keep track of search or navigation trails and showed their usefulness. `sketchBrain` also keeps track of query trails and post-query navigation trails, which are based on click streams following issued queries. As in Fig. 7 and 8, search queries and post-navigations are sketched in the workspace of `sketchBrain`. Using the JDIC library (JDesktop Integration Components), `sketchBrain` gets action events and shows them in our workspace. The event includes accessed page URLs, parent page URLs, spent time for reading, and etc. For example, as in Fig 5, assume that a user starts from the page with “Apple”, and then clicks the link for “Steve Jobs”. At that time, in the system catches the clicked URL and its parent, so it can make a new association between “Apple” and “Steve jobs”. When a search result page is viewed for a query, the tool does the same thing. The page that the user clicked is created on the workspace and has a relation with the query topic. In the example shown here (Fig. 6), the query “iPod” has been connected with the two topics corresponding to the pages the user clicked.

![Figure 7. A Snapshot of the Navigation Trails](image-url)

Figure 7. A Snapshot of the Navigation Trails
4.2.3 Topic Extraction and Association Recommendation

Topic Extraction module creates a topic from a specific information object that has been found by the user. It recommends a topic when the user wants to add the object as an occurrence. It is currently implemented by extracting its title name of the object and pruning meaningless characters, but can be extended to extract a more representative topic.

Associations defined between topics are automatically generated in this system. That is, the association agent is capable of extracting appropriate associations between topics that the user might be interested in. This automatic generation is important to facilitate users’ information searching job because users visit numerous Web sites (occurrences) and save a number of topics from them. Moreover, making a relationship between a newly created topic and a previously collected topic is useful to utilize users’ history knowledge.

Thus, the agent extracts a proper association for two given topics from the
set of relevant documents. We firstly focus on the user-related documents, i.e. the documents that the user has browsed or are given as a retrieval result. From the relevant documents, the agent extracts some sentences that contain the given pair of topic. Next, the most frequent verb appearing in those sentences is selected as an association for the two topics. Hypothetically, a verb would express a main meaning of a sentence and, if the sentence includes the topics, then the verb should contain the meaning of relationships between the topics. Therefore, based on the hypothesis, the agent extracts a certain verb and makes it as an association. As an example, “CLP-300 colour laser printer is manufactured by Samsung…” is translated into a triplet like <laser printer, manufactured, Samsung>, where the verb, “manufacture” frequently occurs in the sentences containing the topics, “laser printer” and “Samsung”.

In addition to the association generation, in this system, we need additional processes such as verifying the generated association and unifying similar associations. Since the agent creates the association automatically, we should determine whether the generated association is appropriate or not. For this purpose, the agent utilizes a Web search engine, Google Search API package\(^5\). That is, the agent creates a query like “Samsung manufactures laser printer” if the association “manufactures” was generated, and then receives the search result. Based on the result, the agent counts how many documents are retrieved. We assume that a small number of documents indicate the association is not reasonable. In other words the association is validated only when a sufficient number of documents are retrieved.

After the validation process, the remaining task is unification of generated associations. In order to avoid generating associations with similar meanings, such as “manufacture” and “create,” the agent employs a simple unification method that utilizes lexical information from WordNet.

We tested the accuracy of the association generation method using five people. It was not intended to be statistically significant but was meant to give a ball park figure and generate issues for future research. Each of the five people received a different seed topic and was asked to contemplate a way of using the topic to search interesting documents related to it. For example, when a participant received the topic “president Bush”, he decided to look for articles where he could identify those who criticized him.

Participants then used our system to search and browse relevant documents for their purposes. While browsing documents, they were asked to mark relevant documents with appropriate topics (e.g. Castro, Iraqi people). Having gathered all documents and generated topics, the system automatically generated associations between pairs of topics as described previously. As a result, 200 documents were observed, and 71 topics were generated from them. The system created 176 triples by suggesting associations.
Overall, 53.41% of accuracy was obtained. By users, accuracy for the associations created for the User 1 was the highest, 69.23%, and the lowest was 29.17% for User 4. We recognized that most of the errors were related to verb extractions. We assumed that verbs would express the main idea of the relationship between two topics. However, the extraction method was too weak to detect the appropriate verb. That is, for compound sentences such as “EU made up of several European countries warns Japan that …,” the method extracted “made” as a verb instead of “warns”. Even though the proper verb was extracted, there were other obstacles. Since the participants sought information mostly from news articles, quotations were sources of errors. For example, in the sentence, “EU said that Google should expose …”, the system extracted “said” as the association between “EU” and “Google” because the query, “EU said Google”, has a large number of hits in the search result although the association is not reasonable. With the difficulty (low accuracy and high time complexity), this associa-
tion recommendation module was not employed in next experiments (in Chapter VI.)

