ADAPTIVE VISUALIZATION FOR FOCUSED PERSONALIZED INFORMATION RETRIEVAL

by

Jae-wook Ahn

B.A. in Sociology, Yonsei University, 1997

M.A. in Library Science, Yonsei University, 2000

Submitted to the Graduate Faculty of

the School of Information Sciences in partial fulfillment

of the requirements for the degree of

Doctor of Philosophy

University of Pittsburgh

2010

UNIVERSITY OF PITTSBURGH

SCHOOL OF INFORMATION SCIENCES

This dissertation was presented

by

Jae-wook Ahn

It was defended on

September 8th 2010

and approved by

Peter Brusilovsky, School of Information Sciences, University of Pittsburgh

Daqing He, School of Information Sciences, University of Pittsburgh

Michael Spring, School of Information Sciences, University of Pittsburgh

Michael Lewis, School of Information Sciences, University of Pittsburgh

Michael Christel, Entertainment Technology Center, Carnegie Mellon University

Dissertation Director: Peter Brusilovsky, School of Information Sciences, University of Pittsburgh

ADAPTIVE VISUALIZATION FOR FOCUSED PERSONALIZED INFORMATION RETRIEVAL

Jae-wook Ahn, PhD

University of Pittsburgh, 2010

The new trend on the Web has totally changed todays information access environment. The traditional information overload problem has evolved into the qualitative level beyond the quantitative growth. The mode of producing and consuming information is changing and we need a new paradigm for accessing information.

Personalized search is one of the most promising answers to this problem. However, it still follows the old interaction model and representation method of classic information retrieval approaches. This limitation can harm the potential of personalized search, with which users are intended to interact with the system, learn and investigate the problem, and collaborate with the system to reach the final goal.

This dissertation proposes to incorporate interactive visualization into personalized search in order to overcome the limitation. By combining the personalized search and the interactive visualization, we expect our approach will be able to help users to better explore the information space and locate relevant information more efficiently.

We extended a well-known visualization framework called VIBE (Visual Information Browsing Environment) and implemented Adaptive VIBE, so that it can fit into the personalized searching environment. We tested the effectiveness of this adaptive visualization method and investigated its strengths and weaknesses by conducting a full-scale user study.

We also tried to enrich the user models with named-entities considering the possibility that the traditional keyword-based user models could harm the effectiveness of the system in the context of interactive information retrieval.

The results of the user study showed that the Adaptive VIBE could improve the precision of the personalized search system and could help the users to find out a more diverse set of information. The named-entity based user model integrated into Adaptive VIBE showed improvements of precision of user annotations while maintaining the level of diverse discovery of information.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

1.0 INTRODUCTION

Today's information environment is getting much more complex day after day. The new medium such as the World Wide Web is unprecedented to any other information resources that have existed in human history in term of its size and its speed of growth. It is naturally unorganized and decentralized from its birth. How to locate relevant information from this extreme complexity and to distill valuable information from the mess is one of the greatest challenges of information science today.

The traditional information overload problem [9] is becoming more evident with the advent of the Web 2.0 [92], where the participation of the users in the creation of contents is one important property. People collaborate with each other to generate contents using Web-based Wiki software. They post and share their everyday experiences or expert level knowledge and views on various issues on their blogs.

However, the Web 2.0 is not just a curse to the information overload. With its expanding size and diversity, people began to see it as the rich resource of information, not just for everyday lives, but also for the professional level. People now try to collect vast amount of information from various sources, understand the complex background mechanism about specific events, and re-produce new information from them. What is getting complicated is not only the information space, but also the mode of information production and consumption.

We need new tools to catch up with this change. So far, the most successful information access tools are the directory services and Web-based search engines. The directory services have existed since the beginning of the Web. The search engines enabled users to find out relevant information easily even from the dumb terminal. They are still the most popular tools for information access. However, current simple browsing and searching frameworks are not capable enough to match with the fundamental change of the information environment.

One of the ideas that can handle this problem is personalized search [88]. The interaction model of personalized search is similar with plain, non-personalized search. It receives queries from users and then returns most relevant document surrogates in a ranked list. However, unlike the traditional searching, the personalized search approach tries to avoid the one-size-fits-all idea and understand user differences – different user interests or different contexts – and then provide more customized results for specific user interests under different contexts.

In order to accomplish this goal, user modeling ideas are usually adopted [44], by which user contexts can be considered for retrieving information. Even though the adoption of the user model and the implementation of personalized search aims to serve the users to retrieve information best fits to their needs, the addition of the one big variable (user model) can increase the complexity of the system.

From the system side, personalized systems should find out how to construct the user models according to various users and how to operate the constructed user models in order to provide effective personalization. From the user side, the increase of the complexity can cause the failure of users' proper mental model construction with which they can understand the functionality and expect the behavior of the systems. This complexity can lead to user frustration.

We have investigated several ideas [4, 5] to solve this complexity problem, especially bearing in mind the concept of exploratory search [83]. The idea stresses on the importance of the user interface, so that users can learn and understand their problems more thoroughly while they actively interact with the system beyond simple look-up search activities and can reach better results.

In the spirit of exploratory search, which gives users more control, we explored options allowing the users to view and manipulate the user models as well as to control the impact of the user models on search results.

A relevance-based visualization framework called VIBE (Visual Information Browsing Environment) [91] was selected in order to achieve this goal. VIBE makes use of reference points called POI (Point Of Interest) and can position documents according to their similarity ratios to the POIs. We defined two groups of POIs – one for the queries and the other for the user model keywords – and displayed the documents according to the VIBE algorithm. This idea implemented a novel adaptive visualization approach, called Adaptive VIBE. The dissertation tries to prove the value of Adaptive VIBE and investigate its properties in order for future improvements.

1.1 CONTRIBUTIONS OF THE STUDY

This dissertation presents three round of studies (two pilot studies and the main user study) for this adaptive visualization approach. In the first pilot study, we conducted an experiment using the log data of the previous text-based TaskSieve study [5]. According to the findings of the experiment, we could see the potential that the VIBE-based interactive visualization could help improving the traditional personalized search approaches.

At the same time, we need to enrich the content of the user model that becomes the foundation of our adaptive visualization. A lot of user modeling approaches still remain at the classic bag-of-words model, where the user interest or the information seeking context are represented as a set of keywords. With this simple approach, the conceptual level semantics of the user models are hardly expressed.

This is where the second pilot study started. An extended version of adaptive VIBE was experimented again, with its user model enriched with named-entities. Named-entities (NE) are semantic categories and a pointer to a real world entities [93], so we were able to extend the user models conceptual power. The experiment results confirmed the benefits of using NEs, when they were mixed with keywords and built user models.

Even though these results are interesting and encouraging, it just revealed the static side of our approach. In the context of the experiment, real user interactions were missing and just the static pictures generated from the log data were evaluated.

Therefore, in the main study, a full-fledged user study was conducted with real users who played the role of information analysts and performed the search tasks designed to reflect real world problems as closely as possible. We defined three conditions: (1) the baseline, (2) Adaptive VIBE, and (3) Adaptive VIBE plus NE-based user models.

The results of the user study found that Adaptive VIBE could help the users to collect better information compared to the baseline. They were able to collect more diverse text fragments that could contribute to solving their tasks. The productivity in terms of the number of good quality user annotations was also high. This increased user performance could have allowed to construct better user models and it lead to the more precise system outputs, in terms of the high rank retrieved documents and the visualizations. The NE-based user models were able to contribute to formulating higher precision annotations by the users while maintaining the similar level of diversity and productivity with the keyword-based user models.

1.2 TARGET AUDIENCE

Even though the Adaptive VIBE system was implemented in the Web environment, we did not assume the everyday users could benefit from our approach immediately. Rather, we expected the users who are willing to investigate enough time and effort to rigorously solve complex problems using the visualization.

Therefore, we defined the target audience of dissertation as the information analysts who have the expertise in analyzing and solving complex problems. This premise was conformed to the entire study procedures, particularly during the design of the tasks and the participants recruiting stages.

1.3 DISSERTATION ROADMAP

Figure 1 shows the roadmap of this dissertation. In Chapter 2, related studies are introduced. They are organized in four sections: (2.2) adaptive visualization, (2.3) comparison of algorithm and UI-based information access, (2.4) Concept-based information access, and two (2.5) preliminary studies conducted by ourselves that are the basis of the current study.

Chapter 3 introduces the idea of the Adaptive VIBE visualization and NE-based user

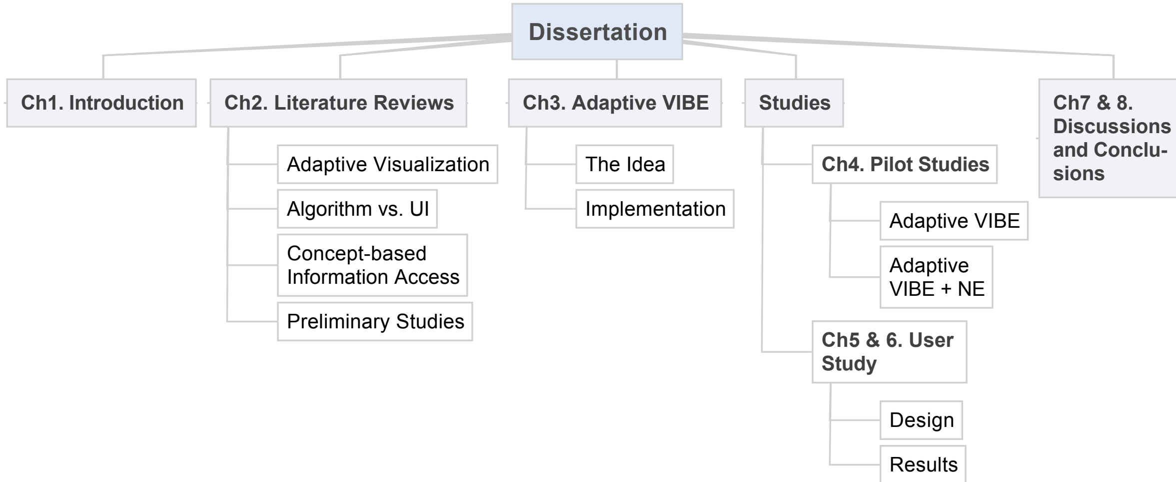

Figure 1: Dissertation roadmap

modeling that comprises the core of this dissertation. In Chapter 4, two pilot studies about the experiments that tested the potential of Adaptive VIBE and Adaptive VIBE with NE-based user models are presented. Chapter 6 shows the results of the user study. Chapter 7 and 8 discuss the meaning of the findings and conclude the whole study.

1.4 DEFINITIONS OF CONCEPTS

1.4.1 User Modeling

A user model is a representation of information about an individual user that is essential for an adaptive system to provide the adaptation effect, i.e., to behave differently for different users [18]. User modeling or user profiling has its root on information filtering [10], which filters relevant information from incoming stream of documents by comparing them with the user profile.

User profiles can be categorized as two groups according to their temporal coverage.

Long-term profiles reflect "long-term interests" that represents users' global or general interest. Short-term profiles reflect short-term interests of users that are more specific and cover shorter duration of user interests. By learning a short-term model only from the most recent observations, user models can adjust more rapidly to the user's changing interests [13].

They can reflect users' specific as well as general interests. Unlike the user models that are mostly concerned about the general characteristics such as a sports lover or computer scientists, some user models can take care of a very short-term task of users. They are called *task models*, which try to accumulate information about a specific user task [121]. The task models are more focused user models in this sense and they can also support multiple sub-tasks, as long as a user is focused on the specific super-task. Task descriptions can be provided explicitly [61] or implicitly by observing user activities [5].

The user model of this study is also the task model, because the adaptive visualization of the study was designed to support information experts who solve specific tasks. It does not serve their global attributes or interests.

1.4.2 Information Visualization

Card, et al [25] defined visualization as the use of computer-supported, interactive, visual representations of data to amplify cognition. They defined information visualization as the use of computer-supported, interactive, visual representations of *abstract* data to amplify cognition. Unlike other visualization area such as data visualization, information visualization focuses more on the visualization of abstract information.

Shneiderman [111] emphasized more on the interactivity of information visualization and defined it as a compact graphical presentation and user interface for manipulating large numbers of items possibly extracted from far larger datasets. Effective information visualizations enable users to make discoveries, decisions, or explanations about patterns (correlations, clusters, gaps, outliers,...), groups of items or individual items.

1.4.3 Named-Entity

Named-entities are information units important to recognition like names, including people, organizations, and location names, and numeric expressions including time, date, money and percent expressions from unstructured text [107].

1.4.4 Ranked List

We define ranked lists as the list of retrieval outputs of modern search engines. They are ranked according to the relevance scores computed by the system and usually include document titles, summaries, and links to the actual body of the corresponding documents.

2.0 LITERATURE REVIEWS

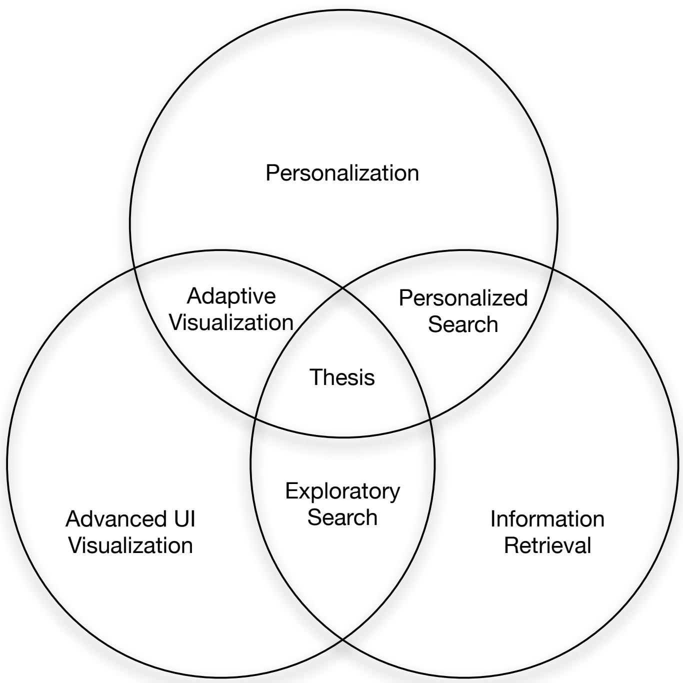

Figure 2: The Taxonomy of the related studies

The goals of this study are two-fold: (1) to introduce a novel adaptive visualization method and investigate its potential and characteristics and (2) to adopt a concept-based user modeling into the adaptive visualization. Among the various domains the adaptive visualization can be applied, this study focuses only on the visualization of search results. Therefore, we can summarize that it has its roots in three broader areas: (1) *Visualization and HCI*, (2) *Personalization*, and (3) *Information Retrieval* . Figure 2 depicts the relationships among these areas and resulting sub-fields. The current dissertation – adaptive visualization for information retrieval – is located at the center of the picture.

Adaptive visualization is an approach to provide customized visual information access methods to the users. It can present different visual layouts or cues to the users, depending on their context or different interests. We proposed an adaptive visualization method by extending a relevance-based visualization method called VIBE (Visual Information Browsing Environment) (Section 3.2) and attempts to test the effectiveness of this Adaptive VIBE in this study.

The concept-based user modeling tries to overcome the limitations of the old keyword-based user modeling. User model is a vital component of almost all personalized systems and they have been mostly using simple keywords (or dictionary words) in order to model their users. This simplicity entailed a lot of problems and various approaches have been introduced to overcome its shortcomings. In this study, a named-entity (NE) based user model was added to Adaptive VIBE in order to represent concepts in the user models.

Because this study chose the search result visualization for Adaptive VIBE, *personalized search* naturally arises as an important area to examine. It is an attempt to incorporate user contexts and interests in order to provide customized search results to the users [88], and forms a backbone of adaptive visualization that can act as an underlying engine for it.

In contrast to the personalized search that usually tends to rely on algorithms or artificial intelligence to produce the personalization, there exists another approach called *Exploratory Search* (Section 2.3.2), which tries to incorporate smart user interfaces into the information retrieval problem, so that the users can interact with the system to reach the better search results. It is distinguished from the personalized search, in that it tries to combine the algorithm and the user interfaces, in order to catch the "best-of-the-two-worlds."

Adaptive VIBE tries to combine the strengths of these two worlds. It incorporates personalization more actively into the visualization-based dynamic user interface and implements them into a single system. Its users can benefit from the flexible visual interface that is customized to their interests while they actively explore the search space.

This chapter introduces related studies that can work as the foundations of Adaptive VIBE. Section 2.1 overviews non-adaptive document visualization methods for search result representation, as they were frequently adopted as the baseline visualization layer by a lot of adaptive visualization methods. Section 2.2 introduces the previous adaptive visualization methods and their characteristics.

Section 2.3 compares the personalized search (Section 2.3.1) and exploratory search (Section 2.3.2) as two representatives of the algorithm- and user interface-based approaches. Section 2.4 introduces concept-based as well as named-entity based information access approaches.

At the end of the chapter, two preliminary studies are presented (Section 2.5). They are direct predecessors and motivations of the current study in terms of personalization and visualization: (1) YourNews study shows our own attempt for open and editable user models for news recommendation and (2) TaskSieve study introduces a personalized information retrieval system that had a flexible text-based user interface for query/user model mediation.

2.1 DOCUMENT VISUALIZATION OF SEARCH RESULTS

The approaches that attempt to visualize retrieved documents are mainly related to the lack of information of the classic one-dimensional ranked lists. Within the ranked lists, it is very hard for users to get enough information about the mechanism that actually retrieved the documents. For example, it is really difficult to understand which word in their queries benefitted more to match a specific document. Even though the ranked lists are provided with some basic information such as document scores, document summaries, and search keyword highlights, they still lack the capability required for understanding complex search tasks.

In fact, there have been a lot of attempts from the visualization side to better represent the search results. TileBars [51] is a good example that attempted to visually represent the term distribution in the retrieved documents graphically and could be understood as a visual extension to the KWIC (KeyWord-In-Context) [78] index-based document snippets in the ranked lists. A similar and more recent example is FeatureLens [37]. Its goal is to visualize the inner structure of a huge text and help users mine meaningful context, which was not easily comprehensible.

Whereas the approaches such as TileBars or FeatureLens focused more on enriching the document representations of the ranked list-like environments, some approaches tried to innovate the one-dimensional nature of the ranked lists and extend it to the two-dimensional space. Several approaches tried to exploit SOM (Self Organizing Map) based two-dimensional maps [29, 62, 65, 76].

Another spatial search result visualization approaches include force-directed layout based visualization, MDS (Multi-Dimensional Scaling) [27], relevance-based approaches such as VIBE [91], or RadViz [85], and InfoCrystal [115]. Among them, VIBE is relatively most famous and has been applied to various domains [11, 31, 54].

Beyond the rather simpler two-dimensional placement of the documents, hierarchical tree or clustering approaches have been applied to this domain. Examples are Vivisimo[1], Scatter/gather [53], Treemap [59], and Grokker [99].

Despite the variations in terms of the ideas and algorithms, they share a common goal – allocate the documents on the visualization plane and create visual cues for better information access. For example, SOM, force-directed layout or MDS allocate documents according to their similarity, so that similar documents are placed closer on the visualization and users can instantly understand the inter-document relationships. VIBE or RadViz exploit reference points and show the documents according to their relationships to the references. When these techniques are applied to the retrieved document visualization, users can visually find out the distribution of different topics within the hundreds of retrieved documents and then locate the information they want more efficiently.

However, they are different in terms of the interactivity with users. Table 1 summarizes how interactive these visualization algorithms are. Some techniques are just static, which means that they just show an overview about the data distribution but not include any interaction with the users. Other techniques include the capability to communicate with users, so that they can show the overview of the entire dataset fist but also can dig into the details according to users' requests [110]. I am specifically interested in the interactive visualizations among them, hoping that they could contribute to enhancing the interaction

[1] http://www.vivisimo.com

Table 1: Interactivity of visualization methods

Interactivity	Visualization Methods
Static	MDS, TileBars, Single-level SOM, Single-level clustering
Medium	Force-directed Layout, Hierarchical SOM, Hierarchical clustering, Treemap
High	VIBE, RadViz, Scatter/Gather

model of static search and implementing a personalized exploratory search system. The following section introduces some examples of the adaptive visualization methods.

2.2 ADAPTIVE VISUALIZATION

In this section, various adaptive visualization approaches are introduced. Adaptive visualization is the combination of visualization and personalization, which is one of the main thesis of this dissertation.

A classification of the adaptive visualization methods are provided in order to better understand the similarities and the differences between the approaches. By understanding the structure of the past approaches, we will be able to expect the strength of the novel method conceptually.