4.2.4 Other Components

In the current implementation, there are three components that are not as fully developed as the other three introduced. Although more complete development of these components would provide added functionality and make the interface more amenable for exploratory search, the lack of the full functionality does not invalidate the main goal of reducing cognitive burden in exploratory search through Interaction Mediation, as proved by the experiments in Chapter VI. It simply means that users need to do some work manually or semi-automatically.

Session Identification is to automatically discover different session boundaries. An exploratory search task may require multiple sessions corresponding to dynamically generated information needs. Because of the nature of exploration, transitions from one session to another occur to reflect changes in the focus of information seeking. Instead of asking users to search/navigate in a separate workspace or use an explicit indicator for a distinct session, we opted for an automatic session identification method. Although session boundary discovery has been studied [51] [42] and needs to be developed for the overall user interaction processes in using our tool, it is included only in our path recommendation algorithm. As a session boundary, a temporary cut-off is used in the time constraint function $T_i$ in TMSA.
Inter-space Manager is responsible for the operations whose domains and ranges are across the two spaces. The operations like Classify and Project seem useful to handle corresponding occurrences from the perspective of users. This module is not implemented in the current prototype.
V Implementation

We implemented a prototype system that includes a search engine and the interaction tool, capturing key ideas of the model, especially the knowledge level and inter-space operations. In the implementation, the basic graphic user interface tool for displaying personal workspace was obtained from Java JGraph\(^6\) and extended for our purposes. Because proposed personal workspace has semantics, all objects are represented in the form of Topic Maps. Therefore, the implementation employed the Ontopia Knowledge Suite (OKS) that provides API for manipulating Topic Maps objects and displayed it on the visual workspace.

5.1 Key Design Goal

There is a key design goal for the system. The goal is to help users for efficient and effective information discovery. From the system design point of view, efficiency in task completion is an important criterion because quality-aiming systems often require a burden on users and/or additional time for the information seeking task. We approach this issue by developing user-friendly interface (e.g. Object moving/naming) similar to popular tools for visual object dealings.

\(^6\) http://www.jgraph.com/
5.2 Implementation Details

5.2.1 Authoring Knowledge Objects in Workspace.

In exploratory web search, users encounter too many web pages to remember every single page. Our system automatically visualizes click-through data of users and represents them with topics and associations. Based on the topics and associations, the interface of our prototype allows users to manage knowledge objects.

Fig. 10 shows how users can add a new topic on the workspace. For example, if we enter “Steven Jobs” in the conversation box, a new topic node is created, as in Fig. 11.

![Figure 10. Adding a Topic on the Workspace-before](image)

Figure 10. Adding a Topic on the Workspace-before
In the knowledge space, semantic association plays an important role of future use. It represents meaningful and interesting relationships that are core in content analytics and knowledge discovery. Fig. 12 shows the workspace which a user-defined association, “is CEO-OF,” is inserted with a caption on the arrow.
In addition, there is the concept of session in our implemented system. In the system, users can explicitly represent the session, and freely manipulate knowledge objects in corresponding session. The way to make a new session is similar to Firefox Web browser. As in Fig. 13, clicking a “New tab” button, a new window that gets a tab name is popped up, and the created session is added to the next of existing session tags.

![Figure 13. Making a New Session](image)

5.2.2 Saving and Loading Workspace

Below Fig. 14 shows how a workspace is saved. The prototype provides a function that can allow the workspace to save a path. Therefore, whenever users want to save their works, they can store them. They can be re-loaded, and users can continue to search/navigate information.
Figure 14. Saving a Workspace
VI Experiment

In this chapter, we report our empirical evaluation. The proposed approach was evaluated in three different ways.

- Experiment 1: Evaluation of Guided Navigation Method
- Experiment 2: Subjective Workload Assessment
- Experiment 3: Evaluation in Information Reusability

In the first experiment, we evaluated the guided navigation method as a component of the interaction tool. Because this method is one of important components, it can be an indirect proof of our interactive information seeking interface. This tool was compared with other guided navigation methods. In the second experiment, we tested whether the proposed tool helps reducing users’ workload (i.e. cognitive burdens) in exploratory search, the primary motivation for devising the proposed method. In the last experiment, we tested the tool for its usefulness in reusing previously encountered information. More specifically, it tested how the proposed tool helps users in performing tasks that require organizing and remembering the results from searching and browsing.

Wikipedia was selected as a test environment for the second experiment where exploratory search was the main task. We chose Wikipedia instead of the entire Web because it provided us with a somewhat controlled environment for topic and association generations. It contains a reasonably large number of encyclopedia articles and covers a very wide range of subject areas, providing a suitable environment where exploratory searches can take place.
6.1 Guided Navigation

We ran our experiment was to find out whether our automatic guided navigation method would help users’ information seeking tasks in general. It was targeted at our guided navigation’s utility in terms of effectiveness by measuring recommendation relevance. The participants manually evaluated relevance of top 10 paths in each case.

6.1.1 Experiment Design and Result

We asked our participants to evaluate how helpful it was to use our tool based on Time-variant Multiple Spreading Activation (TMSA). The participants were divided into three groups: the first with our time-variant spreading activation algorithm, the second with a method based on TF-IDF using the latest query, and the third with random recommendations. The participants freely formulated their own search queries and got engaged in browsing. Their tasks were to find relevant homepages given a broad question. The task is similar to Topic Distillation Task in TREC 2003 Web track (TREC-2003 Web Track). We allocated a broad topic like “Korean IT industry” to the participants who had to find a list of related homepages, not any pages about the questions, which provide meaningful information on the query topic. For making more realistic exploratory environment, after experiments, we requested participants to make a summary describing the topic. We regarded information that is not included in the summary as an not meaningful information.

In the course of finding relevant homepages, the participants often had to get engaged in navigation, at which time the system made recommendations.
The participants were asked to evaluate the recommended top 10 paths for their relevance. The average number of links (including internal links, external links, redirects, and binaries) in articles that the participants had visited was 38.9. Each of the five participants performed six tasks, and the total number of browsing actions was 112. Table 3 shows the result that compares three different methods. % means that how many recommendations are correct as a fraction of 100.

<table>
<thead>
<tr>
<th>Accuracy for Relevance</th>
<th>TMSA</th>
<th>TF/IDF</th>
<th>Random</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>72.78%</td>
<td>57 %</td>
<td>33.4%</td>
</tr>
</tbody>
</table>

It clearly shows that our method based on TMSA outperformed the other two methods. Moreover, the absolute value for accuracy is very promising. However, TF/IDF shows relatively low performance. The value is analyzed by that exploratory searches were very frequently oriented and reoriented. Thus, previous query was expired in short time, and users wanted to find different issue related to his broad need. It made TF/IDF become the bad result. Random suggestion recommended paths randomly, so, because there are many meaningless links, its accuracy was the worst.

### 6.2 Reducing Workload

Our second interest was to find out whether the system implemented based on the two-level model would help reducing workload of users. Given the motivations of our work, workload is a reasonable measurement to test the tool’s
efficacy because it measures how much effort is required to complete an exploratory search task. In this experiment, we used a special instrument, the subjective workload assessment technique (SWAT) [37]. This method has been utilized for evaluating three criteria: time, mental effort, and stress.

We asked the participants to perform a total of 10 exploratory search tasks in the Wikipedia environment where the articles were judged for usefulness in learning background and detailed information for exploratory search tasks. In this experiment, we utilized a simple English Wikipedia\(^7\), and evaluated efficacy of our information seeking interface as an aid to exploratory search. Each task has one topic selected from the topics of 10 different Wikipedia categories. This topic classification scheme uses nine shared categories\(^8\) for all Wikipedia articles. For a more realistic exploratory search environment, we provided blank forms that they had to fill out. The forms are composed of two parts: semantic annotation and summarizing. Semantic annotation is to annotate information about what related entities appear in texts, and summarization means answering non-factoid questions such as “writing a state of the art” and “writing important background information”. To minimize potential biases like leaning effects, the participants applied two methods, with and without the interface, in an alternating fashion.