2.2.1 Classification of Adaptive Visualization Methods

Adaptive visualization is an attempt to improve information visualization by incorporating adaptation. Through adaptation, users can modify the way in which the system visualizes a collection of elements (or documents) [103]. Unlike the non-adaptive information visualization methods provide the same visualization to all users, adaptive visualization aims to provide different visualizations according to different user interests or contexts.

The non-adaptive document visualization methods have a wide variety of ideas and various adaptive visualization approaches have been introduced and tested in order to better support the users. Despite its relative new appearance compared to the traditional information visualization, the diversity of the approaches span quite large. Therefore, it would be worthwhile to systematically classify the adaptive visualization methods.

A related classification effort can be found in [21], where a family of adaptive content presentation approaches were introduced. Even though it is discussing about the more generic adaptive "presentation" of contents and not about more specific adaptive "visualization", it could provide us with the hints on how to classify the existing adaptive visualization methods.

The adaptive content presentation in [21] was classified into two big categories: (1) Content adaptation and (2) Content presentation. The former can be understood as the adaptation of contents itself. It includes sub-categories such as "approaches based-on page and fragment variants", "content selection", and "content structuring". The latter is to adapt the way the contents are presented and includes "approaches based-on abstract information", "relevance-based techniques", and "media adaptation." These two broad categories could be rephrased as "what to adapt" and "how to adapt (or adaptively present)."

This method cannot be directly applied to the adaptive visualization. However, it provided an insight to pick up two representative categories of adaptive visualization. Figure 3 shows a classification of adaptive visualization methods.

First of all, some adaptive visualization approaches prepare multiple visualization methods and provide them selectively according to different user characteristics. This is the simplest approach and we can notice the similarity to the "content selection" approach of the adaptive content presentation.

The second category is the visualization methods that adaptively change the structure of the visualization – "Visual Structure Adaptation". This is similar with the "content adaptation" method in the adaptive content presentation classification, in that it tries to filter out more relevant contents and to provide them in adaptively generated dynamic visual structures.

The third category is "Adaptive Annotations", which is mostly interested in emphasizing specific information using different visual annotation methods such as icons or colors. It is

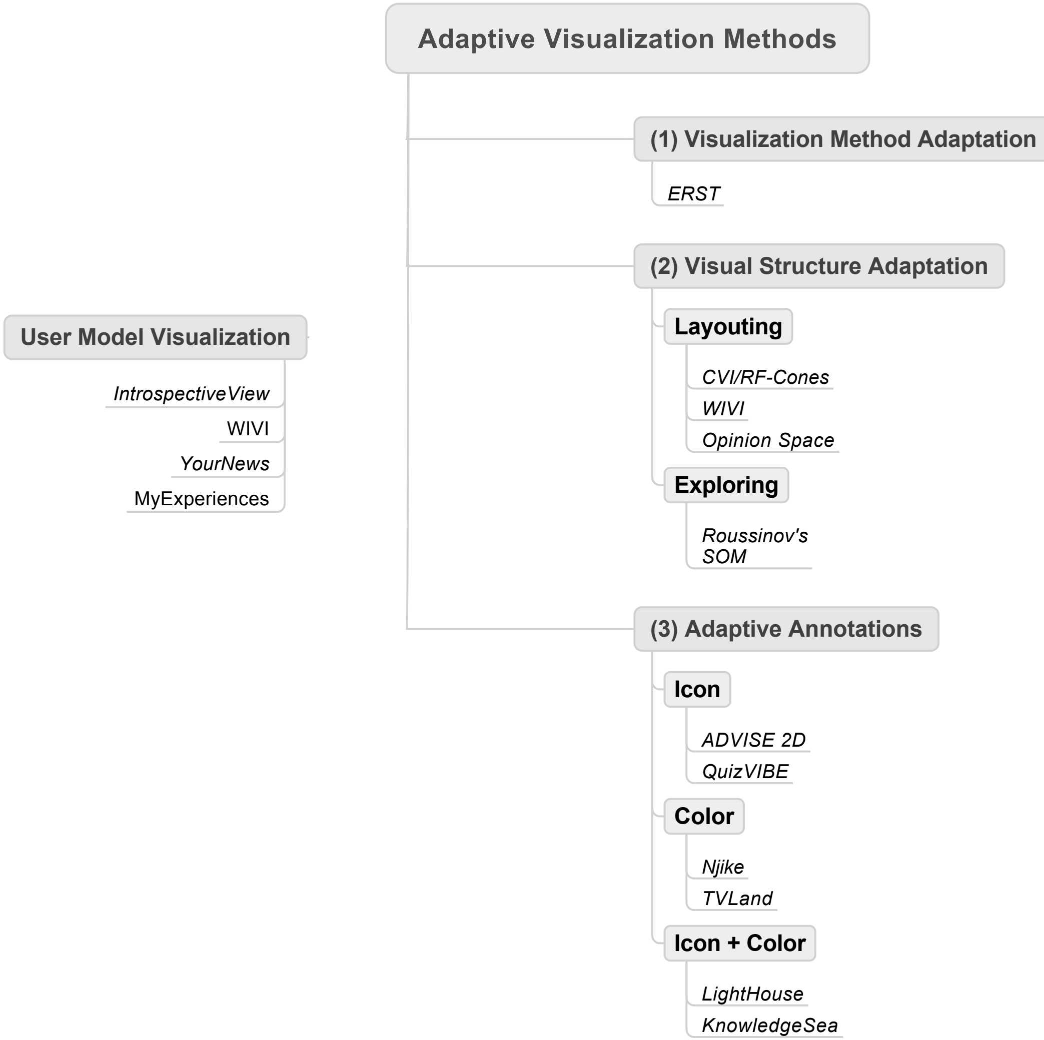

Figure 3: Classifying adaptive visualization approaches (italicized nodes are study examples)

similar with the "Priority on focus" or "Priority on context" adaptive presentation methods that are interested in emphasizing specific contents and/or preserving context of them.

The structure of these three categories can be understood as a *stack*. The first "Visualization Method Adaptation" stack forms the foundation of the other two. Any structure of annotation adaptation methods need to select the baseline visualization method first. The "Visual Structure Adaptation" is the parent of the "Adaptive Annotation" stack. The

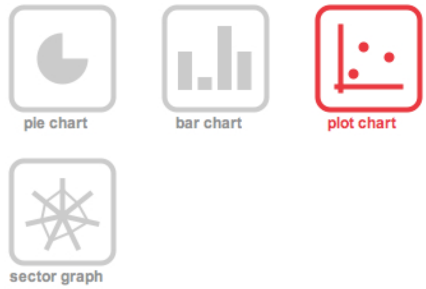

Figure 4: ERST's representation selection interface [47]

structure adaptation is inherited to the adaptive annotation, so that the icon and/or color-based adaptation can be made on top of it. The exemplar adaptive visualization methods belonging to each category shares the characteristics of the parent stack.

In addition to these categories, an independent group is defined – "User Model Visualization." The user models are core components for personalization in general as well as adaptive visualization. In most cases, they are hidden as black boxes and did not allow users' examination or control on them. However, some approaches tried to visualize the contents of the user models hoping that the openness and user control could improve the overall system performance.

The following sections examine the adaptive visualization examples of these four categories.

2.2.1.1 Visualization Method Adaptation Sometimes, a system can choose totally different visualization methods according to user ability or attributes, rather than make variations within a single visualization method. ERST (External Representation Selection Tutor) [48, 49, 47] could provide a selection of information display forms (plot chart, table, pie chart, sector graph, bar chart, Euler diagram) by users' background knowledge of external representations (KER) and task types. In a study [48], a Bayesian network was constructed in order to reveal the relationships between the background knowledge, task types, and the information display forms by collecting real data from the participants.

2.2.1.2 Visual Structure Adaptation The adaptive visualization methods that adapt the structures include two techniques: (1) varying the layout of the visualization or (2) providing methods to easily explore within the visualization. The first method shows different visualization layouts according to varying context of users and the second method receives user interactions and navigates to the different parts or levels of the visualizations.

CVI and RF-Cones [118] belong to the first group. They tried to help users to navigate the problem space with dynamically changing view points and similarity-based layouts. The users could explicitly feedback with the system in order to select a specific viewpoint that best matches their interests on a 3D sphere and then extend their navigations to the cone-tree based visualizations that layout the contents according to the similarities to the user profiles.

WIVI [72] is a adaptive navigation system for Wikipedia articles. It adopted two different visualization schemes for the user model side and the recommended element side. For the user model, which is the past visit history of Wikipedia articles, a simple graph visualization was used. For the recommendation of the future visits, concentric circles were used. The articles with higher recommendation scores were placed on the inner circles, whereas lower score articles were placed on the outer circles. Users were able to see different concentric layouts of the Wikipedia articles according to the different distribution of recommendation scores.

Opinion Space [14] visualized the user opinions on a 2-dimensional spatial visualization. Users are asked to answer a survey on a specific topic and then a visualization pops up. The high dimensional attributes in the survey was reduced to the 2-dimensional space by the PCA (Principal Component Analysis) algorithm and the users could see where their opinions are located in the visualization. The location of their opinions represented similarities to specific concepts and the users could navigate through the space by dynamically adjusting their original answers to the survey.

Roussinov [103] implemented a multi-level SOM (Self Organizing Map), so that users could explicitly explore the document tree implemented in the multi-level map. Unlike the approaches before, this approach is more focusing on the ability to travel through the complex visualizations rather than providing users with adaptively differentiated visualization layouts.

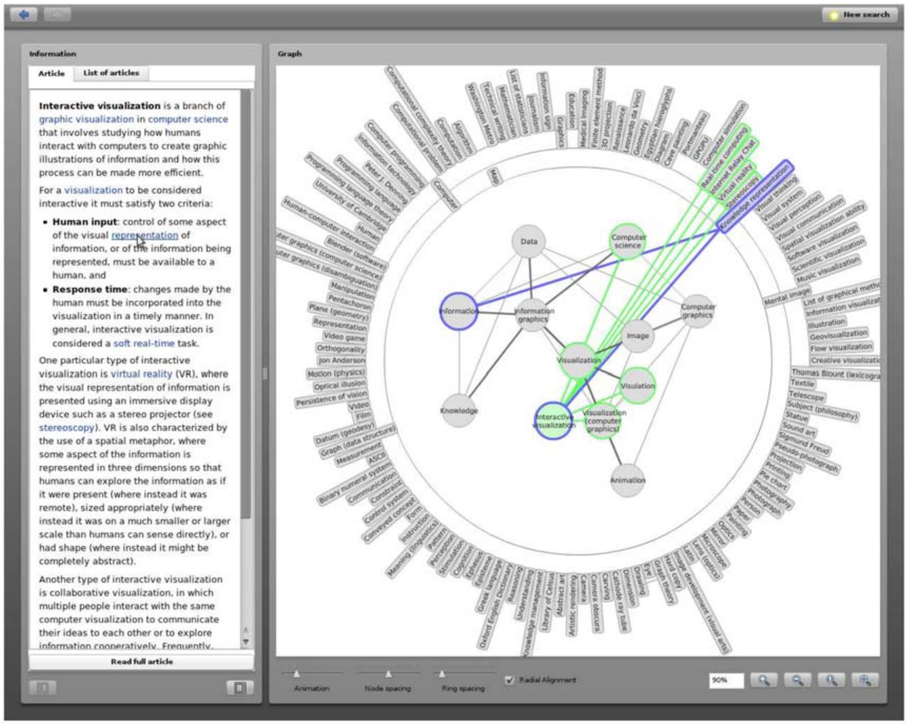

Figure 5: WIVI visualization example [72]. Visit history is shown in the center of the con-centric circles and recommended articles are located in the outer circles.

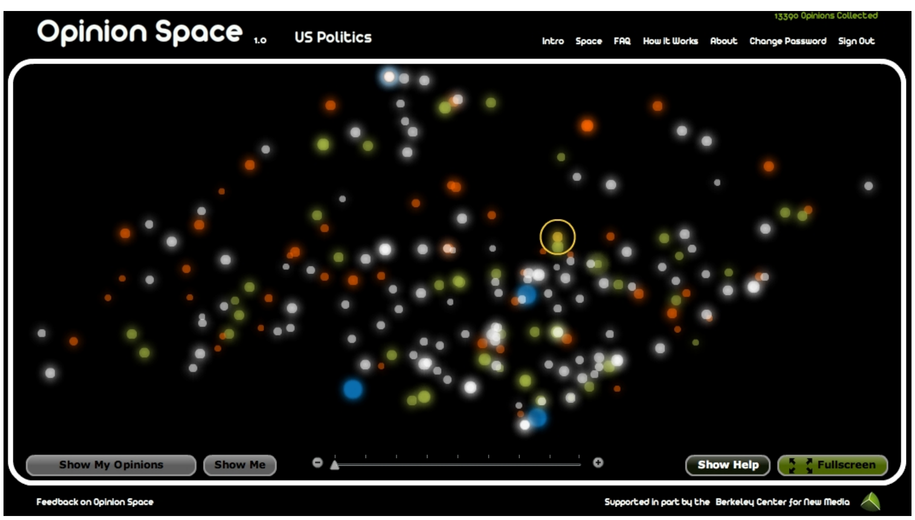

Figure 6: Opinion Space [14]

Table 2: Visual structure adaptation methods and their baselines

Adaptive visualization	Baseline visualization approaches
CVI/RF-Cones	3D SOM, Cone-tree
Wivi	Concentric circles, Radial tree
Opinion Space	PCA
Roussinov	SOM

The users could zoom in the specific cells in the map and could see more detailed child maps within the parent cells. By exploring down to the sub-maps, they could navigate down to the lower level in the entire hierarchy. Users could explicitly control the system using commands like "Please be more specific," "Please try to generalize more," "Fewer concepts, " and "Re-do it!" These explicit feedbacks could return adaptive SOM visualization to the users.

As can be seen from the examples, these adaptive visualization methods are based on the non-adaptive visualization approaches. Table 2 compares the visual structure adaptation methods and their baseline visualizations.

2.2.1.3 Adaptive Annotation Using the visual elements such as colors or icons, some adaptive annotation approaches tried to give more focus on a specific contents in the visualizations. ADVISE (ADaptive VIsualization for Education) 2D [17] implemented this approach based on the well-known Force Directed Placement visualization (FDP). FDP is a graph visualization method, where the nodes in the graph repel each other and the edges connects those nodes. Therefore, the layout of the visualization is determined by the repelling and attracting forces between the nodes and the edges [40].

On top of the graph layout constructed by the FDP algorithm, ADVISE 2D tried the adaptation technique by incorporating different set of icons. Because the corpus used in [17] was C-language education examples, the nodes in the FDP graph represented the educational examples connected to each other. The adaptation was achieved by displaying icons that represented if a student whether *ready* for a specific example. Therefore, the students could visually determine the examples to try by their progresses, even before examining the contents of the examples.

QuizVIBE [6] is a similar approach but the baseline visualization is different. It made use of the VIBE visualization [91], where the C-language quizzes were displayed according to their similarities to the C-language concepts. Different set of icons were annotated beside the quizzes in the visualizations and the icons were determined by the progress of the students.

Along with the icon-based adaptive visualization approaches, other systems tried colors in order to emphasize specific parts of the visualization while maintaining the context. Njike [90] implemented a treemap-based visualization in order to visualize the user visited pages and the concepts in the domain model. Different colors were used for painting the treemap cells according to the similarity between the user interest and the domain model.

Some visualization methods used the color and the icons at the same time. Gansner [42] used the FDP visualization again for adaptively visualizing TV programs. However, color-based adaptation was used instead of the annotation-based adaptation. The system called TVLand could separate regions of the FDP graphs and used the background color of the regions as the adaptation key. Because TVLand was the TV program recommender system, the color keys were used for showing the recommendation results. For example, regions with many recommended shows were colored yellow and regions with fewer recommendations were colored dark blue. At the same time, by using different font sizes, background color, and label frames of the TV show titles, the system could explain which programs were watched during the specific amount of time.

KnowledgeSea [19, 20] made use of the SOM visualization for the personalization and social annotation of the educational contents. It built a baseline SOM placing C-language tutorials on the 2-dimensional spatial map and personalized the color of each cell contained in the map according to the visit traffic of the users and the group of users. Each cell could represent these two different traffics using foreground and background colors, using high intensity colors for high traffics. Therefore, on the self organizing map that shows the key concepts in the C-language education domain, the users could see which concepts were covered by themselves and which concepts were visited by other group members. Along with

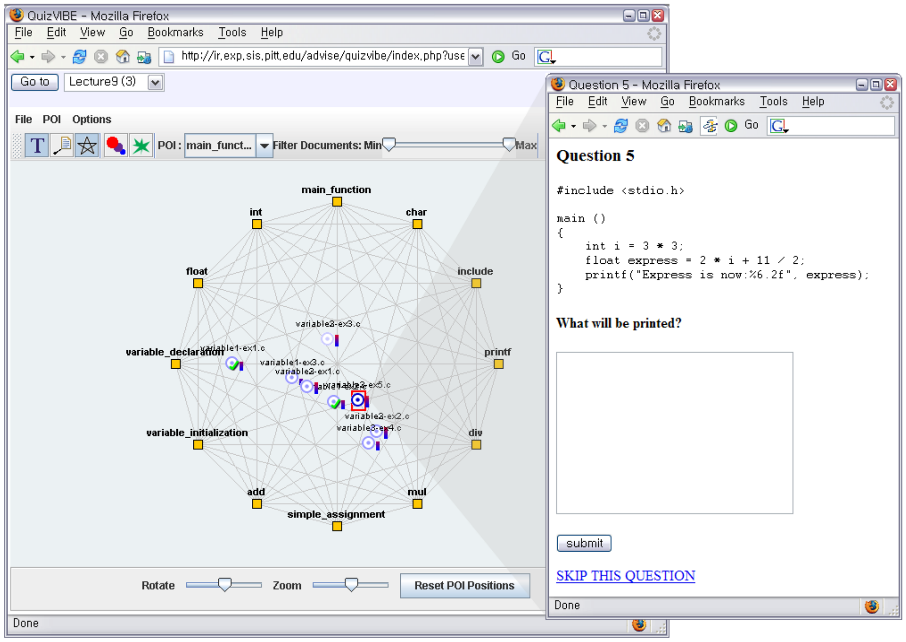

Figure 7: QuizVIBE visualization example [6]

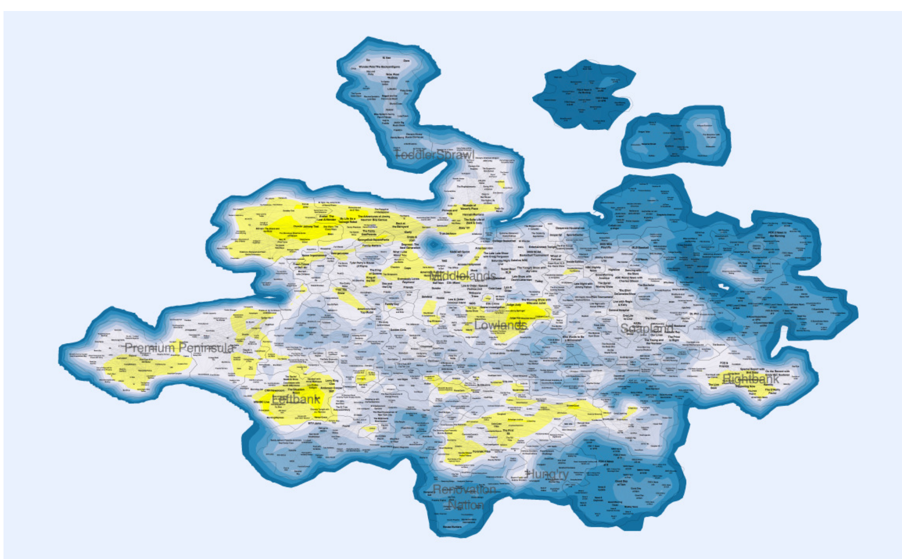

Figure 8: TVLand visualization example [42]

the traffic information, users' preference and notes were annotated using different sets of icons, so that the students could see the social recommendation and notes on the tutorials.

Lighthouse [73, 74, 75] introduced an interesting adaptive visualization mechanism for information retrieval. Users could give feedbacks to the system on retrieval results returned by their queries. The retrieved documents were displayed as circles in a 2- or 3-dimensional spaces where their positions were determined by their inter-similarity scores. The users could easily distinguish the contents of each document in the visualization because the document titles were displayed beside the visualization and visibly connected to the document icons in the visualization.

Users could annotate a part of documents as relevant or not relevant and then the relevance scores were calculated for the remaining documents. The estimated relevancy was marked on the document icons or textual titles using different colors and lengths of the colored-shades. For example, a very relevant document had a longer green shade as its title background and a non-relevant document had a red background. This color scheme (green to red) was overlaid on the visualization, so that users could see the distribution of the relevancy information and their relationships with the document positions.

Table 3 summarizes the adaptive annotation methods and their baseline visualization approaches.

2.2.1.4 User Model Visualization The last group of studies related to the adaptive visualization is the user model visualization. It is an attempt to show the contents of the user models to the users and sometimes even let them edit the user models, so that they could remove the noise in the user models and complement missing information.

YourNews [4] explored an on-line news filtering system that was equipped with a user model viewer/editor. Unlike the expectations, the result showed that the manipulation from the user side could lead to the poorer system performance. Wærn [123] has reported similar results regarding non-visualization user models.

WIVI tried the visual open user model using a graph visualization method. WIVI's past visit history viewing function presented in Section 2.2.1.2 can be seen as a simple example of graphical user model visualization. Users could compare the Wikipedia pages

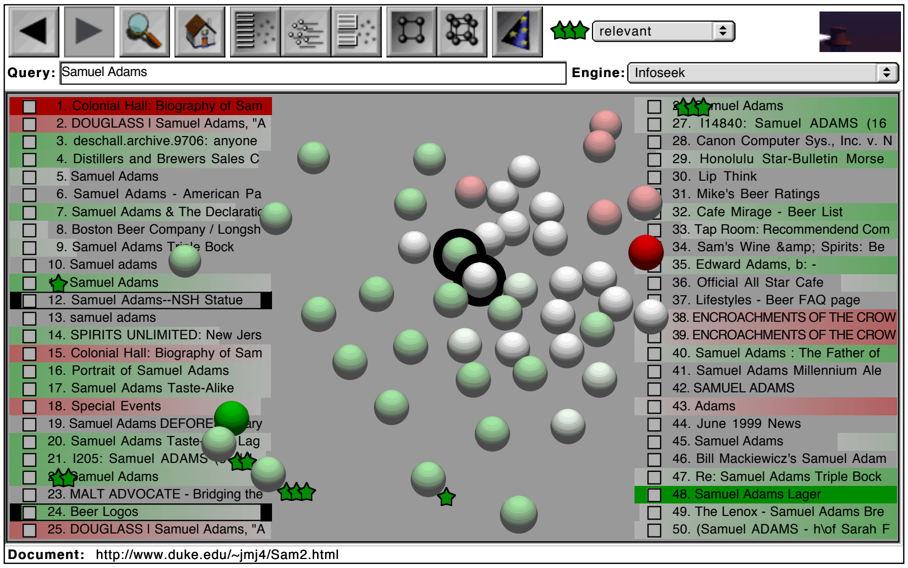

Figure 9: Lighthouse visualization example [74]. Relevant items are marked in green (lower-left side circles) and non-relevant ones are marked in red.

Table 3: Adaptive annotation methods and their baselines

Adaptive visualization	Baseline visualization approaches
ADVISE 2D	FDP
QuizVIBE	VIBE
Njike	Treemap
TVLand	FDP
Lighthouse	FDP (2D/3D)
KnowledgeSea	SOM

they had already visited in the user models and could compare them with the candidate pages recommended by the system in the concentric rings.

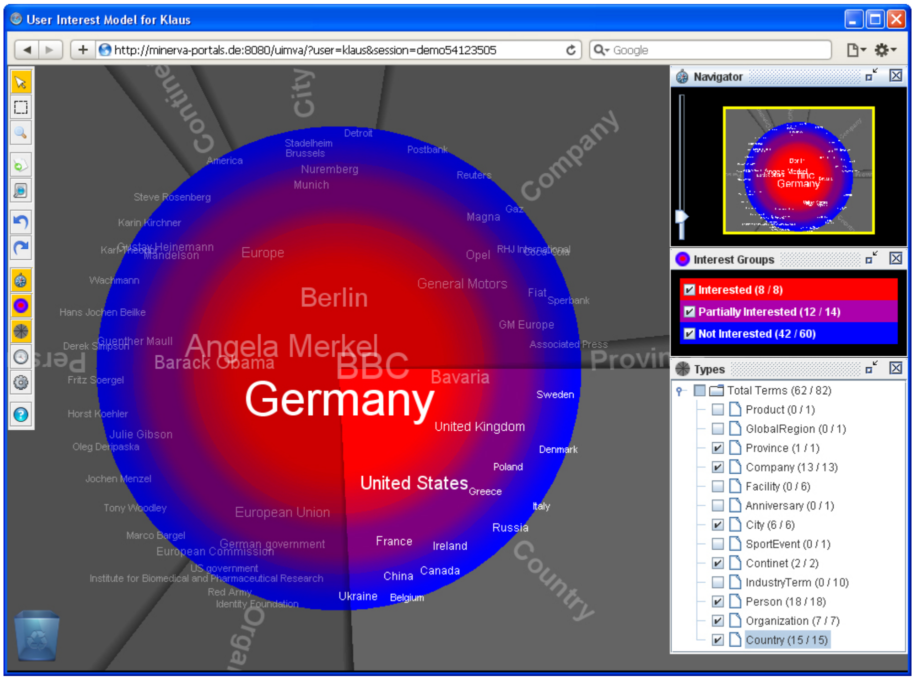

Figure 10: User model visualization example – IntrospectiveViews

IntrospectiveView [8] shows an even more evolved user model visualization. It visualized concepts in ontologies in a circle and used different levels of colors and font sizes according to user interests. Because it was based on ontologies rather than a random set of keywords, users could see the relationships among the visual user model contents (for example, the different types of the elements). Figure 10 is a screenshot of IntrospectiveViews[2].

MyExperiences [69] visualized the open learner model (OLM) in order to let the user of adaptive learning systems see their user models and the process to construct them. The learner model was represented as a tree structure and therefore the treemap algorithm was used for the implementation of the open learner model. Using various interaction methods such as selection and zooming in the treemap, users could navigate through the open learner model in order to learn about the structure of their learning model.

[2] http://www.minerva-portals.de/research/introspective-views/IntrospectiveViews-v.2

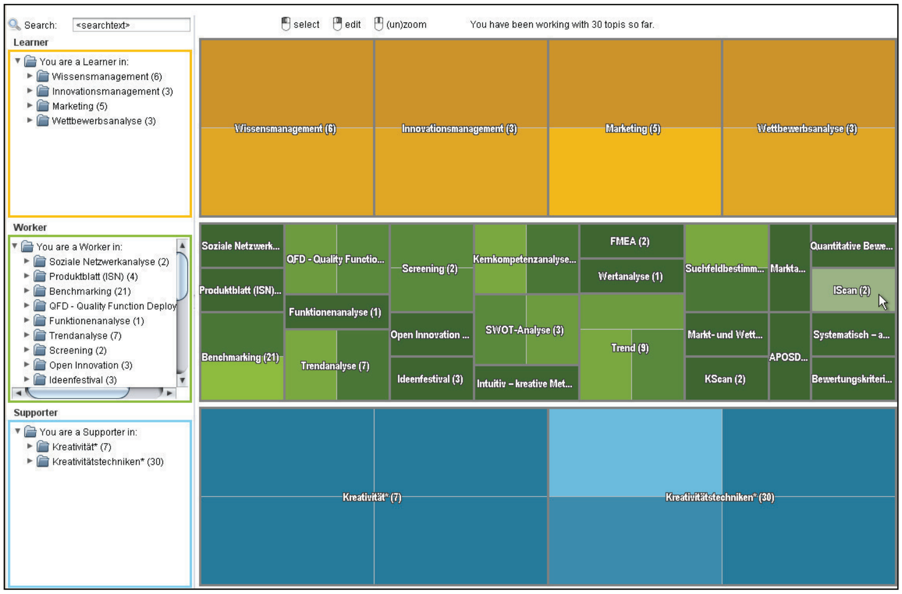

Figure 11: User model visualization example – MyExperiences [69]

2.3 ALGORITHMS VERSUS USER INTERFACES FOR INFORMATION ACCESS

Adaptive VIBE is an adaptive visualization approach designed for the efficient information access. It aims to help users visually explore the target document space for problem solving. The search process is reinforced by the system that understands the users' search contexts. Therefore, it can be seen that Adaptive VIBE inherits strategies from two rather contradicting approaches: (1) approaches to provide users with personalized search results by strong algorithms (*personalized search*) and (2) approaches that allow users to interact with the system so that they can explore the problem space and locate the relevant information (*exploratory search*).

They are contradictory in that the former approach usually puts more focus on the personalization algorithm whereas the latter puts more emphasis on the system-user interaction. Of course, the long history of general information retrieval has a very strong tradition in algorithm-based problem solving, for example, the Cranfield test [32], SMART experiments

[106], and the TREC[3] evaluation framework. However, this study limits the discussion to the personalized information retrieval only.

2.3.1 Algorithm-based Approaches – Personalized Search

Personalized search is defined as an attempt to provide individualized collections of documents to the users, based on some form of models representing their needs and the context of their activities [88]. The personalized search results are tailored to the preferences, tastes, backgrounds and knowledge of the user who expressed it with the form of queries. Compared to the approaches that impose more focus on the interaction between the user and the system, personalized search has more focus on algorithms to generated the personalized results.

In order to understand the difference, it is worthwhile to examine the mechanisms by which the personalized search is working. Pitkow [95] described two methods for personalized searching: (1) query augmentation and (2) result processing. In the query augmentation stage, the user query is extended by the system considering the context of the user search. After the search engine retrieves documents based on this augmented query, the search result is examined and modified again to better reflect user context.

The user queries play a significant role in the information retrieval process. It is the starting point of the whole retrieval process and works as the primary mean for sending user intentions to the systems. However, it is well known that users are not good at formulating effective queries. They tend to provide very short (two or three terms) queries [57], which are not enough to fully express their information needs and sometimes miss important contexts. Personalization can help augment or expand the user queries by estimating users' latent contexts or interests and eventually provide better results.

Most modern search engines rely on ranked lists in order to present the search results to the users. They tend to rank the retrieved documents by their relevancy scores and display 10-20 documents per pages. Users mostly lack time and effort to examine a lot of documents returned by the systems and it is a very important issue to place the relevant documents on

[3] http://trec.nist.gov

top of the lists. Again, the system can estimate the user contexts and reprocess the original search result in order to move more relevant documents into the top of the lists.

These two stages were all integrated into a single system in Pitkow's Outride system [95] but they can be selectively adopted by individual systems too. That is, a system can achieve personalization by query augmentation or by result processing only.

Pretschner et al. [98] and Micarelly et al. [88] provided similar schemes to Pitkow's. They categorized the personalized search algorithms: (1) re-ranking, (2) filtering or part of retrieval process, and (3) query expansion. Query expansion is same with query augmentation of Pitkow's terminology but they elaborated the result processing stage as "re-ranking" and "filtering or part of retrieval process."

A lot of personalized search engines adopted the re-ranking or the filtering method [7, 16, 95, 98, 109, 114, 116, 117] while query expansion or augmentation is still widely used [82, 30, 109, 95, 77]. Our own previous studies contributed to the both categories. For example, YourNews [4] and TaskSieve [5] systems all incorporated the re-ranking and the filtering algorithms. Even though it was not a personalized system, NameSieve [3] could support the pseudo-personalization by using a named-entity selection user interface and filter by the users' selection of the named-entities. It could provide the query augmentation with they help of the user interface too. Detailed descriptions about those systems are provided in Section 2.5.1, Section 2.5.2, and Chapter 3.

2.3.2 User Interface-based Approaches – Exploratory Search

In contrast to the approach focusing on algorithms such as personalized search, *exploratory search* is more interested in users' ability to control, learn, and discover information during the interaction between the system and the users. It greatly emphasizes on interactive user interfaces for information retrieval and understands the information retrieval process as learning or investigation process rather than the simple lookup search [83].

Unlike the lookup search where the users relatively know well what to search, the target space and the nature of the problem of exploratory search is uncertain. White et al. [124] identified the uncertain situations as: (1) the target of search is unknown or partially known, (2) the search begins with some certainty about what is known but changes into one unknown and unfamiliar on exposure to new information and (3) users recognize useful information objects by scanning through information resources, evaluating their usefulness and determining the content/structure of a set of information objects.

Therefore, it requires strong human participation in an iterative and exploratory process. In order to encourage the user participation and the active communication between the user and the system, the role of efficient user interface is highly praised.

Relevance feedback [82] can be viewed as a technique that complements user exploration of the search space [124]. It learns about user interests by their feedbacks [104] and the relevance information acquired from the feedbacks is used in order to improve the result for the next round of search process. Many relevance feedback approaches have been confirmed to increase the performance using variety of techniques.

Rocchio's relevance feedback model [102] or Ide's Ide-dec-hi modification [56] update the query vector regarding the positive or negative feedback of users about the retrieved documents. Probabilistic models make use of user feedbacks in order to update its relevance probability estimation [101, 120]. Users can provide the relevance information implicitly [64], explicitly, or the systems could perform the similar process even without any user feedback by using the technique called pseudo-feedback [34].

However, it is still relying on the algorithms, rather than the active participation of the exploratory search idea. Especially, the implicit feedback based approaches or the pseudo-feedback method requires almost no user participation. Moreover, a lot of traditional relevance feedback model is based on the classic ranked list turn-taking interaction model, so it is not very natural for users to see the result of their feedback immediately and understand what aspect of their feedback helped to update the retrieval results.

A lot of exploratory search approaches tried to overcome these limitations by applying various advanced user interface technologies. TileBars [51] graphically presented the query term distribution in the search results so that the users can easily understand the relationships between the query terms and the retrieved documents. Flamenco [52] and Relation Browser [24] adopted a facet-based user interface for sub-exploring the search results. Scatter/Gather [53] introduced a cluster-based search interface. Open Video [84] and PhotoMesa

[112] enabled easy exploration of multimedia resources. HARVEST [46] used visualization features such as maps and timeline browsers.

We tried to incorporate the idea of this exploratory search into Adaptive VIBE. VIBE is a highly interactive visualization algorithm and its users can explore and investigate the search space flexibly. We also tried to combine the personalization to the VIBE-based search system, so that we can achieve the advantages of the two worlds. Detailed conceptual background and implementation information is provided in Chapter 3.

2.4 CONCEPT-BASED INFORMATION ACCESS

Traditional textual information access approaches (including user models for the personalized information access) were mostly based on the idea called *bag of words*. It is a keyword-based approach that separates all *words* appearing in texts (including queries, documents, or user models) and puts the words into a specific representation form (*a bag*). Among the diverse representation methods (or models) of texts, vector space model (VSM) [105] or probabilistic models [100] have been regarded as standards and showed relatively good performances.

When the words are placed into the vectors or their probabilities are calculated, it has also been a standard procedure to remove very common words called "stop words" and stemming them into their root forms [97, 67]. Depending on the models, a specific weighting function such as TF-IDF (*term frequency* and *inverse document frequency*) is applied or the probability of the words are used [105, 82].

This keyword-based approach has a long history of popularity and showed good performances. However, it has many shortcomings. Because the documents, phrases, and sentences are broken into independent words, the relationships among them instantly disappears. At the same time, the meaning vanishes. For example, *the bear ate the human* and *the human ate the bear* are treated equally in the keyword-based bag-of-words models [87] and the difference of the meaning between the two sentences disappears. The shortcomings of the keyword-based information access methods can be summarized as follows.

1. Polysemy and synonymy problem – a single word can have more than one meaning

(polysemy) and multiple words can have the same meaning (synonymy) [122]. The conventional bag of word approaches do not distinguish these differences and it leads to the performance degradation of the information access systems.

2. Independence assumption of words – all models that are based on the bag of words assumption lose the notion of term ordering [87]. The lost term ordering leads to the loss of meanings.

Therefore, various ideas have been introduced to overcome these limitations. Adaptive VIBE is one of those attempts, which tried to implement a concept-based user model that can provide a better representation of user interests. In the following sections, some examples of the concept-based information access approaches are introduced, including concept-based user modeling methods, document modeling, and concept-based information retrieval algorithms.

2.4.1 Semantic User Modeling

Gauch [44] defined two user profile approaches that tried go beyond the old keyword-based approaches. They are "semantic network profiles" and "concept user profiles." Semantic network profiles stored concepts in the nodes of the weighted network structures and retained the connectivity information among the nodes. Therefore, semantic networks could overcome the unrealistic assumption of the keyword-based approach that they were semantically independent and could express more abundant meanings. By incorporating the concepts' inter-relationships, the network user profiles could resolve the word ambiguity problem of the terms that had been originated by polysemy and synonymy of keywords. Examples of this method include InfoWeb [45] and SiteIF [79].

Concept profiles are similar with semantic network profiles in that there exist the links between concept nodes. They made use of the concepts represented as nodes in a tree to represent user interests rather than simple independent keywords and considered the relationships among the concepts.

One important characteristic that differentiates concept profiles from the semantic network profiles is the existence of hierarchical taxonomy or ontology. Semantic network links among the concepts are flat (node to node as in SiteIF) or close to a flat level (planet and satellite in InfoWeb) but the hierarchies adopted in concept profiles are much richer because they usually incorporates already existing hierarchies.

Some examples of these ready-made hierarchies are ODP (Open Directory Project)[4], Yahoo Directory, Google Directory, or ACM Topic Hierarchy. Ontologies can also be used for the hierarchy instead of the simple taxonomies and they have strength to express more diverse relationships of the nodes, compared to simple "is-a" or "has-a" relationships of the taxonomies.

Examples of this approach include OBIWAN [98], Persona [26], PVA [28], and Smart-Push [60]. Daoud and others introduced another attempt of ODP-based semantic graph for matching documents and user models [35] and Semeraro et al. used WordNet for representing documents [108]. Another interesting example is IntrospectiveView [8] introduced in the previous section. Even though it focused on the open/visual user models, it also tried to utilize ontologies in order to categorize the keywords included in the visual user models.

2.4.2 Named-entity based User Modeling

The advanced user modeling methods introduced in the previous section mostly emphasize on the structure of the user models. Both try to improve the simple keyword-based approaches by connecting the elements either in the form of networks or hierarchies and eventually elevate them to the concept or semantic level.

Along with the structure of the user models, we can consider the contents to be stored in the user models too. In many cases, the contents of the advanced user models are still the "dictionary words." They are single dictionary words and mostly stemmed to their root forms and lose significant capability to deliver meaning morphologically. It is a critical shortcoming especially for interactive information access systems.

One of the most possible alternatives to the simple keywords is the named-entities (NEs). They are information units important to recognition like names, including people, organizations, and location names, and numeric expressions including time, date, money and percent

[4] http://www.odp.org

expressions from unstructured text [107].

Even though they are called named-entities and many of them are actual name of people, location, or things, they are not just names. Rather, they are semantic categories and a pointer to a real world entity such as city, an organization, a movie, a book, or a historical event [93]. They are much richer in semantic content than simple vocabulary words and particularly useful for improving performance for news detection and cross-language retrieval [68, 80]. Mihalcea [89] discussed the idea of using named-entities for indexing document content, and they found that the size of the index can be greatly reduced whereas relevant documents still can be retrieved.

To our knowledge, there has been no attempt to directly incorporate named-entities for user morel construction. However, name-entities have been successfully adopted by analytic systems such as [12], where user interaction and feedback plays a key role as in the personalized information access systems.

In our previous attempt [3], we were able to construct named-entity based pseudo-faceted search system and could provide better information access method compared to the baseline information retrieval system. Therefore, we can expect named-entities as high quality semantic representation methods and can use them for enhancing the user model representation.

In Chapter 3, the named-entity extraction method, an example named-entity extraction, and the usage of the named-entities in the user modeling for Adaptive VIBE are introduced.

2.4.3 Concept-based Document Modeling

LSI (Latent Semantic Indexing) [39] tried overcome the simple keyword-based document modeling approaches by extracting concepts from the target documents. In the keyword-based document modeling, the document spaces are represented as term (keyword)-document matrices. Usually the dimensionality of these matrices are quite large, because each dimension represent each keyword.

LSI decomposes these matrices using the linear algebra technique called SVD (Singular Value Decomposition) and reduces the term space. The resulting reduced term space becomes a concept space.

Hofmann [55] introduced a probabilistic version of LSI, called PLSI (Probabilistic Latent Semantic Indexing). LDA (Latent Dirichlet Allocation) [15] is another probabilistic approach for discovering semantic meaning from the textual resources, but it focuses on extracting topics. Even though there is a view [87] that these approaches are still relying on the bag of words assumption, it is also clear that they can be differentiated from the traditional keyword-based bag of words and have advantages to discover more meanings than mere keywords.

2.4.4 Language Model-based Information Retrieval Approaches

Due to the recent popularity of the probabilistic language modeling in the field of information retrieval [96], various approaches belonging to this family attempted more semantic approaches beyond the simple keyword-based models. It is not possible to cover the entire field, so some of the examples are briefly introduced in this section.

Language models calculates the probability to generate a query given a document, $P(q|D)$ and it is decomposed to the product of the probability of each keyword. The simplest language model exploited the unigram model, which assumes that the probability of observing each keyword of the document is independent to other keywords, and therefore equivalent to the bag of word assumption.