The participants’ rating of SWAT range between 1 (the best) and 3, and the result of workload analysis is presented in Table 4. Our interface received a total score of 4.6, which is a significant improvement over the case without the interface, when the score was 6.4. In particular, the difference was the greatest for mental efforts as intended and expected for the interface. These observations showed that our new information seeking interface helped reducing workload in three different ways in the task of exploratory search.

### 6.3 Information Reuse

Since our two-level model and its manifestation as a tool were devised to help users reducing cognitive efforts in information seeking processes, manifested by searching and browsing activities, we decided to focus on information reuse activities in information seeking. In the web environment, users often have to skim through an overwhelming amount of information, suffering from information overload, before their goals are achieved. Our experiment is an exploratory study designed to investigate whether our tool helps users in organizing, remembering, and reusing the information that has been encountered. Our tool
was compared against two other systems designed for the same purpose: the Favorites tool in the Web browsers and the Stuff I’ve Seen (SIS) system [11]. The Favorites tool (or book mark tool) was found to be useful for PVR (Post-valued Recall) [49] and SIS was developed to search the information that has been seen in the past.

6.3.1 Experiment Design

The three methods, the Favorites tool, SIS, and sketchBrain, were compared in six different tasks by ten groups of users, each consisting of three undergraduate students. In total, 30 users were employed for six different tasks using three different methods. The six tasks consist of questions in six different domains like Medicine and Sports. The tasks were designed as follows. For a task, the participants (users) were first asked to read 30 pre-selected web pages. One minute per page was given to simulate an information skimming situation. The participants were then asked to organize the pages using the given tool within one minute. After the preparation stage, they were given three information hunting questions elicited from the 30 pages they read, such as “Name the two new members to the Board of Directors of Good Samaritan Hospital in New York.” The participants were timed for completion of each question answering. Since the maximum time given to each question was five minutes, the time taken for an unsolved question was assumed to be solved in five minutes, the maximum. In order to minimize user dependency and learning effects, the users were assigned to six tasks using three different methods in an alternating fashion (see Fig. 15). Each user evaluated each method twice for different tasks, and each
task was given to the three users in an effort to minimize user dependency. Three users used the three methods in different sequences for different tasks so that there is little learning effect on average.

<table>
<thead>
<tr>
<th></th>
<th>T_1</th>
<th>T_2</th>
<th>T_3</th>
<th>T_4</th>
<th>T_5</th>
<th>T_6</th>
</tr>
</thead>
<tbody>
<tr>
<td>U_1</td>
<td>M_1</td>
<td>M_2</td>
<td>M_3</td>
<td>M_1</td>
<td>M_2</td>
<td>M_3</td>
</tr>
<tr>
<td>U_2</td>
<td>M_1</td>
<td>M_2</td>
<td>M_3</td>
<td>M_1</td>
<td>M_2</td>
<td>M_3</td>
</tr>
<tr>
<td>U_3</td>
<td>M_1</td>
<td>M_2</td>
<td>M_3</td>
<td>M_1</td>
<td>M_2</td>
<td>M_3</td>
</tr>
</tbody>
</table>

T: Task, M: Method, U: User

Figure 15. Experimental Design for Each Group

To ensure that every participant has some familiarity with the three tools, we gave them a tutorial with 10 minutes of practice sessions in the same place with all the participants together. Following is a brief description of the other two methods compared against our knowledge space tool.

The Favorites tool is often used to save visited pages for future references. A user can create folders and put a page to be remembered into a folder. Folders are like topics in our tool and can be organized in a hierarchy. Most browsers are equipped with this tool.

SIS was developed to facilitate information reuse for various information resources. It provides the capabilities of fast unified indexing of various files in a desktop and of filtering files based on queries, file types, and time. It is now available on the Windows desktop.
6.3.2 Result and Analysis

Figure 16. Comparison for Task Completion Time

The time measures collected for individual users for all the tasks were averaged to see the difference among the three methods. Since there were ten groups, each consisting of three participants, and six tasks performed with each tool, a total of 180 data was averaged for each tool. Each data point is for a task consisting of three questions solved by a participant using one of the tools.

The comparison result is shown in Fig. 16. It took about 50 seconds on average to solve the problems using our tool, but 88 (about 76% longer) and 70 seconds (about 40% longer) using the Favorites tool and SIS, respectively. Using the method 2, i.e. SIS, that has the extended search functions only, participants often produced no answer within the time limit because the pages were not pre-organized in their own ways. Although SIS didn’t require any extra user efforts
to organize the pages, the time spent on the organization was only one minute, once for all the tasks. If the initial investment for our tool is spread across all the questions, the extra time spent is very small.

The comparison data for different tasks as in Fig. 17 are even more encouraging in that our tool outperforms the Favorites tool for all the tasks and SIS for all but one case (task 1). Task 1 was easy for SIS because a search query with a proper noun can easily retrieve the relevant page but not so easy for our tool because the proper nouns were not used as topics.

Our further analysis of the experimental results revealed relative advantages and disadvantages of the three methods for the given tasks. The Favorites tool is easy and efficient to use in organizing pages in a hierarchical way, but makes it rather difficult to look for specific information. In a shallow hierarchy,
the granularity level is usually too high to pinpoint a folder that may contain the information. When it is deep or skewed with a deep branch, it would be time-consuming to repeatedly go down the hierarchy to locate a folder that may contain the information. Besides it is impossible to express a semantic relationship between two pages.

In the case of SIS, a big advantage is that users do not have to take an extra step for constructing a knowledge space and yet find information using various contextual cues. However, it suffers from all the problems associated with query-based search engines, such as inability to formulate a meaningful query or recall contextual cues. It still has to rely on searching information space without personalized knowledge space.

Our tool based on two-level model has the best of the both worlds. With the knowledge level operations, the information that has been encountered in the past is semantically organized in a personal conceptual space. The “Sequence Guide” feature allows users to easily go back to a past query session and the corresponding topic mat in knowledge space to provide a simplified view. Although the time required for the construction of knowledge space in the experiment was artificially limited to one minute, however, it is entirely possible for users to feel that the additional efforts necessary for knowledge space construction is an added burden.

In order to see statistical significance of the results, we employed ANOVA that is used for determining statistical significance of differences among different groups. Table 5 shows that the mean for our tool was better than those of Favorite and SIS.
Table 5. ANOVA Result

<table>
<thead>
<tr>
<th>Methods</th>
<th>Mean</th>
<th>Std.Deviation</th>
<th>Lower Bound</th>
<th>Upper Bound</th>
</tr>
</thead>
<tbody>
<tr>
<td>1: Favorites</td>
<td>87.69</td>
<td>98.82</td>
<td>62.16</td>
<td>113.22</td>
</tr>
<tr>
<td>2: SIS</td>
<td>70.09</td>
<td>67.67</td>
<td>52.61</td>
<td>87.57</td>
</tr>
<tr>
<td>3: Our Tool</td>
<td>50.33</td>
<td>43.78</td>
<td>39.02</td>
<td>61.64</td>
</tr>
</tbody>
</table>

ANOVA puts all the data into one number (F) and gives us one P for the null hypothesis. The value was equal to $F(2, 177) = 3.866 \ (p < 0.05)$, and the difference was reliable at the 95% confidence level. It means that users were more likely to say that our tool had superior information reusability.

Table 6. Pairwise Comparisons

<table>
<thead>
<tr>
<th>Used Method</th>
<th>Used Method</th>
<th>Sig</th>
</tr>
</thead>
<tbody>
<tr>
<td>1: Favorites</td>
<td>2: SIS</td>
<td>0.39</td>
</tr>
<tr>
<td></td>
<td>3: Our Tool</td>
<td>0.015</td>
</tr>
<tr>
<td>2: SIS</td>
<td>1: Favorites</td>
<td>0.39</td>
</tr>
<tr>
<td></td>
<td>3: Our Tool</td>
<td>0.305</td>
</tr>
<tr>
<td>3: Our Tool</td>
<td>1: Favorites</td>
<td>0.015</td>
</tr>
<tr>
<td></td>
<td>2: SIS</td>
<td>0.305</td>
</tr>
</tbody>
</table>