Gao et al. [43] tried to relax this unrealistic assumption by proposing a bigram model which word dependencies were not restricted to adjacent words and only considered the strongest word dependencies.

Cao, et al. [23] introduced another approach for relaxing the unigram or term independence assumption. This study focused on the use of manually built information source, WordNet, in order to avoid the noise inherent in the automatic word co-occurrence analysis approaches. Also, this thesaurus approach had another advantage, that is, it could uncover some relationships among terms, which was hardly discovered by automatic approaches.

Metzler and Croft [86] connected the InQuery inference network [22] and language modeling approach in order to extend the structured query framework of InQuery. InQuery relied on Bayesian networks for the structure queries, which could express the proximity of query terms (mostly phrases) and could combine the terms or phrases related to each other such as synonymous terms. Because the structural representation is on the query side, this model cannot produce automatic combination of the keywords or phrases. However, it surely is a good example to represent the connected keywords or phrases in the Bayesian networks and support the semantics among them.

2.5 PRELIMINARY STUDIES

The main goal of this study is provide an adaptive visualization method that can help users to explore the search space without losing the context. Therefore, it combines the three worlds – *visualization/HCI*, *personalization*, and *information retrieval* – in order to maximize the user control in the personalized information exploration environment.

This section introduces two preliminary studies in the same context. They aim to provide interactive user models in the form of "open and editable user model" (YourNews study [4]) and the "query-user model mediation" (TaskSieve study [5]). Even though they lack the graphical visualization system such as VIBE, they tried to maximize the potential of the text-based user interfaces in order to implement personalized exploratory search environment.

2.5.1 YourNews Study

YourNews is a content-based news recommender system. In order to sort and filter relevant news stories relevant to individual user's interests, it builds user models representing the interests by observing user interests. The recommendation process is done by comparing the new incoming news articles and these user models. The user models are comprised of vectors of keywords and they are open to the users (Figure 12).

They even can edit the contents of the user models with a simple user interface. Figure 13 visually explains how to edit the user model contents. Users can disable the part of the user model (keywords) or can add a new keyword to it. The change of the user model is reflected

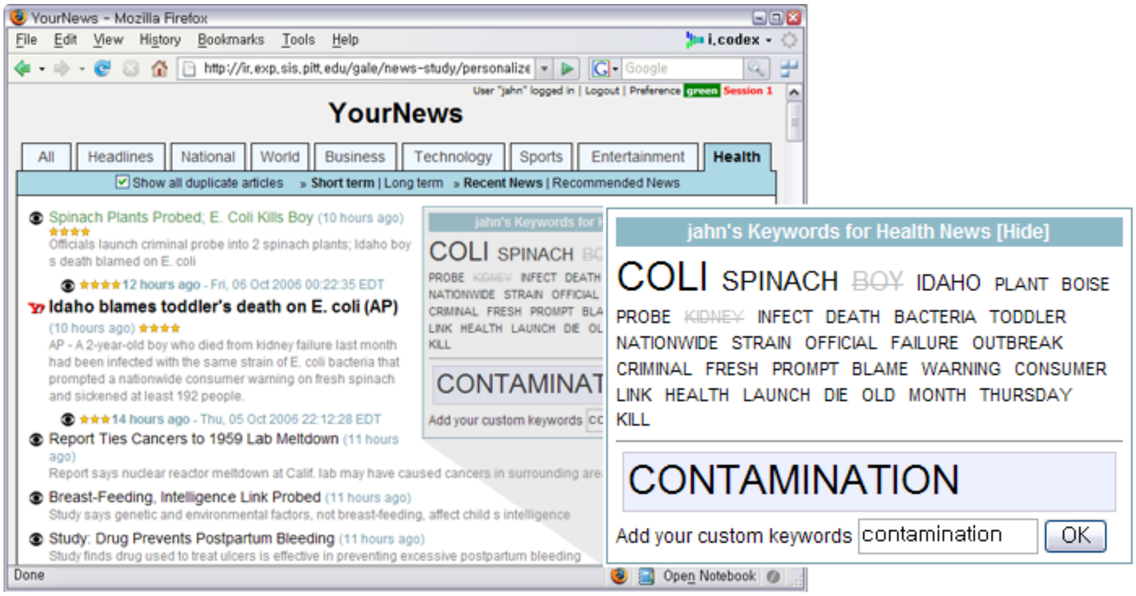

Figure 12: YourNews system and the "open user model"

to the recommendation results on the fly, so that the users can test and explore the effect of the control on the recommendation.

The news articles are organized into nine topical tabs and each topic has its own user model. Along with the topics, different spans of user interests are implemented into two different types of user models – short-term and long-term user models. Users can freely switch between different topics or between short-term and long-term interests.

2.5.1.1 Evaluation and Findings A user study was conducted with 10 information science graduate students. Two topics were chosen from events that occurred between September 25th, 2006 and October 15th, 2006: "School security (National)" and "Food contamination (Health)."

The subjects were asked to search for information about these two topics using the YourNews system and the baseline system, which is a stripped down version of YourNews with the open user profile feature disabled. They were also asked to save the search outcomes for the performance analysis. The precision and recall of the ranked-lists generated by

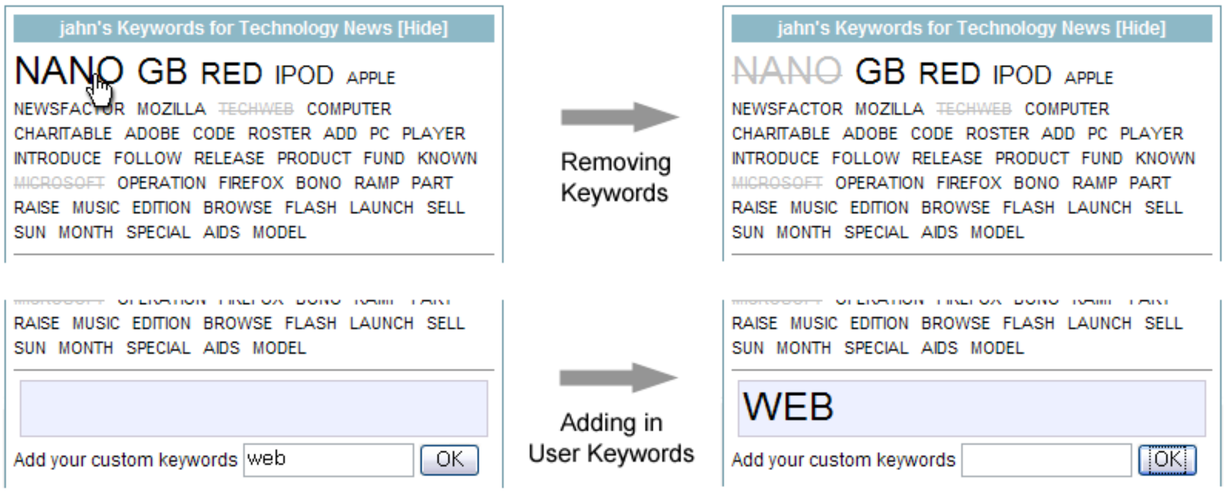

Figure 13: Open user model and how to edit it

the systems and the users' final search report were calculated per each system against the groundtruth information encoded by three independent study coordinators.

However, the experimental system with the open user model showed poorer results compared to the baseline, in terms of both precision and recall, for all tasks. Unlike our expectation, the keyword manipulation of the open user model harmed the performance of the YourNews system. By comparing the different user model manipulation actions, "removing" keywords harmed the performance about four times more than "adding" keywords.

2.5.1.2 Lessons Learned The core of YourNews was the editable open user model and the goal of the study was to prove if it could contribute to improving personalized news recommendations. The open user model showed short/long term user interests per news category and let the users freely modify its contents in order to reflect their interests more precisely.

However, the result of the user study contradicted to our expectations. The editable nature of the open user model caused the decline of the recommendation performance. Overfitting was speculated as the source of this contradiction. That is, the user model was optimized too much to the local user interests and lead to the global performance drops.

Despite this problem, the idea of transparent user model was accepted positively. Moreover, more flexible user modeling methods that can avoid the overfitting issue would still be able to contribute to the system performance. This observation lead to the development of the systems with alternative user model control methods, TaskSieve and Adaptive VIBE.

2.5.2 TaskSieve Study

In a prototype system called TaskSieve (Figure 14), the contents of the user models were visible to users and let users mediate between different methods for mixing the effects of the user query and the user model. Users could select one of three ranking methods that consider: (1) query only, (2) user model only, and (3) query and user model at the same time with equal weights. They could mediate among these three settings using a simple tab-based interface, sensing how their query or the user models contributed to generating different search results.

TaskSieve originally aimed at supporting information analysts rather than everyday users, so it has a special tool called a "shoebox" or a "notebook", which is a core component for information analysis and foraging [94]. In the shoebox, users are able to save short text fragments they consider to be containing important information while they are conducting the search tasks. Therefore, the system can naturally observe the content of the information in the shoebox and then estimate the interest or the context of the user's search task.

The user model in TaskSieve is called as a task model, which means a "user model for a specific task." Therefore, it does not represent users' general interests or their characteristics but it contains information about the task the user is trying to solve for a special session of information retrieval. It contains such information as "the name of a governor where a train accident occurred in Austria", which is a specific task, rather than whether the current user is a sports fan or a student studying information science.

When the task model is constructed, it is used for re-ranking the initial search result. By default, the system gives same weights to the query and the task model (50 to 50). However, it allows users to select different weights clicking on one of three tabs. Therefore, the users can get personalized search results (list of documents that are re-ordered by the re-ranking process) that better reflects their interests in terms of their tasks to be solved rather than the non-personalized and non-contextual document lists (before re-ranking) with higher control over the personalization algorithm.

TaskSieve displays the user model and user query information in the document summaries too. The document summaries of TaskSieve are called as Task-Infused Snippets (Figure 15). Traditional search engines use a mechanism called KWIC (Keyword-in-context) [78] that highlights the user query terms and shows surrounding sentence fragments around them in order to represent the context where the query terms appeared. Task-infused snippets infuse the task model information into these KWIC style document surrogates. It highlights the task model terms as well as the query terms in different colors. Moreover, the sentences to be displayed in the task-infused snippets are selected adaptively, considering the user-selected task model weights. That is, if the user puts more weights on the task model side, the snippets show sentenced that are more similar to the task models.

2.5.2.1 Evaluations and Findings To assess the value of TaskSieve's task modeling and the query/task model mediation features, a user study was conducted using a full-fledged version of TaskSieve as the experimental system and a normal search system as a baseline. The TDT4 (Topic Detection and Tracking Phase 4)[5] news corpus that was expanded by He and others [50] was used as a test dataset. It was adapted to the intelligence analysis program called GALE (Global Autonomous Language Exploitation)[6] supported by DARPA.

Ten subjects recruited from the School of Information Sciences at the University of Pittsburgh participated in the experiment. To ensure they best fit the profile of an information analyst, participants were required to be native English speakers and have been graduate students trained in search (i.e. a graduate course in information retrieval.)

The evaluation metrics used in the study included system performance and user performance measures, which were based on document and passage level precision and the usability

[5] http://projects.ldc.upenn.edu/TDT

[6] http://www.darpa.mil/ipto/programs/gale/gale.asp

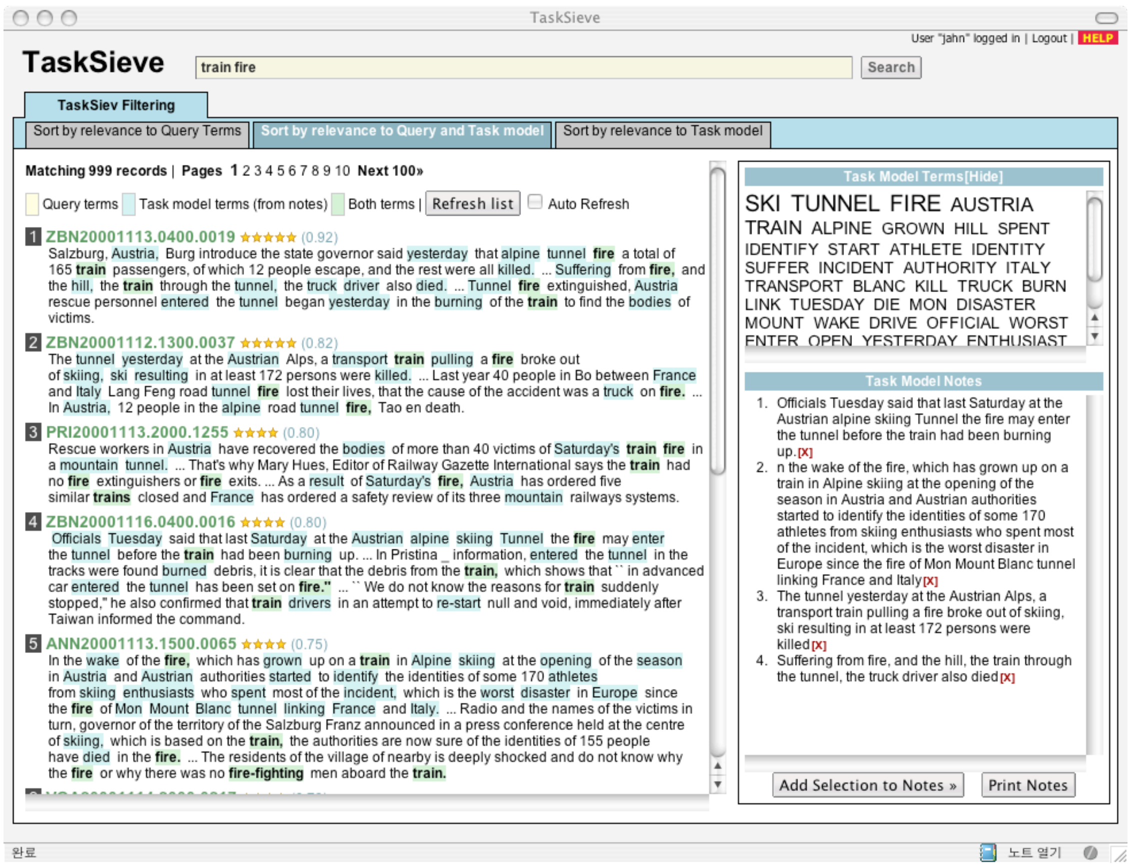

Figure 14: TaskSieve interface

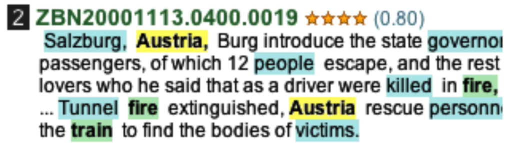

Figure 15: Task-infused snippet example

measures regarding the systems' support in task-based exploration processes, especially those examining the interactions between the users and the systems.

The results showed that TaskSieve was better providing high precision ranked lists and guiding users to open relevant documents. However, unlike these two system performance measures, the user performance measure which was the average precision of the final product of the study – notes taken by the subjects – did not show any difference between TaskSieve and the baseline system.

The source of this discrepancy was analyzed as users' ability to filter out irrelevant information provided by the baseline system. That is, even though the baseline system made more errors than TaskSieve, users could discard those errors and improve the average precision to the level of TaskSieve. An evidence of this analysis is the user productivity. High quality (precision) user notes of the TaskSieve outnumbered those of the baseline system. The users of TaskSieve could produce more high quality products than the users of the baseline.

2.5.2.2 Lessons Learned TaskSieve introduced an interface that could mediate between the user query and the user model. It used a tab-based user interface to control the importance of the two components for retrieving personalized search results. This mediation was also applied to the document snippets, using different color highlights for the keywords either from the query or from the user model. Users could know which component – their queries or user models – contributed which part of the retrieved results and then could control their effects on the search results.

The user study revealed that the TaskSieve system could provide users with more precise results than the baseline search engine. Even though the average precision of the users' final annotations was not able to beat that of the baseline, it was shown that the TaskSieve users could collect more high precision notes than the baseline.

The success of TaskSieve encouraged us to move forward to the stronger user interface that would give higher control and more flexibility to the users. Therefore, in this dissertation, we present the interactive visualization-based adaptive information access method, Adaptive VIBE.

3.0 THE ADAPTIVE VIBE TECHNOLOGY

This chapter introduces our novel adaptive visualization method, **Adaptive VIBE**. The concept of the adaptivity of Adaptive VIBE and the implementation details are also presented.

The two preliminary studies introduced in the previous chapter, YourNews (Section 2.5.1) and TaskSieve (Section 2.5.2) share the same goal – to provide a personalized environment that the users can understand the mechanism of the system and flexibly explore and control it using an advanced user interface. They tried to combine the two different approaches: personalized search and exploratory search, which are focusing on the personalization algorithms or the user interfaces respectively.

YourNews attempted to show the contents of user model visually and let users edit them in order to improve the quality of the news recommendation. In TaskSieve, the users were provided with open user model again but a different method was used for allowing them to control the personalization process. This time, they were able to change the importance of the two core element of the personalized search: user query and user model.

Both studies showed the potential and the limitations of the ideas and we decided to move forward to this stream of idea. Adaptive VIBE is the next step of the personalized information access methodology that combines the algorithm and the user interface. Even though the previous approaches – YourNews and TaskSieve – exploited visual and interactive user interfaces, they were based on the rather older paradigms.

1. **Text-based user interface** – Even though the help of the Web technology that support the full potential of texts, they still cannot match the expressive power of graphical means.

2. **Ranked-lists** – Both use ranked-lists in order to present the recommendation and search results. The ranked-lists are just one-dimensional (rank) and easy to understand. However, this limited dimensionality lacks the applicability of the exploration techniques.

Adaptive VIBE aimed to overcome these limitations by extending a relevance-based spatial visualization algorithm called VIBE (Section 3.1). It separated the user query and the user model visually in the VIBE visualization, so that the users should be able to flexibly explore the search space and benefit from the personalization more easily. Adaptive VIBE was implemented as a Java applet and was integrated into a Web-based personalized search framework. The details of the Adaptive VIBE system is described in Section 3.2.

At the same time, the extension of the old keyword-based user models was done using named-entities (NEs), which are more semantic elements compared to the keywords. The NE-based user modeling process for the Adaptive VIBE system can be found in Section 3.2.3.

3.1 THE IDEA

Adaptive VIBE is based on the traditional VIBE (Visual Information Browsing Environment) visualization algorithm devised by Olsen et al [91]. VIBE is a reference point-based spatial visualization. That is, it shows the objects according to their similarities to special reference points, called POIs (Point of Interest). Usually, POIs represent specific concepts that are meaningful in the search space.

Adaptive VIBE is an extension of the original VIBE and added the personalization feature to it. Therefore, it is essential to understand the original VIBE algorithm first and then we can explain how Adaptive VIBE was constructed. At the end of this section, we try to compare TaskSieve and Adaptive VIBE, so that we can compare the differences between them and to contrast advantages of the new Adaptive VIBE approach.

3.1.1 The Original VIBE

VIBE (Visual Information Browsing Environment) [91] is a reference point-based information visualization technique. The reference points are called POI (Point of Interest) and the objects to be visualized are placed on the screen according to the similarity ratios to the POI. The main idea is that if a document is more similar to POI P_a than POI P_b, then it is placed closer to P_a than P_b, and the closeness is determined by the document to POI similarity ratio.

For example, if a document has similarities of 0.3 and 0.6 to POI P_a and P_b respectively, the similarity ratio to these two POIs is 1:2 and the document is placed one-third of the way from P_b on the line connecting those two POIs, because it is twice as similar to P_b than P_a. The detailed algorithm for placing a document among multiple POIs was presented in [91].

Users can drag and move POIs anywhere they want and the document locations are updated depending on their similarity ratios to the POIs. That is, they can see that the similar documents are following the movement of the POIs. The more similar the documents are, the further they follow the POIs. Therefore, they can easily discover which documents are more similar to a certain POI by their locations and can also learn the degree of similarity by the documents' degree of movements.

Another advantage of VIBE (including other visualization approaches) is that it can represent much more documents than the ranked-lists. In contrast to the ranked lists that usually show only 10 to 20 documents per page and to the fact that users mostly navigate within first 2 or 3 pages, VIBE has the capability to show hundreds of documents at the same time on a single screen, even without scrolling. Figure 16 is a screenshot of the original VIBE implementation.

3.1.2 Our Approach – Adding Adaptivity

The idea to make the basic VIBE framework adaptive is simple – *by adding the user model POIs in addition to the query POIs and then visually discriminate those two groups using special layouts*. The queries and user models are core components generally found in personalized information retrieval systems. Usually, text-based information retrieval systems let

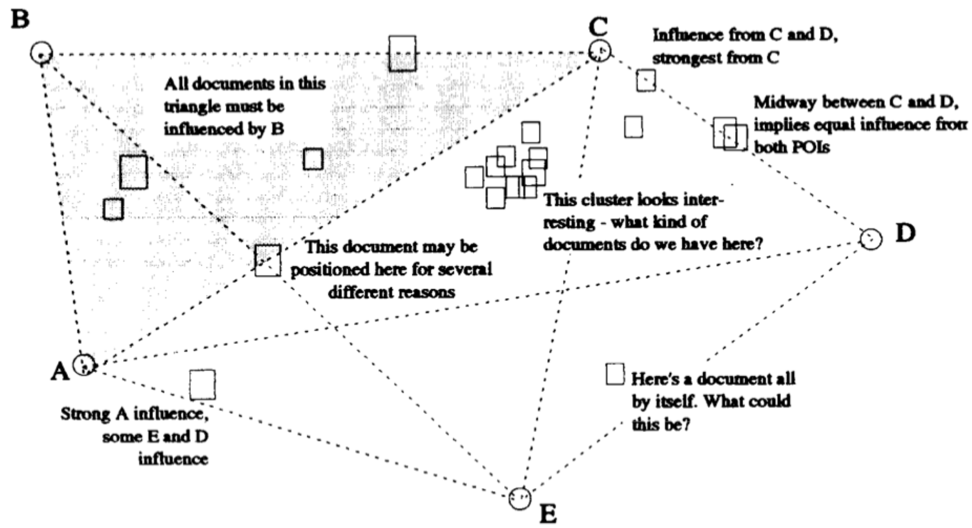

Figure 16: The original VIBE implementation in [91]

users type in their query terms and display the retrieved lists in a ranked list. The query terms are highlighted in the summaries accompanied with the document titles or hyperlinks. The main concern of the preliminary studies (YourNews and TaskSieve) were how to represent the user model and the queries and what is the best method to control them.

Adaptive VIBE provides a method that represents the user model and the user queries in a single visualization plane and separates them spatially. It defines the user model and the query as POIs, and the retrieved documents as the objects to be allocated according to the similarities to the POIs. It adds the adaptivity to the traditional VIBE visualization by restructuring it. By separating the user queries and the user model spatially, it can separate the target documents spatially too. The documents more similar to the user model will be located closer to the user model POIs and the ones more similar to the queries will be located to the query POIs.

Figure 17 is an example of Adaptive VIBE visualization. In this example, a query “NUCLEAR WEAPON” was provided by a user and by tracking his interest, a user model containing the terms “KOREA”, “JAPAN”, “NORTH”, “TORNADO”, and “SHELL” was constructed. Two components – the query and the user model – were displayed as POIs

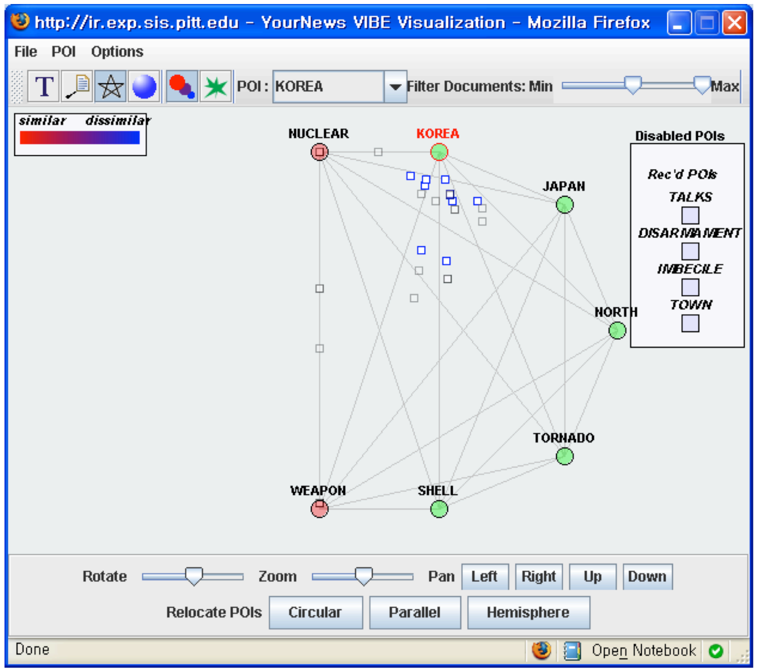

Figure 17: Adaptive VIBE visualization

(discs) and the retrieved documents by the personalized search engine were displayed as small squares. The query and user model POIs were separated in the space and were painted in different colors. The query POIs were displayed in red colors on the left hand side and the user model POIs were displayed in green colors on the right hand side. This separation was provided to make the visualization adaptive.

The user query and the user model terms played the key role to retrieve the documents from the entire corpus. The distribution of the documents within them is easily visible by examining their spatial proximity to those POIs separated into two groups. There are some documents probably describing general "nuclear weapon" issues but we can also notice that some articles may be focusing on a very specific event about "North Korean nuclear weapons" by examining the locations of the documents around the "KOREA" POI.

This example clearly shows the advantage of the Adaptive VIBE framework. It can add context to the traditional information retrieval visualization. Non-adaptive information retrieval systems or information visualization systems cannot distinguish the current context

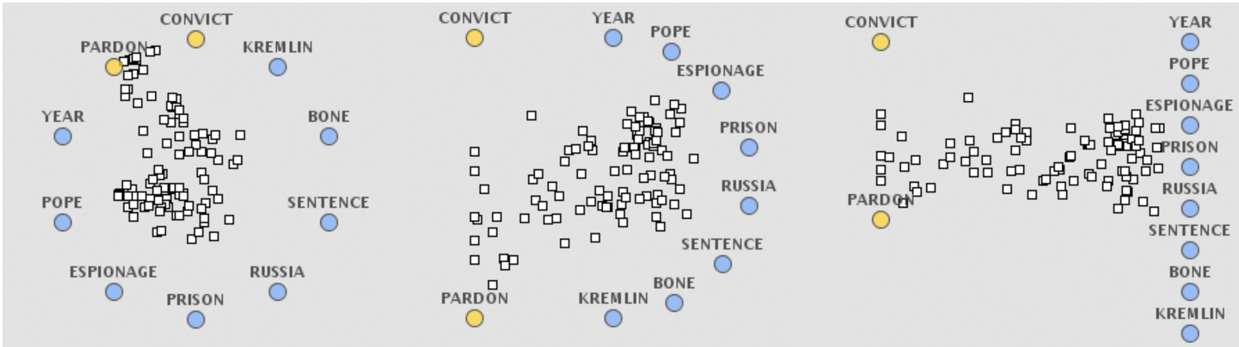

Figure 18: Classic (a) and Adaptive ((b) and (c)) VIBE visualization

of the search task – North Korean nuclear weapon development program – from the general topic. Without the context about the user interest, the system might have mixed other nuclear weapons issues (e.g. the Iranian nuclear weapon) and the current user's interest. With the information in the user model, it could filter out the irrelevant documents and display more clearly the relevant documents that are closer to the user interests.

In Adaptive VIBE, this advantage was achieved by incorporating user models as POIs and separating it into a different group of POIs. Traditional VIBE systems usually placed the POIs equivalently in a circle. However, by spatially separating two POIs groups as in the previous example, we could also expect the separation of documents according to their contexts. Therefore, we added two more layouts to the circular layout of POIs, named as Hemisphere and Parallel. They are compared in Figure 18.

The layout used in Figure 17 is called Hemisphere (Figure 18 (b)). It splits the basic circular layout into two hemispheres and places two POI groups on those two hemispheres separately. The Parallel layout maximizes the separation of two groups by placing the POIs in two parallel columns as in Figure 18 (c).

Unlike the classic VIBE (Figure 18 (a)), which does not separate the spatial positions of these two groups of POIs, the Adaptive VIBE layouts separate them horizontally (left and right, or yellow and blue in the figure above). Eventually, the document space is separated according to whether they are more similar to queries or user models.

Compared to the previous approach used in the TaskSieve study, this method can provide much more information and user control. Users can easily understand the relatedness of large number of documents to the queries or to the user models visually and interactively, so that they can efficiently find out relevant documents. At the same time, they can save more time and effort than using ranked-lists, because they do not have to explore deeply down to lower rank pages.

3.2 THE IMPLEMENTATION

3.2.1 Overview

Figure 19 shows a screenshot of the implementation of the Adaptive VIBE system. The Adaptive VIBE visualization itself was implemented as a Java applet and occupies the largest space within the system, so that it can display as many documents as possible and allow users to explore and analyze them.

The user begins the initial search by entering freely their search terms according to their tasks to be accomplished. Initially, it is identical to the conventional non-personalized searching, but after they start collecting relevant passages from the documents snippets or from the fulltexts, they can save them to a special area called "Task Model Notes", which is an implementation of the shoe-box in the sense-making processes by [94]. The task model is instantly built by analyzing the contents of the Task Model Notes and then displayed to the users in the visualization as well as in the term-cloud format.

As can be assumed from the word "Task", the searching and reranking by the personalization part identical with TaskSieve. In fact, Adaptive VIBE was integrated into the TaskSieve framework replacing the ranked-lists with the Adaptive VIBE visualization. The difference in terms of *implementation* of Adaptive VIBE from TaskSieve is as follows.

1. The HTML-based Ranked-lists were replaced by Adaptive VIBE Java applet. It means the list of documents are presented as squares in the visualization. The document summaries included in the lists are displayed using a different mechanism within the visual-

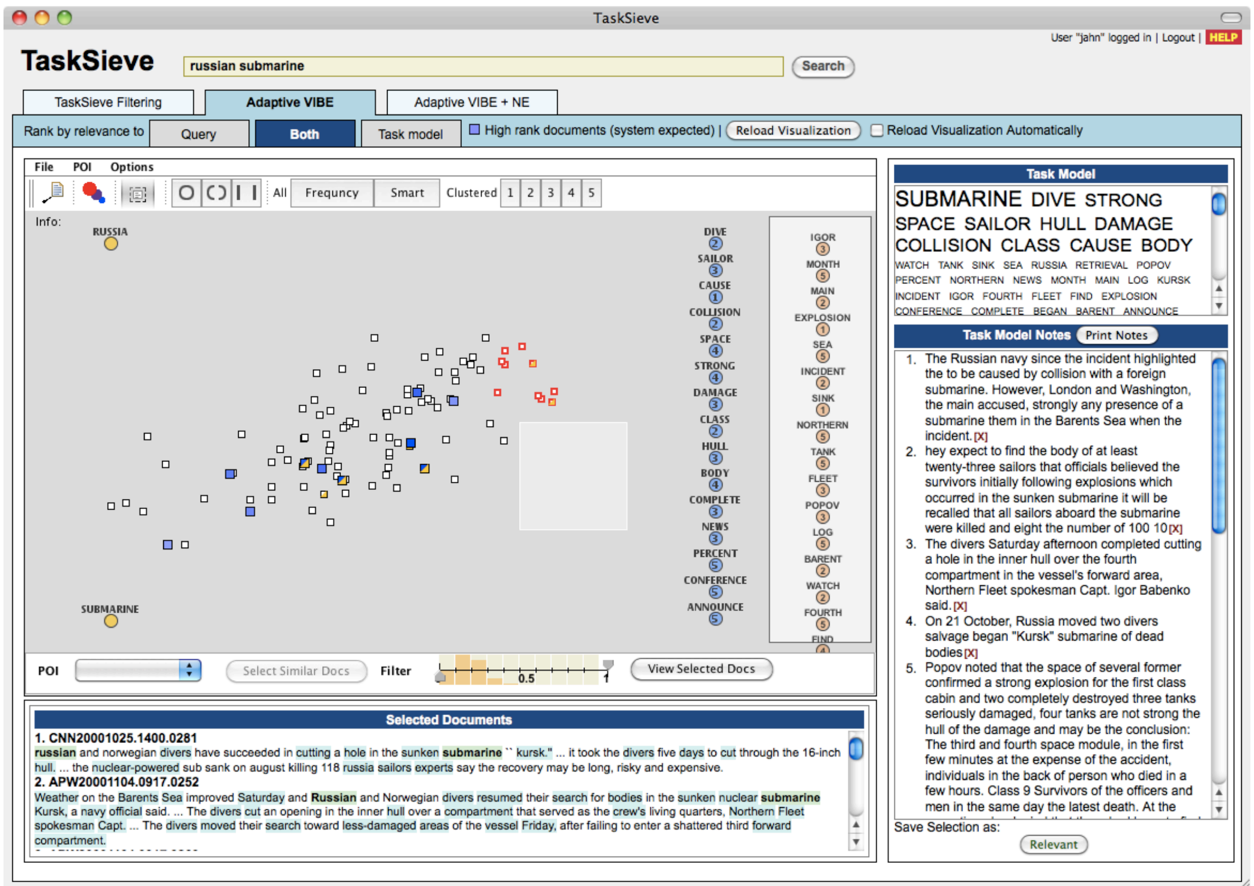

Figure 19: Adaptive VIBE prototype implementation

ization.

2. Users can open the full-text from the visualization.
3. A list of documents can still be displayed in a separate panel, but they are activated from the visualization.
4. User model was extended to use named-entities as more conceptual elements than the conventional keywords. This extension is explained in Section 3.2.3.

The user can manipulate the visualization, interact with it, and then select one or multiple documents that may contain relevant information they are seeking for. The document list is shown in the box below again so that the user could do the next iteration of searching by annotating/saving newly found relevant passages. The existence of this textual document list is important because users do not want want to use graphical representations alone for navigation [70, 72].

When the contents of the user model is updated by accumulating the relevant passages to the Task Model Notes box (shoebox), the user model can be updated with a new set of user model POIs. This can be done automatically whenever the user model contents change or manually by clicking on a button. As in TaskSieve, the user can decide the importance of the user model to her/his own query by clicking on a tab among "Relevant to Query, Both, or Task Model."

Figure 20 illustrates the process described so far. Basically, user query and user model (except the first round where the user model is empty) works together to retrieve candidate documents. Adaptive VIBE visualizes the documents with the query and the user model information on the same space (**but separated spatially**) and let the user find out relevant documents. Users are expected to sort out relevant documents more easily and efficiently, and then extract the required information from the documents in order to store them into the shoebox. The change in the shoebox is reflected to the user model on the fly, which returns a new set of documents to be analyzed for the next iteration.

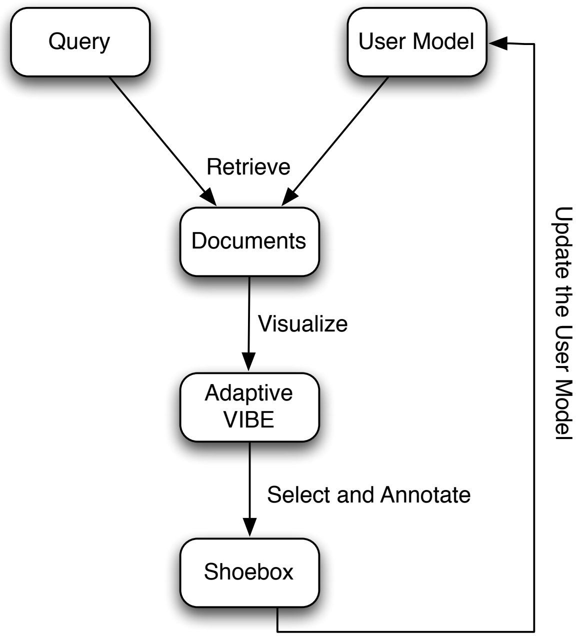

Figure 20: Adaptive VIBE – system flow

3.2.2 User Modeling in Adaptive VIBE

The process how the user model is created in the Adaptive VIBE framework was described in the previous section. User annotates text fragments from retrieved documents into the notebook. From the notebook, important keywords are selected and the user models are updated on the fly.

Adaptive VIBE was integrated into TaskSieve and it shares the user model keyword selection process. The keyword selection from the notebook utilizes a variation of the classic term frequency (TF). We assumed that the keywords appearing frequently in the notebook were more important than the less frequent keywords. Therefore, we defined a *note frequency (NF)* as follows.

$$NF = \sum_{i}^{|note|} TF_{note_i} \tag{3.1}$$

All keywords included in the notebook were sorted by their NF scores and the top

$N = 300$ keywords were selected to be used in the user models. The selected keywords built the user models. The user model was used in two ways: (1) to filter and rerank the initial search results and (2) to be displayed to the users (open user model).

As can be seen in Figure 19 (upper right hand side box), the open user model shows the top N highly frequent keywords in the notebook. The most frequent keywords are painted in bigger and bolder fonts. Unlike YourNews (Section 2.5.1), users cannot disable or manually add keywords.

Along with this cloud-format open user model, Adaptive VIBE displays the user model contents as POIs in the VIBE visualization, which is the core idea of Adaptive VIBE. Users can distinguish the user model POIs from the query POIs by their locations in the Adaptive VIBE layouts (Hemisphere and Parallel) or by different color coding (blue as user model POIs and yellow as query POIs).

Users can organize the user model POIs using the feature called "POI Dock," where they can temporarily disable the POIs. Please see Section: 3.2.4.2 for the details of the POI Dock feature.

3.2.3 Named-entity Based User Modeling

The second proposal of this study is to extend the keyword-based user models. As discussed earlier, keyword-based user modeling lacks semantics. In order to overcome this limitation, we decided to adopt named-entities as conceptual user model components. The expected benefits of the named-entities can be summarized as follows.

1. **Overcome the term dependency problem** – pure keywords in the user models are considered independent from each other and it is evidently an unnatural assumption. Some named-entities are represented as phrase forms where dependent words are bundled together.
2. **Overcome the stemming effect** – Porter or Krovetz stemming algorithms are powerful when they are used solely for systems, as in the indices of information retrieval systems. However, if they are revealed to users in the interactive information access systems, subtle meanings can be lost due to the effect of stemming. For example, users may need to

distinguish "Russia" and "Russian" for certain tasks, but the stemmers normalize them equally to "Russia." Named-entities can clearly discriminate the former as a name of a country and the latter as people.

3. **Recover number-based concepts** – usually, information retrieval systems drops numbers in the texts during the indexing stage because they are considered not to contain important meanings. Even though the numbers are included in the indices, it is very hard to understand if they mean count, day, or time. Named-entities can detect the numbers conceptually and solve this limitations.
4. **Understand the cross-references among concepts** – the named-entity extractor used in this study can generate the cross-references among entities, which are rich with semantic information by themselves. One example of this cross-reference information is "Barack Obama", who can be referred differently. He can be called as "Barack Obama," "Obama," "Mr. Obama," "He," "The president," "Mr. President," etc. The entity extractor we used can capture this difference and give us a unique identifier for this person, or an entity.
5. **Categorize by the entities types** – the extractor could annotate each named-entity as `PERSON`, `LOCATION`, `ORGANIZATION`, `DATE`, `TIME`, etc.

3.2.3.1 Named-entity Extraction A named-entity extractor developed by our project partner at IBM was used for this project [41]. It is based on a statistical maximum-entropy model, which recognizes 32 types of named, nominal and pronominal entities (such as `PERSON`, `ORGANIZATION`, `FACILITY`, `LOCATION`, `OCCUPATION`, etc), and 13 types of events (such as `EVENT_VIOLENCE`, `EVENT_COMMUNICATION`, etc). Figure 21 shows an example of the named-entities extracted from a document.

The tagger uses a large inventory of features to perform the entity detection, including part-of-speech tag, text chunk, WordNet, and syntactic structure information. Additionally, it also performs co-reference resolution for the identified entities, linking pronominal and nominal instances with their named antecedents (if they have one) and identifies and classifies relations between the discovered entities. This co-reference information could be used for name disambiguation. For example, the name "`ski_lovers`" and "`who`" (line 7

```
<DOCNO>ZBN20001113.0400.0019</DOCNO>
<ENT>
LOCATION NAM 107 114 107 114 FALSE ZBN20001113.0400.0019-E0 Mention-0 Salzburg Salzburg
ORGANIZATION NAM 126 129 126 129 FALSE ZBN20001113.0400.0019-E1 Mention-2 Burg Burg
COUNTRY NAM 3208 3221 3208 3221 FALSE XDC:Cntry:United_Kingdom Mention-156 United_Kingdom United_Kingdom
COUNTRY NAM 3227 3233 3227 3233 FALSE XDC:Cntry:Germany Mention-157 Germany Germany
PEOPLE NOM 1466 1475 1466 1475 FALSE ZBN20001113.0400.0019-E75 Mention-74 ski_lovers ski_lovers
PEOPLE PRO 923 925 923 925 FALSE ZBN20001113.0400.0019-E75 Mention-51 who who
PERSON NAM 3278 3283 3278 3283 FALSE XDC:Per:_deli_kaprun_ Mention-161 Kaprun Kaprun
OCCUPATION NOM 597 605 597 605 FALSE XDC:Per:_deli_kaprun_ Mention-37 spokesman spokesman
PEOPLE NOM 1741 1749 1741 1749 FALSE ZBN20001113.0400.0019-E78 Mention-90 residents residents
PEOPLE PRO 1778 1781 1778 1781 FALSE ZBN20001113.0400.0019-E78 Mention-91 they they
PERSON NAM 1914 1919 1914 1919 FALSE XDC:Per:judith_miller Mention-96 Miller Miller
PERSON NAM 2313 2322 2313 2322 FALSE XDC:Per:judith_miller Mention-113 Miller_456 Miller_456
PERSON PRO 1983 1984 1983 1984 FALSE XDC:Per:judith_miller Mention-98 he he
```

Figure 21: Named-entity annotation

and 8) in Figure 21 were annotated as a same entity. "Who" is a relative pronoun that refers to the "`ski lovers`." This information was annotated using the same entity ID (`ZBN20001113.0400.0019-E75` in the 8th column) in this single document.