In addition, information reusability was different, depending on the methods. Table 6 shows the detailed analysis about statistical significance for pairwise comparisons. In the result, the difference between our tool and Favorites was significant with a very high level of confidence. However, the significances of the differences between other pairs were less confident.
6.3 Subjective Comments

Analysis of the written survey questions and the participants’ answers provided after the completion of the tasks reveals further insights about our tool. This survey was given to them after experiment 3. The ratings provided by the participants were based on a five point scale, 5 being a complete agreement. Table 7 shows the result.

<table>
<thead>
<tr>
<th>Question</th>
<th>Rating</th>
</tr>
</thead>
<tbody>
<tr>
<td>The tool is useful in skimming a large amount of information and organize it</td>
<td>4.29</td>
</tr>
<tr>
<td>The tool doesn’t require much effort in organizing information.</td>
<td>3.29</td>
</tr>
<tr>
<td>It is easy to understand how to utilize the knowledge space.</td>
<td>3.52</td>
</tr>
<tr>
<td>The tool can be deployed to any software platform.</td>
<td>3.10</td>
</tr>
</tbody>
</table>

In the survey, the participants agreed quite strongly that the tool is useful (question #1) and easy to utilize (question #3). However, they didn’t agree as strongly that the tool does not require much effort, suggesting for further improvement in the knowledge space construction facility. The response to the last question indicates that the tool is not yet complete and should be tuned further. It gives not only a direction for further work but also an expectation that its utility can be improved further when the tool is more sophisticated and usable.
People also commented that our tool should employ a full text search engine for knowledge objects. Because we focused on implementing knowledge object manipulations and interaction mediation, we paid no attention to the knowledge search. There were also important comments about screen limitation and speed. In addition to the space for search engine, our workspace surely requires extra space for visualize it. Therefore, usual monitors were lack of space for showing everything. Speed for path recommendation had also a problem. Because it considered many structural features like network topology, so it took too much time. In future work, these limitations should be considered.
VII Conclusions

This thesis has proposed a novel interactive information seeking interface as a way to help exploratory search. Although its complete supporting is still challenging, this thesis provides underlying model and interaction framework for exploratory search. Chapter III studied how people search, and emphasized the fact that, when people do search, they conceptualize accessed information in a very personal way. Their conceptualization and implicit operations for its use are designed in underlying model, two-level model so that we have a connection between traditional information space and cognitive knowledge space. Chapter IV focused on how the underlying model implemented in the Web and provided framework for interaction mediation. In the framework, new personal workspace that is cooperatively organisable was proposed and intelligent supports for recuing cognitive burden were attempted. Chapter VI demonstrated that it is possible to significantly reduce cognitive burden for exploratory information seeking via various experiments.

In this chapter, the contributions of this thesis are revisited and future work is described. This research should help users face less cognitive burden for more productive learning and investigation.

7.1 Contributions

As mentioned in introduction, this thesis makes two important contributions to the area of information retrieval.

1. This thesis have proposed new environment based on our own model for exploratory search, which explicates operations at the knowledge
level and across the information and knowledge spaces in addition to the typical information level operations, searching and browsing.

2. This thesis presented an interaction framework and novel intelligent interaction mediation between users and the Web. Especially, path recommendation algorithm showed promising performance.

7.2 Future Work

There are a number of directions for further research. First of all, the two-level model can be extended further and implemented in other ways with different emphases. For example, it would be useful to search using a topic-association-topic triplet as a query. In this case, information objects need to be indexed accordingly. Second, automatic generation of topics and associations require further research, which is essential to reducing users’ burden in constructing their own knowledge space. Third, the function of Session Identification module should be more fully developed, again to reduce users’ workload further. Finally, a more complete system with Inter-space Manager must be developed for fuller manifestation of the two-level model and deployed to a real user environment for more extensive experiments.
탐험적 검색을 위한 정보 탐색 인터페이스