It also supported the entity reference across multiple documents, which was annotated as "`XDC`" (meaning cross-document) in the same column. Therefore, `XDC:Cntry:United_Kingdom` (line 5) could be treated as a unique entity regardless of its form ("United Kingdom," "UK," or "She") across documents. Another example is the person name "judith_miller" which appeared three times in Figure 21 (line 13 to 15) with three different forms: "`Miller`", "`Miller_456`", and "`he`" (last two columns). We could disambiguate that these three instances refer to an identical person and treated them as a unique entity. Finally, document number, entity ID, and their frequencies were stored in the database after the disambiguation process, so that we could use them for constructing the concept-based user models.

3.2.3.2 User Model Construction We applied the same mechanism used for constructing the keyword-based user models. They relied on the user annotations and tried to extract most meaningful keywords from the user notes. Just like this approach, we could extract the meaningful named-entities from the notes.

As shown in Figure 21, every document used in the experiment was annotated with the named-entity extractor and the extracted entities were stored into the database. By

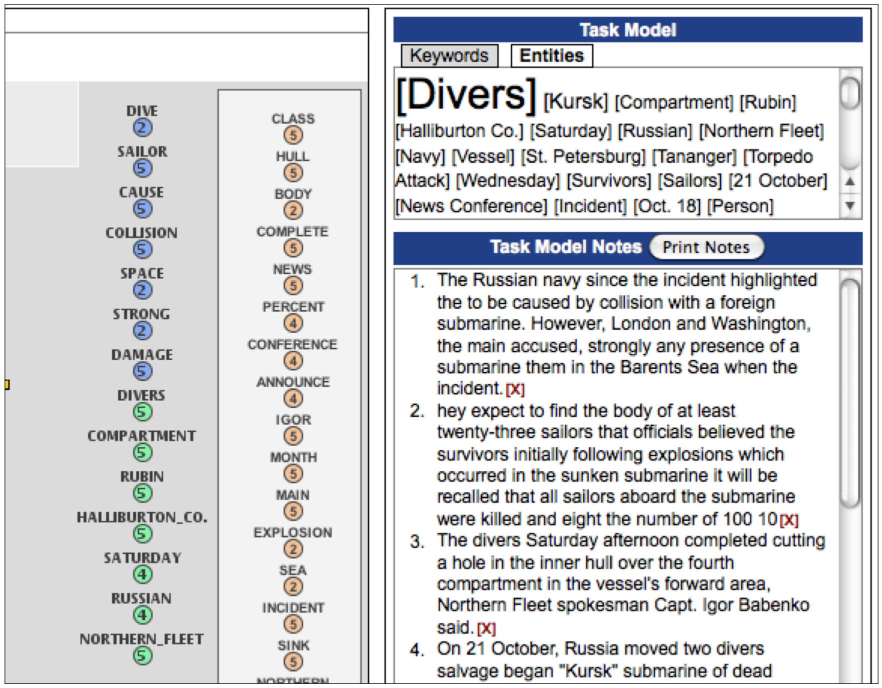

Figure 22: Named-entity based user model and visualization

observing the user action that stores a specific note from the documents, the system can search the database and which entities are included in the note. After analyzing the entity weights using the *NF (Note Frequency)* scores (Equation 3.1), the system could list meaningful named-entities just like in the keyword-based user model.

We placed the named-entity based user model in the textual format at the same location with the keyword-based user model and let the users switch between two models using a simple tab-based interface. They could compare and examine the contents of those two user models. At the same time, the VIBE visualization showed the named-entity user model as POIs too. They were displayed next to the keyword-based user model POIs, but painted in different color, so that users could distinguish them quickly in the visualization (Figure 22).

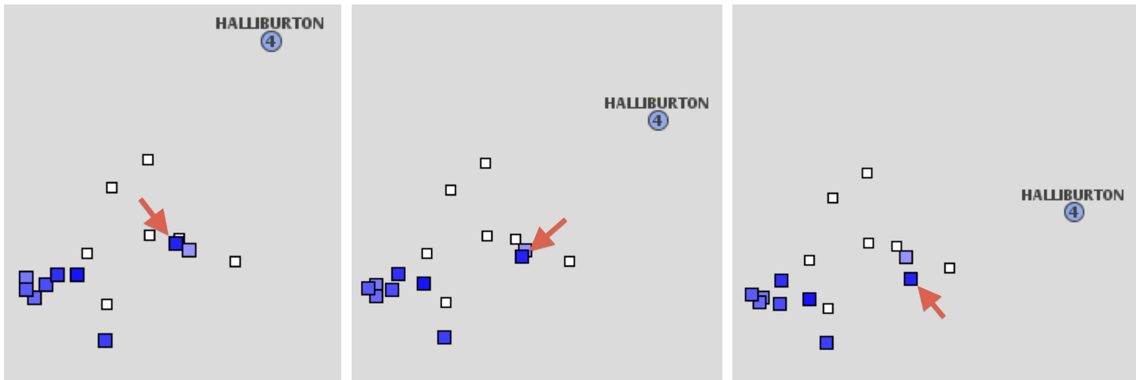

Figure 23: Dynamic POI movement. According to the movement of POI "HALLIBURTON," some documents show smooth and dynamic movements (for example, the arrow marked square), whereas another ones remain static.

3.2.4 Interactive Features

Along with the core functionality of Adaptive VIBE, we added more features that can empower the adaptive visualization in the Adaptive VIBE environment.

3.2.4.1 Dynamic Movement of Documents Following POIs Even though the original VIBE supported moving POIs by mouse dragging, it could not show the in-between steps of the movements. It just could show them jump from the starting point to the finish point of a movement. Consequently, the document locations were not updated dynamically[1].

In our Adaptive VIBE implementation, the POIs could be dragged naturally, so that the users should be able to observe the dynamic movements of POIs (and the following documents). They did not just see the POIs and the documents jumping from one position to another. They could observe the in-between animating movements of the objects and could more clearly understand the similarity relationships among the objects. Figure 23 illustrates the dynamic movements of POIs supported by our VIBE implementation.

[1]VIBE implementation Version 3.5 (built in September 1997)

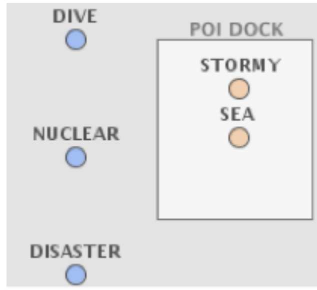

Figure 24: Adaptive VIBE features – POI Dock

3.2.4.2 POI Dock for Visual User Model Manipulation Sometimes, POIs need to be disabled for various reasons. Therefore, users can move the POIs to a special area called "POI Dock," so that the POIs locked in the dock are disabled and make no effect on the visualization. It can be re-enabled by dragging out from the dock by the users. This POI-docking feature has several advantages.

First, users can have control over the user model POIs recommended automatically by the system. That is, it is almost equivalent to editing the contents of the user models. Even though the system tries to estimate best user model elements to represent user interest, it cam sometimes make errors which may lead to the inaccuracy of the personalized output. This feature can allow users to fix the errors themselves.

Second advantage comes from the characteristics of user models that can be defined separately according to the topics or the time range they cover [44]. However, the current design of TaskSieve assumes one single task per each session and supports just one user model. Even though one big task can be sub-divided into several sub-tasks, it is an overkill to define multiple user models for each of them. At the same time, because the sub-tasks are correlated to each other under the hood of the parent task, it is not an appropriate approach to simply divide them. Therefore, instead of defining multiple user models, users can use the POI dock feature so that they can disable some POIs that are not relevant to the current sub-task but were relevant to another sub-tasks.

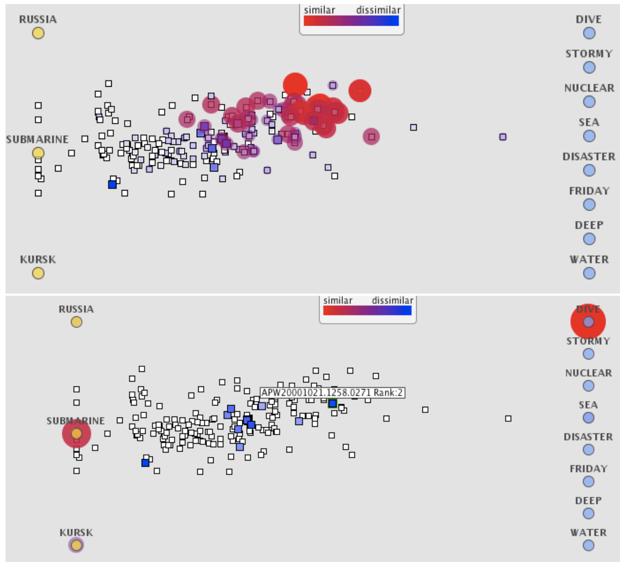

Figure 25: Adaptive VIBE features – Similarity Overlay Disc. Users can find relevant documents from a specific POI (above), or can find similar POIs from a specific document (below)

3.2.4.3 Similarity Overlay Disc The basic VIBE algorithm places documents closer to the related POIs and can help users understand the POI-document relatedness. When it is unclear, they can move a POI and observe the documents following the POI. The more the documents follow, the more they are related to the POI. However, users may want to get that information while they keep the positions of the POIs. At the same time, they may need to know the inverse relationship. That is, to find out more related POIs from a specific document. Figure 25 shows a feature supporting this need. In the example above, the user moves the mouse cursor over the POI "DIVE" and the red-to-blue spectrum colored discs are overlaid on top of the related documents. The size of the disks is proportional to the relatedness too. The example below shows the reverse usage – finding POIs from a document.

3.2.4.4 Document Filter Because the goal of Adaptive VIBE is to overcome the limitation of the ranked list that can show efficiently just 10-20 documents on the screen, it

Figure 26: Adaptive VIBE features – Document Filter

displays hundreds of documents at the same time on the screen. Even though it increases the probability to recover useful information, it can sometimes produce a severe clutter on a small area of the visualization too. Therefore, we provided a simple mechanism to hide documents by adjusting the lower and the upper thresholds of POI to document similarities. A double slider widget was used for this purpose (Figure 26). It lets users decide the similarity range of documents to be displayed for a specific POI. For example, users can set the range as 0.5∼1.0 for a POI "DISASTER" in order to display only the documents that are very similar to the keyword or concept "DISASTER."

3.2.4.5 Marquee Selection In our pilot study (will be discussed in Section 4.1), we found that the relevant documents could be found nearer to the user profiles in the visualization. In order to let users easily access the documents using this idea, we added a spatial marquee tool. In Figure 27 (above), the current user was interested in the documents that might have information about the "KUSRK" and/or "DISASTER" and decided to examine the contents of the documents. S/he therefore defined a rectangular region (displayed dim) and then the documents in that rectangle were automatically selected (squares in red borders). The selected document list was provided to the user with task infused snippets (Figure 27 below) . They could also open up a full-text view window by clicking on the document titles.

3.2.4.6 Visual Relevancy on Document Icons Even though our approach is purely two-dimensional visualization based, we can still make use of the ranks (or the relevance scores) of the documents, which were generated using the query and the user models. This information was added to the Adaptive VIBE visualization in order to provide users with

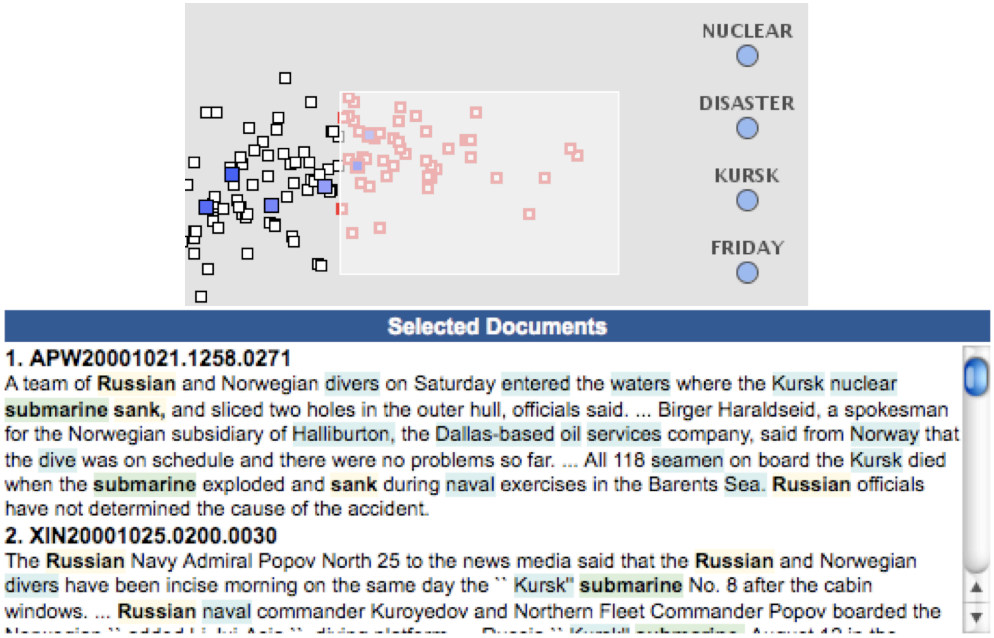

Figure 27: Adaptive VIBE features – Document Selector

additional information about the top N documents estimated by the personalization engine. The blue squares shown in Figure 19 are top 10 documents in the retrieved set, which have the top 10 relevance scores calculated by the system.

3.3 SUMMARY

In this chapter, we showed the concept and the implementation of our adaptive visualization approach, Adaptive VIBE. Adaptive VIBE was designed to present the personalized search results efficiently, by separating the spaces between user queries and their user models. Therefore, the users can see the adaptive space visibly and they can explore the space and locate relevant information interactively.

We also introduced the concept-based user model for Adaptive VIBE. It was constructed using the named-entities, which are much more semantic elements compared to the simple keywords. We expect this new user model can better represent user interests and contribute to improving Adaptive VIBE.

4.0 PILOT STUDIES

This chapter introduces two pilot studies that were conducted in order to test the feasibility of the Adaptive VIBE system and named-entity based user modeling. Adaptive VIBE is our personalized search system that can show the search results using an adaptive visualization algorithm. We also extended the keyword-based user model of Adaptive VIBE by adopting named-entities as semantic elements that could express more meanings than simple keywords.

Before starting the user study (Chapter 5 and 6), we were able to conduct offline experiments that did not involve human participants. Instead, we used the log data from our the previous personalized search study – the TaskSieve study. We used the data to simulate the visualizations in the Adaptive VIBE and the Adaptive VIBE plus named-entity user model framework and could see how the visualizations really looked like.

We observed whether and how the relevant and non-relevant documents formed different clusters in those adaptive visualizations. If the relevant documents are more clearly separated from the non-relevant documents with the discrimination power of the user models, the visualizations were regarded as better than those with less discriminations. This assumption was made in the same context with [38], where the discrimination power of keywords in the VIBE visualizations was explored. Even though TaskSieve was not a visualization system, its log data was still valid enough for the simulation because it included the user feedback and user model information.

4.1 ADAPTIVE VIBE EXPERIMENT

This study was conducted in order to test the potential of Adaptive VIBE (Section 3.1). Using the log data extracted from the TaskSieve user study, we could re-generate the visualizations from the log data (containing user queries, user model snapshots, and the returned documents from the system).

Because we also had the groundtruth information for each task, we were able to observe if the visualization created from this experiment could provide proper information that can guide the users to relevant documents. The following sections explain the pilot experiment procedures and discusses its findings.

4.1.1 Procedures

In order to test our adaptive visualization idea, we were able to experiment with the real user data extracted from the log file of the TaskSieve study. A dataset was constructed using the log file that contained the following attributes.

- User ID
- Topic ID of the target task
- Query terms
- User model (Top N weighted terms)
- Documents retrieved using the query and the task model (top 100)
- Relevance information about the retrieved documents

That is, we could get every query issued by the real subjects and the retrieved results from the TaskSieve study and the corresponding important information such as user model contents at each time frame. We were also able to have the relevance information about the retrieved documents thanks to the ground-truth information we used in the TaskSieve experiment. We could re-constructed every snapshot of all users, their queries, and the retrieval results.

Therefore, with this dataset, we could just replace the ranked lists actually shown to the subjects in the TaskSieve study with the Adaptive VIBE visualization. Not only we

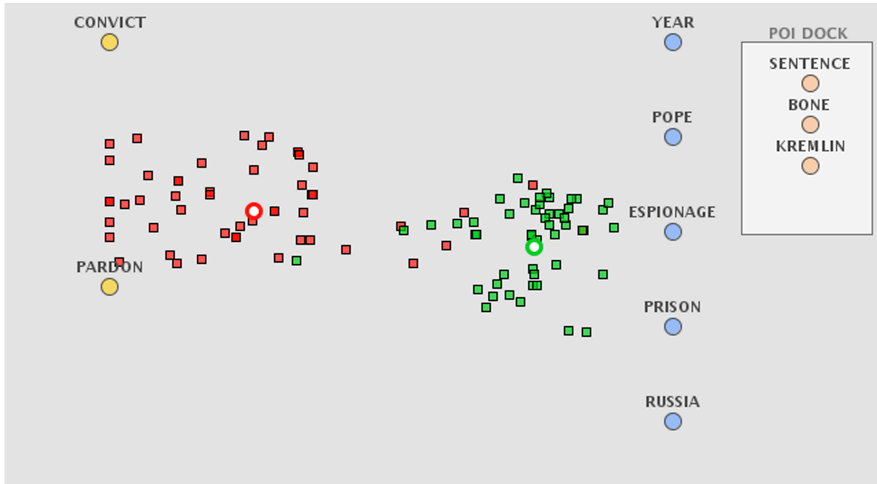

Figure 28: Adaptive VIBE experiment using log data of TaskSieve

converted the textual ranked list to two-dimensional visual document representation, but we also examined the relationship between the spatial representation and the relevancy of each document.

Figure 28 shows one example of the visualizations we created by this procedure. The query terms appears in the left column, as yellow circle POIs, "CONVICT PARDON." The user model contents when this query was issued were "YEAR", "POPE", "ESPIONAGE", "PRISON", and "RUSSIA." The retrieved documents are displayed in squares. The difference between the ranked lists and this visualization is: (1) the document arrangement is now in two-dimensional VIBE visualization and (2) the relevant and non-relevant documents are marked by green and red colors respectively. Of course, the relevance information was not visible to the subjects in the TaskSieve.

As seen in the figure, it was very interesting to find that the relevant (green) and non-relevant (red) documents created clear clusters. We did not try any conventional clustering methods for this separation. It was made naturally by the query and the user model POIs of Adaptive VIBE. What was also interesting was that the relevant document clusters (green) was spatially closer to the user model side than the query side.

Table 4: Horizontal positions of the cluster centroids (in screen coordinate)

Layout	Radial	Parallel	Hemisphere
Relevant	304.3	300.9	337.7
Non-relevant	283.9	207.96	295.3
Difference	20.4	92.94	42.40
(Relevant–Non-relevant)	($p < 0.01$)	($p < 0.01$)	($p < 0.01$)

4.1.2 Evaluations and Findings

Figure 28 is of course one of the good examples out of 105 visualizations we created. However, when we examined the whole 105 records, we found statistical evidences that the observation above is consistent across the entire dataset.

In order to check if the user model effect that attracted the relevant documents and the separation of the relevant and non-relevant document clusters could be generalized, we compared the location of the centroid of relevant/non-relevant document clusters and the separation of the two clusters respectively. Table 4 to Table 6 show the results. In Table 4, we can confirm that the average distance between two clusters is significantly large, especially with the Parallel layout.

In Table 5, we compared the variance between clusters in order to examine the separation between clusters. We compared this score among three layouts and found the Parallel and Hemisphere layouts that visually/spatially separate the query and the user model showed bigger separation between the relevant/non-relevant documents compared to the default Radial layout that did not perform any visual separation between the query and the user model.

We also compared the within cluster variance (Table 6). This time, the Parallel and the Hemisphere showed bigger variance again, which suggested the bigger spread of the clusters. However, this variance was much smaller than the between cluster variance and it proved not to hurt the separation of the clusters.

Table 5: Cluster separation (between cluster variances)

Layout	Radial	Parallel	Hemisphere
VAR_{BC}	109,325.05	220,362.31	186,679.57
Difference	.	111,037.26	77,354.52
(from Radial)		($p < 0.01$)	($p < 0.01$)

Table 6: Cluster dispersion (within cluster variances)

Layout	Radial	Parallel	Hemisphere
VAR_{WC}	12,124.21	16,788.85	15,617.01
Difference	.	4,664.64	3,492.79
(from Radial)		($p < 0.01$)	($p < 0.01$)

4.2 NAMED-ENTITY BASED ADAPTIVE VIBE EXPERIMENT

In the previous pilot study (Section 4.1), we used the log data from a past study in order to check how the adaptive visualization idea would look like. The same idea could be applied to the next question – if named-entity (NE) based user models for adaptive visualization are effective. We were able to use the same technique and the dataset for this second pilot experiment.

The difference here was that NEs were added to the previously keyword only user models, as implemented in Section 3.2.3. Figure 29 shows an example of this experiment. Unlike the previous experiment (Section 4.1), there existed NEs in the adaptive VIBE user model such as "`russia`", "`russian`", and "`united_states_of_america`." If they were not NEs and conventional keyword-based techniques were applied, they would have represented as "`russia, unite, state, america`" because of the stemming and stopword-removal.

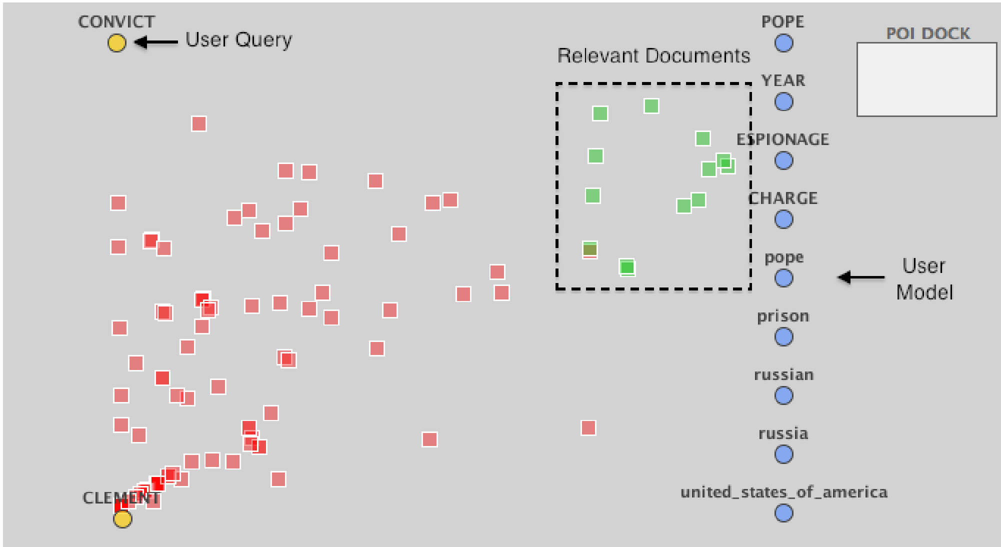

Figure 29: Adaptive VIBE based on conceptual user modeling

4.2.1 Procedures

We could examine the quality of the adaptive visualization according to the condition whether the NEs were used for the user model or not. Moreover, we were able to test which combinations of keyword+NE made better or worse visualizations.

Because the groundtruth data was still valid for this study and we could detect the relevant and non-relevant document clusters in the visualizations, we could adopt a clustering validity measure to calculate the quality of the visualizations. A measure called Davies-Bouldin Validity Index (DB-index) [36] was used (Equation 4.1). The nominator of the equation is the average distance within a cluster (cluster compactness) and the denominator is the distance between clusters (cluster distance). Therefore, it gives smaller number to better clusterings (compact and mutually distance clusters).

$$DB = \frac{1}{n}\sum_{i=1}^{n} max_{i=j}\left\{\frac{S_n(Q_i)+S_n(Q_j)}{S(Q_i,Q_j)}\right\} \tag{4.1}$$

$S(Q)$ = average distance within a cluster Q

$S(Q_1, Q_2)$ = distance between two cluster centroids

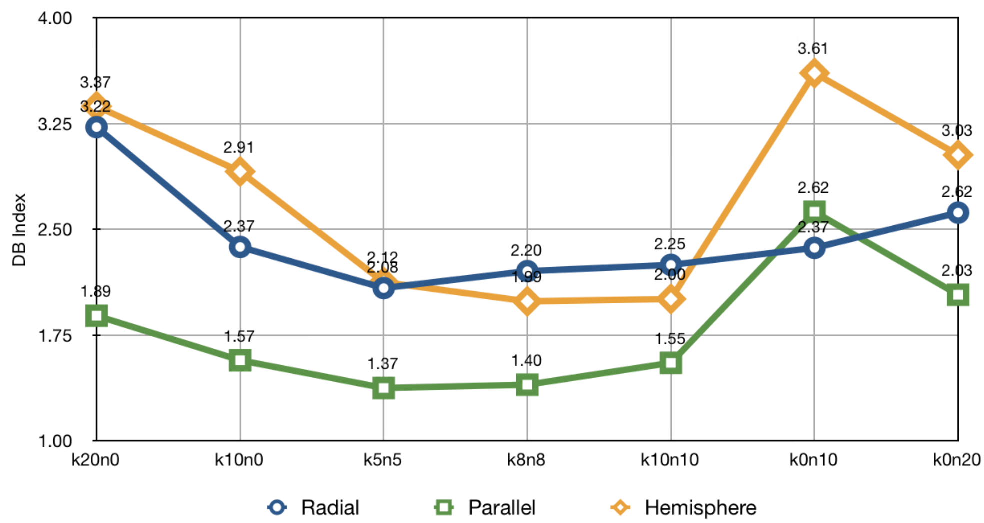

Figure 30: Comparisons of different keyword+NE mixtures and resulting adaptive visualizations

4.2.2 Evaluations and Findings

Figure 30 shows the distribution of DB indices according to different combinations of the keyword+NE mixtures. Here, kxny means x keywords and y NEs in the user model. For example, k5n5 represents the user model was comprised of 5 keywords and 5 NEs. The three lines represent the three Adaptive VIBE layout algorithms.

Overall the Parallel layout showed the best results, which is consistent with the previous experiment. Among the different mixtures of the best layout, 5 keywords and 5 NEs (k5n5) showed the best results. Keyword only (k10n0) or NE only (k0n10) showed worse performance than this half-and-half mixture of keyword and NEs and their differences were statistically significant. Bigger number of keyword+NE mixtures (k8n8 or k10n10) did not show any improvements.

We then analyzed which term of the two in the DB-index – cluster compactness and cluster distance really contributed the performance of the mixtures. Table 7 shows that the cluster spread was not different between three mixtures that showed differences in terms of the overall DB-index scores. However, the cluster distances were different among the

Table 7: Comparing within-cluster spread and between-cluster distance

Layout	Within-cluster spread			Between-cluster distance		
	k5n5	k10n0	k0n10	k5n5	k10n0	k0n10
Radial	87.67	81.25	87.67	55.23	49.94	47.68
Parallel	154.47	152.13	161.44	**151.63**	138.05	125.31
Hemisphere	110.54	109.50	110.54	82.97	70.76	74.78

mixtures, providing the biggest distance with the best performed mixture, k5n5. This result suggests that cluster distance contributed the best overall scores. That is, 5 keywords and 5 NEs in the visual user models could well separate the relevant and non-relevant clusters and produced the best resulting visualizations.

Table 8 compares the horizontal positions of the relevant and non-relevant cluster centroids. It shows that the centroids of the relevant document clusters were always bigger than

Table 8: Comparing horizontal positions of cluster centroids (in pixels)

Keyword/NE Mixture	Clusters	Radial	Parallel	Hemisphere
k10n0	Relevant	313.58	318.56	350.35
	Non-relevant	302.99	188.19	304.74
	Distance	10.59*	130.37*	45.61*
k5n5	Relevant	315.73	332.02	361.77
	Non-relevant	301.46	192.71	308.44
	Distance	14.27*	**139.31***	53.33*
k0n10	Relevant	300.68	269.44	328.31
	Non-relevant	294.71	161.63	291.77
	Distance	5.96**	107.80*	36.54*

($^*p < 0.01$, $^{**}p = 0.038$)

those of the non-relevant documents. Because the user models were placed on the right-hand side of the visualization, this suggests that the relevant documents were attracted by the user models. This result is consistent with the previous experiment.

4.3 SUMMARY

Before starting the user study on the effectiveness of Adaptive VIBE and the named-entity (NE) based user modeling for adaptive visualization, we conducted two experiments using the log data from our previous personalized search study. From the results, we could confirm the potential of both approaches.

1. The proposed Adaptive VIBE algorithm could separate the relevant documents in the search space from the non-relevant information.
2. The relevant documents were placed closer to the user models, which can be used as a clue for the users to locate relevant information more easily.
3. The NE-based user models outperformed the keyword only user models for Adaptive VIBE, when they were mixed with keywords.

Please refer to [1, 2] for detailed information about the study procedure and result analysis.

5.0 USER STUDY DESIGN

The Adaptive VIBE system implementation in Chapter 3 was validated using the offline-experiment methodology. The results shows potential of the Adaptive VIBE visualization and the concept-based user modeling (Chapter 4). Encouraged by the successful outcomes of the two pilot studies, a user study was conducted in order for deeper understanding. This chapter presents the methodology and the procedure of the user study. A formal definition of research questions and the hypotheses are provided too.

5.1 HYPOTHESIS

There are two main research questions in this study. First is whether the adaptive visualization-based information retrieval framework could help users to achieve better results than the textual ranked-list based systems. Second is whether the incorporation of semantic features into the user model would be able to help users to achieve better results, especially in the adaptive visualization setting.

Therefore, two broad hypotheses were defined in order to answer the questions.

H1: The adaptive visualization-based information retrieval framework will produce better results than a text-based personalized information retrieval systems, which is the baseline.

H2: The named-entity based user model of the adaptive visual search system will help users produce better results than the keyword-only user models.

The first hypothesis was to test the Adaptive VIBE system by comparing it with a baseline personalized search system. The Adaptive VIBE implementation (integrated into

TaskSieve framework) introduced in Chapter 3 was the experimental system. The original TaskSieve system with ranked-lists and the keyword-based user model was the baseline.

We could compare the systems in terms of two aspects: the system performance and the user performance.

System performance is the ability of a system to provide relevant information to the users. It can be measured by observing the outputs of each system (ranked lists or visualization).

User performance is the ability of a user to locate relevant information with the help of each system.

The former is identical with usual performance measures for information retrieval or recommender systems. The latter is not to measure *users'* own ability originated from their personal differences. Rather, it tries to measure the combined performance of the users and the systems. If a system provides good results and a user can fully exploit them during the interaction with the system, the user performance is regarded high. This perspective is important in order to measure the performance of exploratory systems.

Performance The "performance" here is broadly defined as a system's ability to support the personalized search systems. We used two groups of measures to probe the performance of each system: (1) precision and (2) diversity.

Precision The precision here is identical to the usual precision, the ratio of relevant documents out of the retrieved documents.

Diversity The concept of diversity is similar to the recall, a popular measure in the information retrieval community – how many relevant items in the corpus were found. However, we did not call it recall because the specific definitions were different from the formal recall. Each measures are introduced in the corresponding sections of the study results chapter.

Therefore, the first hypothesis was divided into the following two sub-hypotheses.

H1-1: The Adaptive VIBE (experimental) system will show the better system performance than TaskSieve (baseline), in terms of precision and diversity.

Table 9: Experiment conditions

Condition 1	Condition 2	Condition 3
TaskSieve – baseline	TaskSieve + Adaptive VIBE	TaskSieve + Adaptive VIBE + Entity-based user model

H1-2: The Adaptive VIBE will guide the users better in terms of the user performance than TaskSieve, in terms of precision and diversity.

The proposed named-entity (NE) based user model in the second hypothesis was implemented in the same framework on top of the Adaptive VIBE system, which was the experimental system in the first hypothesis. The Adaptive VIBE + NE-based user model was defined as the second experimental system (section 3.2.3) and it was compared with Adaptive VIBE with no named-entity support.

The same group of measures were used in order to test the second hypothesis. However, the baseline and the experimental systems are different this time.

H2-1: The Adaptive VIBE system plus named-entity based user models (Adaptive VIBE+NE, experimental) will show the better system performance than the keyword-based Advise VIBE (baseline) system, in terms of precision and diversity.

H2-2: The Adaptive VIBE+NE system will guide the users better in terms of the user performance than the keyword-based Advise VIBE system, in terms of precision and diversity.

Table 9 summarizes the experimental designed using three conditions. By comparing Condition 1 and Condition 2, the first hypothesis was tested, and by comparing Condition 2 and Condition 3, the second hypothesis was tested. In the next sections, data collection, study process, and the measures for testing the hypotheses are described.

5.2 DATA COLLECTION

5.2.1 Text Corpus

TaskSieve and the two experimental systems (Adaptive VIBE and Adaptive VIBE+NE) were all designed by the task-based information exploration concept in mind. Therefore, this study could reuse the resources used in the previous studies (i.e. TaskSieve study) without changing much: the text corpus, search tasks, and the groundtruth information for measuring the performances.

As described in section 2.5.2 already, the document collection used in the experiment was an expanded TDT4 English test corpus, in which there were 28,390 English documents published between October 2000 to January 2001 [50]. The original TDT4 test collection was expanded in order to resemble the tasks performed by intelligence analysts. 18 of the original TDT4 topics were enriched into so-called GALE topics and human annotators constructed the groundtruth information per each topic. Each GALE topic contained an overarching task theme and up to 10 different but related sub-tasks. The search outcomes of these topics were a group of selected useful passages that could be used to answer the questions raised in the tasks/subtasks [5].

From this text corpus, three test topics were selected to be used in the user study. The next section describes the procedure and method for selecting the topics.

5.2.2 Topics

As described above, there were 18 topics in the expanded TDT4 collection. Even though they were generated carefully, there existed deviations among the topics in terms of difficulties. In one of our previous studies that used this corpus, we found topic differences that affected the results [3]. We wanted to select equivalent topics that are comparable with each other in this study.

In order to compare the difficulties of the topics objectively, we devised a measure by considering the distribution of groundtruth information in the corpus. In some topics, the answers are concentrated in a small number of documents whereas the answers are dispersed

Table 10: Distribution of relevant information among topics

Topic	40009	40048	40021
$Complexity_{topic}$	71.46	64.12	73.98

across a lot of documents in other topics. The former case would be relatively easier because users would not have to explore the search space a lot.

Therefore, we defined the standard deviation of relevant passage count per document as a pseudo topic complexity measure (Eq. 5.1).

$$Complexity_{topic} = SD_{relevant\_passage\_per\_document} = \sqrt{\frac{1}{|Doc_{rel}|}\sum_{i=1}^{|Doc_{rel}|}(|Passage_{rel}| - \mu))^2} \tag{5.1}$$

The complexity was calculated for each topic and three topics whose standard deviations were almost equivalent were selected as the study topics. The three topics and their scores are shown in Appendix C and Table 10 respectively.

5.2.3 Participants

5.2.3.1 Power Analysis Thirty three participants were recruited from the University of Pittsburgh and Carnegie Mellon University[1]. In order to decide the sample size, power analysis [33] for analysis of variance (ANOVA) was conducted. The analysis allows to determine the sample size required to detect an effect of a given size with a given degree of confidence. Because three conditions were defined in the study design, ANOVA was selected among the various set of statistical methods.

According to the analysis, the sample sizes required to detect medium ($f = 0.25$) or large ($f = 0.4$) effects were $N = 53$ or 22 respectively, for $k = 3$ groups (conditions), with

[1]The study was approved by the Institutional Review Board at University of Pittsburgh and consent forms were collected from all participants.

Table 11: Demographic information of the participants

Category	Statistics			
Age	Max=48	Min=19	$\mu = 26.1$	SD =6.5
Gender	Male=23	Female=10		
Education	Undergraduate=7	Master=16	PhD=10	
Confidence in Search	$\mu = 8.09$ (max=10)	SD=1.14		

the significance level of $p = 0.05$. Therefore, 30 participants were to give medium to large size effects.

5.2.3.2 Participants Acting as Information Analysts The participants were expected to play the role of intelligence analysts because the proposed personalized information visualization method required enough training and iterative exploration of the problem space beyond the casual information retrieval activities. It would be most idealistic to recruit real analysts. However, due to practical issues, those who could act as an analyst as closest as possible were recruited. In order to realize the similarity to the analysts, the following criteria were checked when recruiting the participants.

Language issue Because the participants would read and analyze relatively large amount of news stories in a short period of time, only native English speakers or those who with equivalent language abilities were recruited.

Information search ability The participants would have enough information retrieval skills in order to complete the intelligence analysis. The experience to attend an information retrieval class or the affiliation to information science schools (or related institutions) were visible criteria that helped to measure the ability.

Education The subject pool was limited to the students enrolled in the University of Pittsburgh or Carnegie Mellon University in order to guarantee the minimal intellectual level of information analysts.

Table 11 shows a summary of the participants' demographic information.

5.2.4 System Configuration

The Adaptive VIBE implementation described in Section 3.2 was used in the user study. During the study, a 17-inch LCD display was used for all participants. We fixed the screen resolution, size of the implemented system window, and the size of the VIBE visualization area across the entire search sessions in order to retain the consistency. The size of the visualizations were 910 by 412 pixels.

In the Adaptive VIBE visualization, we configured the system to display top 15 POIs enabled by default and the remaining lower rank POIs are placed in the POI dock (Section 3.2.4.2). We chose the number 15 according to the vertical size of the visualization and the result of our second pilot study (Section 4.2), where 10 to 16 mixture of keywords and NEs showed the best results. We were able to align the optimal 15 POIs vertically (Parallel mode in Adaptive VIBE) within the limit of 412 pixels, avoiding the clutter of the user model POIs (Figure 19).

5.2.5 Collected Data

Two types of data were collected: (1) log data of the interaction between the participants and the system and (2) participants' subjective feedback. The systems observed every activity of the participants while they were using the system to solve the search tasks. The data types collectible from the system logs are shown in Table 12.

The subjective feedback data was collected directly using the questionnaires and the interviews. They asked the participants to describe what characteristics and features were useful in order to accomplish their tasks.

Table 12: Collected log data attributes by systems

Category	TaskSieve (Condition 1)	Adaptive VIBE (Condition 2 & 3)
Searching	User query Ranked list (document id, score)	User query Visualized documents (document id, position)
User model	User notes User model content – keywords (with weights)	User notes User model content – keywords or NE (with weights)
User actions	Querying Add/Remove notes Page navigation Open documents	Querying Add/Remove notes Open documents Manipulate POI (move, doc, layout selection) Features (marquee, filter, etc)

5.3 PROCEDURE

According to the study design (Section 5.1), the subjects tried to solve three tasks using three different systems. The order of the systems was randomized by Latin square design in order to avoid any possible learning or fatigue effects. They were asked to read a one-page introductory statement to the experiment, and to complete a demographic questionnaire (Appendix A) focusing on their search experience and consumption of news.

50 minute training sessions were given to the participants where the experiment coordinator explained every feature supported by the systems and let the participants solve a real task as a training. The training task was picked up from the TDT4 collection as the experiment tasks. It was exactly same with the real tasks (4000, 40021, and 40048) in terms of the structure. No real task was given to the participants before starting the main sessions.

The training session was especially important for the visualization-base systems (Condition 2 and 3) because the systems were new to the participants and most of them were completely unfamiliar with Adaptive VIBE's visual exploration feature.

Even though the main session was 20 minutes and no attempt to complete the tasks after the given time was allowed, the participants were given up to 30 minutes per system during the training. We allowed more training time because they needed both to be explained about the system and to practice themselves. We also expected the role of information analysts who could proficiently operate the search systems for the task completion and tried to help them to reach the minimum (and equivalent) proficiency level for manipulating the system. We did not mean to compare the difficulty of learning the system features between the baseline and the experimental systems.

The coordinator demonstrated how the new features could be used to solve the given problems. We tried to make the training session as realistic as possible, so that the participants could formulate their own search strategies using the new systems. The coordinator observed their activities during the training and ensured that they were ready to play the role of information analysts using the systems after the training. The participants could freely ask questions about the systems and any misunderstanding was corrected.

The results of the pilot studies – the relevant information was located closely to the user model in the Adaptive VIBE visualization – were also objectively introduced to the participants. However, they were *not* directed to replicate the expected behavior according to the pilot study results. The information about the probable proximity between the relevant information and the user model was given to the participants and it was made clear that the participants could freely choose their own search strategy, in order to retain the objectivity of the training.