공학부 박호건

웹이 대중적인 인기를 얻기 시작하면서, 많은 사람들은 이것을 다양한 목적으로 활용하기 시작하였고, 이에 따라 많은 인지 부담이 생기기 시작하였습니다. 이런 인지부담은 적합한 검색 질의 어를 만들 때, 새로운 정보에 대해 정리하고 다음 검색 방향을 설정할 때 등의 다양한 상황에서 발생하게 됩니다. 그러나 대부분의 검색 엔진은 정보를 다루는 동안 생기는 부담보다는 오직 한번의 검색에만 적용되는 검색 질의 처리와 그에 상응하는 브라우징 기능에만 집중하고 있습니다. 그래서 본 학위 논문에서는 사용자와 웹 사이의 상호작용을 위한 프레임워크를 제안합니다.

이 프레임워크는 웹에서 가장 많은 인지 부담이 많이 발생하는 탐험적 검색(Exploratory Search)작업에 그 초점을 두고 있습니다. 이 탐험적 검색 작업이란, 어떤 주제에 관한 배경 지식을 습득하고, 의사결정을 하기까지 다양한 정보들을 수집하여, 최종적으로 의사결정을 하기까지의 모든 과정을 의미합니다. 이 작업에서는 다른 작업보다 많은 질문을 보내고, 더 많은 페이지를 방문해야 하기 때문에 본 연구의 제안을 평가하기 위한 좋은 환경이 될 수 있습니다.

이 프레임워크 위에서도, 저는 사용자의 인지 부담을 줄이기 위한 새로운 정보 검색 인터페이스를 구현하였습니다. 이 인터페이스는 검색엔진과 함께 정보 객체들을 다루기 위한 개인적인 작업공간(Personal Workspace)을 제공함

65
이 작업공간은 사용자와 시스템이 공동으로 만들어 나가는 공간으로, 효율적이고 조직적인 검색을 위해 사용자가 접했던 정보를 개념화 (Conceptualization)하는 것을 도와주고, 미래에 유저가 재사용 시 도움을 주기 위한 궤적(예: 검색 퀘리, 브라우징 경로, 사용자의 상호작용 기록)을 기록합니다. 이를 바탕으로 몇 가지 지능적인 모듈을 제안하여 사용자와 웹 사이의 효율적인 검색을 돕도록 유도하고자 하였습니다.

제안하는 첫 번째 모듈은 사용자의 브라우징 경로를 미리 추천하는 역할을 합니다. 이 브라우징 경로의 추천은 인지 공학에서 널리 활용되고 있는 Spreading Activation 기술을 활용합니다. 본 연구에서는 과거 시스템과 사용자와의 상호작용 정보를 바탕으로 몇 가지 유용한 가중치 스키마를 제안하고, 이를 실제 추천에 활용합니다. 두 번째 모듈은 사용자가 접한 정보들로부터 토픽을 추출하여, 작업공간에 표현하여 줍니다. 여기서 추출된 토픽은 사용자가 현재의 검색 작업에 대한 인지 부담을 최소화하고, 정보를 자신만의 관점으로 재구성할 수 있도록 도와줍니다. 여기서 토픽들은 각각 그 근원이 되는 URI(보통은 URL 을 활용)를 가집니다. 세 번째 모듈은 사용자의 검색활동 사이의 의미(Semantics)를 추출하여 주는 관계 추출(Association Extraction) 모듈입니다. 이 모듈을 통해 위에서 추출한 개념화 된 토픽 사이에서 사용자가 홍미 있어할 만한 관계를 추출 합니다. 여기서의 관계는 정보검색에서 유용하게 사용할 만한 20 여가지 관계를 선별하여 사용하였으며, 기존의 웹 자원과 MIT 와의 공동연구를 통해 생성한 상식기반 지식베이스인 ConceptNet 을 활용하여 추출합니다. 네 번째 모듈은, 사용자의 검색활동을 검색의도나 관심사에 따른 세션의 전환을 식별합니다. 하나의 세션은 하나의 검색 의도 아래 이루어진 모든 검색 및 브라우징 활동을 의미합니다. 이 역시 사용자와 시스템간의 상호작용에 기반하여, 확률적으로 의도전환이 될 시점을 미리 예측하여, 시스템의 인터페이스와 브라우징 경로 추천 등에 활용할 수 있도록 도와주는 역할을 합니다.
프레임워크의 모든 모듈은 본 연구에서 제안한 Two-level Model 을 바탕으로 하고 있습니다. 이 Two-level Model 은 유저의 머리 속에서 이루어지고 있는 인지 공간과 실제 정보가 존재하는 정보 공간을 분리하고 있으며, 이들이 효율적으로 분산 처리되고 활용될 수 있도록 합니다. 본 연구에서는 이 모델 위에서 이루어 질 수 있는 핵심 현상을 분석하고, 활용될 수 있도록 미리 정형화해 놓았습니다.

이런 새로운 검색환경이 실제 탐험적 검색에서는 어떻게 사용자의 인지부담을 줄일 수 있는지 알아보기 위해, 30여명의 피실험자를 고용하여 다양한 실험을 해보았습니다. 사용자의 인지부담을 얼마나 줄일 수 있는지 SWAT Test 를 시도하여, 20%의 인지부담 경감효과를 확인하였으며, 추가적으로 사용자가 이 프레임워크를 사용하였을 때 정보 검색 작업의 효율성 그리고 정보의 재접근성을 실험(즐겨찾기와 Microsoft DeskTop 검색엔진(Stuff I’ve Seen)과 비교)한 결과, 높은 신뢰도의 통계적 유의확률로 정보의 효율적인 조직을 돕고 있다는 사실과, 이들의 재사용성을 확인할 수 있었습니다.
References


[19] JDesktop Integration Components: https://jdic.dev.java.net/


71


[57] Yee, K., Swearingen, K., Li, K., and Hearst, M, “Faceted metadata for image


Acknowledgements

I would like to express my sincere appreciation to my advisor, Professor Sung-Hyon Myaeng. He has always provided me with thoughtful advices about not only good research but also man’s life. It was the greatest fortune for me to be instructed by him. I am also grateful to the other members of my thesis committee, Professor Geehuck Lee and Professor In-Young Ko. Their critical comments have helped me make more valuable thesis.

Much of the research in this thesis is the result of collaboration with many colleagues. Gwan Jang, Sooran Jo, Hyungchul Roh, and Jong-Wook Choi worked with me. I am grateful to them for giving much help.

I have also enjoyed working with members of IR&NLP Lab. Yu-Chul Jung, Yun Jin, and Yoonjae Jeong encouraged me to keep a big picture in mind and gave me valuable feedback on my research. Heung Seon Oh, Young Ho Kim, and Yingshi Tian have lead research and taught me a model for research. Hyo-Jung Oh, Sa-Kwang Song, Sung-Pil Choi, Hae-Gyung Kim, Bashar Awad Mohammad Al-Shboul, Nguyen Khanh Ly, Jihee Yoo, Yoon-Jung Choi, Wookhyun Shin, Seong-Chan Kim, and Keun-Chan Park, all provided moral and valuable technical support.

I would also like to thank my friends in ICU. Many good friends that dream of a big future have lived together. I wish to thank all CSE 03, 신내림, 위그이, and TD members. Especially, I always thank to Hyungchul Roh, Keonkook Lee, Dae-Won Ko, and Dong-Hyun Lee. We are taking a great step forward! All my friends have made the history of ICU and it will be continued! In addition, I also would like to express thanks to Hyo-Joo Kim. Her enthusiasm always encouraged me.

This thesis is dedicated to my father, my mother, and Hyang-Mi. They thoroughly supported me and always gave me everlasting trust. Without their help, I can’t even imagine to be here.
Curriculum Vitae

Name : Hogun Park
Date of Birth : Aug. 30. 1984
Sex : Male
Nationality : Korea

Education

2000. 3 – 2003.2   Taegu Science High School
2003. 3 – 2006.8   Information and Communications University (B.S.)
2006. 8 – 2008.8   Information and Communications University (M.S.)

Career

Academic Experience

*International Conference*

1. Speaker, 10th Interactional Conference on Enterprise Information System (ICEIS 2008), Barcelona, 2008
2. Speaker, 21th IEEE International Joint Conference on Neural Networks (IJCNN 2008), Hong-Kong, 2008
Publications