After each session, the participants were asked to fill out a post-task questionnaire (Appendix B), and finally took 10-minute exit interviews. Figure 31 illustrates the procedure.

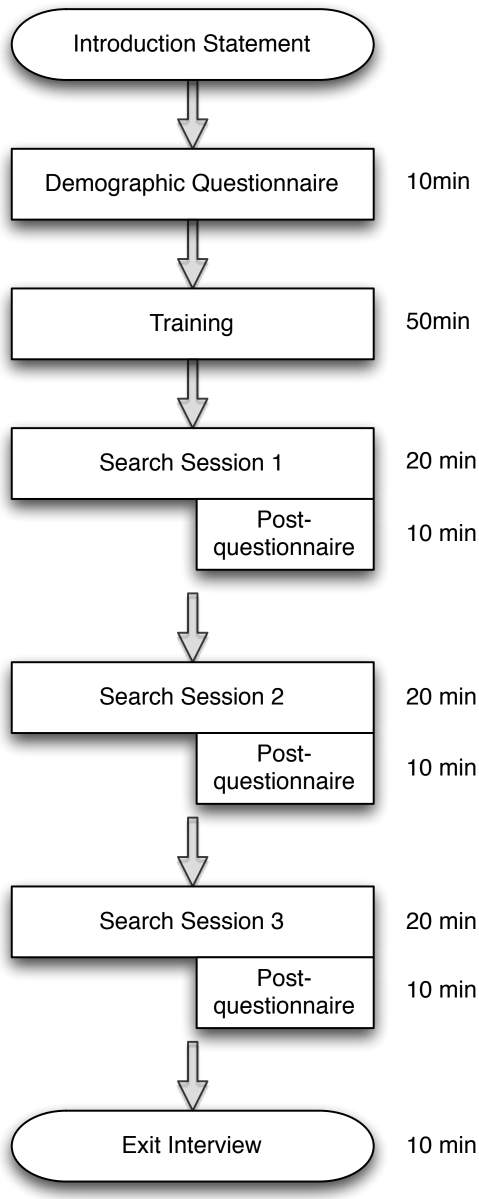

Figure 31: Study procedure and the time plan

6.0 RESULTS

This chapter presents the result of the user study that was conducted in order to evaluate Adaptive VIBE. It compares three systems: (1) TaskSieve (the baseline), (2) Adaptive VIBE[1], and (3) Adaptive VIBE with NE user modeling[2].

TaskSieve is a text-based personalized search system that has a simpler user interface for mediating user queries and user models. Adaptive VIBE extended this simple user interface using the VIBE visualization and provided a much more flexible mediation and exploration mechanism for users within the personalized search space. Adaptive VIBE with NE user modeling extended the keyword-based user models by incorporating named-entities as more semantic elements beyond the simple keywords.

The human participants were asked to solve specific tasks using the three systems. After finishing each task, they answered the questionnaires about their opinions on each search system. We first introduce these **Subjective feedbacks** of the study participants (Section 6.1).

Along with the subjective feedback analysis, the **Objective log data analysis** is presented in three broad stages:

1. **Analysis of user activities**
2. **System performance analysis**
3. **User performance analysis**

The first stage provides some descriptive statistics that can provide a basic understanding about the user activities. The second stage shows the performance analysis of the system

[1] Referred as "VIBE" in this chapter.
[2] Referred as "VIBE+NE"

outputs. That is, whether the systems returned good results to the users. It is subdivided into two categories: (1) **Performance of ranked list** and (2) **Performance of visualization**. This subdivision is due to the nature of Adaptive VIBE that operates in two stages: (1) search and rank documents by the personalization engine and (2) visualize them using the VIBE algorithm. Therefore, it is important to check the performance of both stages.

After the result is presented to the users, the users take the turn. They explore the visualization space, examine the spatial information, check the document summaries, and decide whether they should do deeper examination of the document contents.

When they think a document is probable to contain the information they are searching for, they open the document and read the full-text. Within the full-text or from the document summaries, they annotation text fragments and store them into the notebook as final reports.

These user activities can be evaluated by checking the documents they opened and the notebook contents they maintained. The second stage of analysis is called user performance analysis and it can measure the performance of user activities during the interaction with the personalized search systems.

Figure 32 summarizes the procedure of the performance analysis process. In the system performance analysis stage, we first checked the quality of the ranked lists that were the source for generating the visualizations (Section 6.3).

Then we could analyze the quality of the visualizations, which were the direct interfaces that the users interacted with (Section 6.4). For evaluating the visualizations, the same methodology used in the two pilot studies was repeated – the cluster analysis of relevant and

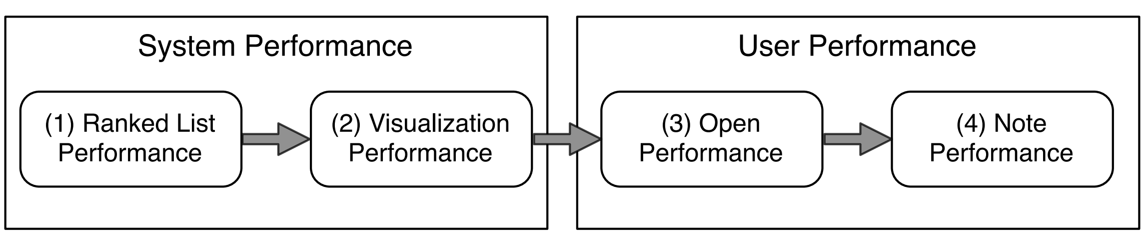

Figure 32: Performance analysis procedures

non-relevant documents. This time, the real visualizations generated during the user study were analyzed.

In the user performance analysis stage, the performances of the user activities are analyzed. Even though the system provides high quality information in the form of ranked lists or visualizations, it is of no use if the user cannot recognize and exploit them. Therefore, it is important to examine users' behaviors and outputs directly affected by the system performance. It is not solely the analysis of the user themselves, but it tries to analyze the outcomes produced during the interaction between the user and the system.

For this purpose, two core user actions were examined: (1) accessing (e.g. opening) retrieved documents (Section 6.5) and (2) annotating text fragments and saving them into the notebook (Section 6.6).

6.1 USER FEEDBACK ANALYSIS

In this section, users' subjective feedbacks collected from the post-search questionnaires are analyzed. Different questionnaires were prepared for each system because of the different feature sets supported by the systems. Except the open question asking about the user comments, all questions used five point Likert scale (1∼5), where 1, 3, and 5 were labeled as "Not at all," "Somewhat," and "Extremely" respectively.

6.1.1 Topic Familiarity and Difficulty

It was one of the most challenging problems from the study design stage to select the topics that may not be unknown to the study participants. The aim of this study is to help users to solve complex problems using appropriate tools. If a participant is too familiar with a topic, then s/he may skip the exploration process and directly locate the answers by using one or two queries.

Therefore, the topic familiarity was first checked using the questionnaires. The average topic familiarity was 1.40 out of 5.0 (SD=0.82), which was quite low considering the score

1.0 was labeled as "Not at all." Figure 33 and Table 13 compare the subjective familiarities across the three topics. Topic 40009 showed significantly higher familiarity than the other two (Kruskal-Wallis rank sum test, $p < 0.001$, pairwise comparisons in Table 14) but it still remained at the very low level (less than 2 out of 5). At the same time, this topic turned out to be rather difficult topic despite its higher familiarity (Figure 34 and Table 15). This result confirms that the study participants were not familiar with the topics at all or had just minimal knowledge and their prior knowledge did not affect the overall study results.

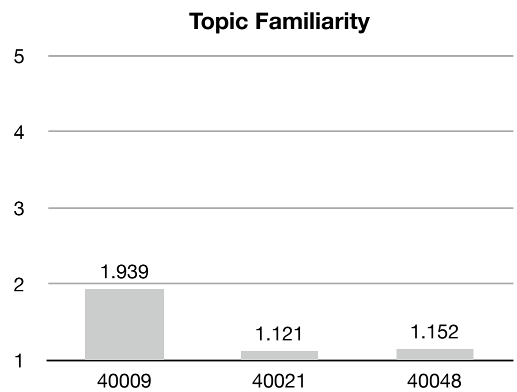

Figure 33: Subjective feedbacks – topic familiarity

Table 13: Subjective feedbacks – topic familiarity

Topic	40009	40021	40048
Mean Topic Familiarity	1.94	1.12	1.15

Table 14: Topic familiarity – pairwise comparisons by Wilcoxon rank sum test

Topic	40009	40021
40021	0.0015	-
40048	0.0015	0.9729

Along with the familiarity, the topic difficulty was another interesting issue to check. Even though a participant is not familiar with a topic, it may be easy to solve from various reasons. For example, a topic can have most of the answers in a couple of documents, then simply locating them with minimal efforts will finish the entire search task.

This issue was one of the main concerns in the study design stage and we tried to maintain equivalent difficulties of the three topics selected for the study (Section 5.2.2). In order to check if this effort was effective, we first asked the participants about their subjective perception of difficulty.

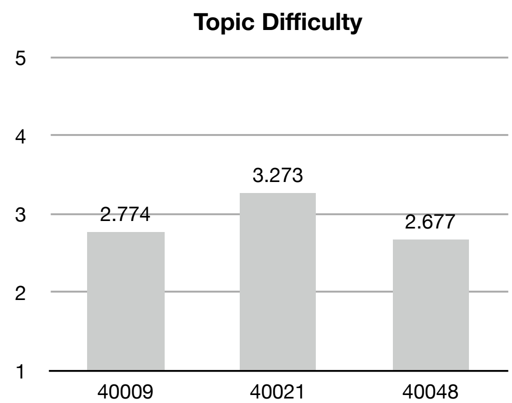

Figure 34: Subjective feedbacks – topic difficulty

Table 15: Subjective feedbacks – topic difficulty

Topic	40009	40021	40048
Mean Topic Difficulty	2.77	3.27	2.68

Table 15 and Figure 34 compare the topic difficulty scores returned by the participants. Unlike our expectation, a difference found among three topics. The most difficult topic was 40021 and the easiest one was 40048. The statistical test confirms this result (Kruskal Wallis rank sum test, $p = 0.032$).

The pairwise comparisons among the topics (Table 16) show that the difference was found between topic 40021 and 40048, which were the hardest and the easiest topics. Even though some subjects pointed out the topic 40009 was difficult too, any significant difference between 40009 and 40048 was not found.

Table 16: Topic difficulty – pairwise comparisons by Wilcoxon rank sum test

Topic	40009	40021
40021	0.144	-
40048	0.522	0.032

Table 17: Subjective feedbacks – positive reaction

System	TaskSieve	VIBE	VIBE+NE
Average Score	3.76	3.18	3.39
SD	0.75	0.98	0.93

6.1.2 Positive Reactions

The participants were asked to report their overall impressions on the performance of the three systems. Even though there was a risk of over-simplification by asking a single question on this matter, it was still a good measure together with other more specific questions in order to catch the big picture.

Table 17 compares the reactions from the subjects. The difference among the systems was close to significant (Kruskal-Wallis rank sum test, $p = 0.053$) and the bigger standard deviations of the visualization-based systems (over 0.9) may worth noting. According to the open interview of the subjects, almost all of them found the visual systems new and complicated at first sights, even though they eventually got used to the systems and liked them. The bigger variances of the visualization systems may reflect the mixed feelings of the participants.

Table 18 compares the relative positive reactions between the baseline and the visualization systems. It summarizes how many subjects preferred System A to System B. As seen in the previous result, TaskSieve achieved more votes than the Adaptive VIBE systems. However, there were good number of participants who gave the same ratings to the

Table 18: Comparisons of relative positive reactions

TaskSieve vs. VIBE	TaskSieve	Same	VIBE
Positive count	18	9	6
TaskSieve vs. VIBE+NE	**TaskSieve**	**Same**	**VIBE+NE**
Positive count	16	12	5
VIBE vs. VIBE+NE	**VIBE**	**Same**	**VIBE+NE**
Positive count	4	20	9

TaskSieve and the Adaptive VIBE systems. That is, about the same number of the participants showed equivalent or higher positiveness to the Adaptive VIBE systems despite their greater complexity. It is also interesting to notice that VIBE+NE got twice as many votes as VIBE (four versus nine). Even though more votes were neutral (20), it reflects that the participants acknowledged the advantage of NE-based user models.

We also checked whether there was any gender difference in terms of the positive reactions to the systems (Table 19). We found no significant difference between male and female participants (Kruskal-Wallis rank sum test) regarding the search systems.

Table 19: Comparing positive reactions by gender

	TaskSieve	VIBE	VIBE+NE
Male	3.78	3.30	3.52
Female	3.70	2.90	3.10
p	0.717	0.377	0.329

Table 20: Subjective feedbacks – satisfaction on the document snippet quality

System	TaskSieve	VIBE	VIBE+NE
Mean	3.58	3.48	3.48
SD	0.83	0.97	0.97

6.1.3 Document Summary Quality and Link to the Opening Action

Table 20 is the participants' satisfaction level on the document snippets that showed the summary of the fulltext even before opening it. It was not surprising that there was no difference among three systems (Kruskal-Wallis rank sum test, $p = 0.92$), because the visualization systems shared the same routine to generate the snippets with TaskSieve.

What was more interesting in Table 21 was the relationships between the summaries and the opening actions. Regardless of the system (Kruskal-Wallis rank sum test, $p = 0.18$), the snippets were regarded positively to lead to open the full texts. These responses were consistent with the observations made from the objective data analysis.

Table 21: Subjective feedbacks – contribution of the snippets to open fulltexts

System	TaskSieve	VIBE	VIBE+NE
Mean	4.18	3.82	3.94
SD	0.85	0.80	0.90

6.1.4 Summary of Findings

The subjective feedbacks of the participants were presented in three categories: (1) topic familiarity and difficulty, (2) positive reactions to the systems, and (3) document summary quality.

They had no or very minimal prior knowledge about the three topics used in this study. Even though we tried to keep the difficulty of the topics equivalent, the subjects found there were easier topics than the others.

We expected the participants would consider the Adaptive VIBE visualization system more complex than the baseline system. However, the subjective feedbacks did not show significant difference.

Lastly, the participants showed very positive reactions to the document summary, which was used across all three systems, as a useful tool to guide them to the relevant documents.

6.2 ANALYSIS OF USER ACTIVITIES

6.2.1 Query Length Analysis

Even the visualization-based search systems such as Adaptive VIBE start the operations with queries. The users need to enter their first information needs in the form of textual queries, and then the systems can start activating more sophisticated techniques such as personalization or visualization.

Table 22 and 23 show the average lengths of the queries the subjects entered during the user study. The length here is defined as the number of words. First of all, they show that the query lengths were longer than those of the casual Web search, which are just 2∼3 words [57]. This result is reflecting the nature of this study and its participants. The background and the role of the participants of this study were more focused on the information analysts rather than the casual Web search. Therefore, they tried to formulate their information needs as precise as possible and the efforts lead to longer query lengths. There was no or very small differences in terms of the query lengths among the systems or the topics.

Table 22: Comparison of average query length by system

System	TaskSieve	VIBE	VIBE+NE
Query Length	5.1	5.1	4.9

Table 23: Comparison of average query length by topic

Topic	40009	40021	40048
Query Length	5.2	4.9	5.1

Table 24: Analysis of visual feature usage

Feature	POI Manipulation	Marquee-selection	Auto marquee
Number of users used the feature	33	19	10
Total usage count	1,315	89	516

6.2.2 Visualization Feature Usage Statistics

A lot of features are supported by Adaptive VIBE. Some of the features were liked by the subjects but some of them were not. Even though we tried to provide only essential functionalities to users in order to avoid confusion, there still existed some mis-assessment of the user needs.

Table 24 illustrates the usage statistics of three important visual features: POI manipulation, marquee-selection tool, and automatic marquee tool. Just as our expectation, the POI manipulation function of VIBE was used by all 33 subjects and they recorded relatively high overall frequency. Each user manipulated the POIs around 40 times on average. However, the marquee-selection feature received much less interest. Only 19 out of 24 users[3] tried it and the per-user usage count was only 4.6.

This is because the subjects preferred to open documents directly from the visualization, not following the "marquee-selection → document list → individual document" scenario, which had been expected.

Therefore, after finishing with 24 subjects, we decided to change the user interface so that we could explore the use of marquee. The direct document open feature was disabled

[3] It was not 33 subjects because 9 was provided with a fixed-style marquee tool.

Table 25: Frequency of POI manipulation

POI Type	VIBE	VIBE+NE
Query	356	303
User Model (all)	299	297
User Model (keyword)	299	173
User Model (NE)	-	124
Average (between Query and User Model)	328	300

and a fixed-size box was displayed instead of the "click-drag" style marquee, expecting the users would find it more convenient. After the change, the remaining 9 subjects used the auto marquee tool and the average usage count was 51.6 per user. The following sub-sections discusses the characteristics of these features in detail.

6.2.2.1 POI Manipulation The ability to move POI and examine the related documents on the fly is the core feature of the VIBE visualization. From the system log recorded during the user study, the we were able to count the number of the user manipulations (move) of the POIs in various conditions and compared them (Table 25).

It was noticed that the POI manipulation frequency of the query and the user model were about the same (paired Wilcoxon signed rank test, $p = 0.706$). Even though the participants were informed about the importance of the user models, they seem to be naturally inclined to controlling their own representation of information needs by manipulating the query POIs. However, it should not be underestimated that they still showed almost the equivalent interests in the user model POIs too.

6.2.2.2 Marquee versus Direct Document Access As described before, the subjects expressed the unfamiliarity with the marquee tool. Rather, they preferred directly opening documents from the visualization. Table 26 shows that over 500% more documents were opened directly from the visualization, than from the document lists generated by the

Table 26: Sources of the “Document Opening” action

Source	Overall	Phase 1	Phase 2
Directly from VIBE	565	565	0
From document lists (generated by marquees)	402	111	292
Average	484	338	146

Table 27: Page navigation

Page	1	2	3	4	5	6	7	8	9	10	11
Frequency	**290**	1	1	1	0	0	0	0	0	2	8

marquee action (Phase 1).

It was assumed that the users had found the dragging action required of the marquee-selection inconvenient and eventually disliked the feature. Therefore, a new marquee was devised that did not need the dragging and the direct document opening feature was disabled (Phase 2). In consequence, the number of documents opened from the document lists increased about three times.

6.2.3 Page Navigation

Page navigation does not exist in the visualization-based systems. One of the strengths of the visualization is that it does not need any manipulation to jump between pages as in the text-based search systems. However, in order to contrast the difference between the text- and the visualization-based systems, it is worth showing the usage statistics of the page navigation of the text-based system (TaskSieve).

Table 27 shows the frequency of the page accesses during the user study. Almost all (96%) of the actions were done in the first page. Even the second page was not virtually accessed at all (only once from 33 subjects!). It is already well known that users usually navigate one or two pages in the casual Web search [57].

The characteristic of the personalized search systems can be the reason of this extremely high concentration on the first page. Personalized search systems usually promote relevant documents very high up to the ranked-lists, in the first or the second pages. Users tend to have a bias that they found everything they need after checking the first or second pages and then give up the deeper navigation down to the lower pages [63].

However, in case of the visualizations, users showed quite different behaviors. A related analysis about the navigation depth is presented in section 6.5.3.

6.2.4 Summary of Findings

In this section, we examined some basic user behaviors extracted from the log data. We had planned to recruit the participants who could act as information analysts. Their behaviors analyzed in this section corresponds to the expectation. They tried to formulate longer queries rather than casual short queries. They also manipulated a lot with the POI of the VIBE visualization, which represents their efforts to explore the visualization spaces.

Even though the depths of the navigation using the baseline personalized search system (textual) were shallow, they showed quite different behaviors with the visualization systems. Finally, the participants preferred to use the direct access of documents in the visualization, rather than the marquee selection.

6.3 SYSTEM PERFORMANCE – ANALYSIS OF RANKED LISTS

This section compares the performance of ranked lists generated by all three systems – TaskSieve, Adaptive VIBE, and Adaptive VIBE plus NEs. For Adaptive VIBE, the ranked lists are like raw materials for generating the visualizations. Using the personalized search results, Adaptive VIBE can produce the adaptive visualizations. Therefore, it is essential to compare the ranked lists first, and then we can move on to the next stage.

There are two types of ranked lists in Adaptive VIBE: (1) marquee selection (Section 3.2.4.5) and (2) visual cue of high rank documents (Section 3.2.4.6). Using the marquee-selection tool included in Adaptive VIBE, users can retrieve a list of documents and they are comparable with the ranked lists of TaskSieve. The top 10 documents with highest relevancy scores are marked with different colors on their icons in the visualization.

Following these features, two types of analysis are presented: (1) comparing ranked-lists and marquee-selection and (2) comparing top 10 documents from the text search and the visualizations.

6.3.1 Measures

We used a family of precision measures for comparing the system performances. Precision is the ratio of relevant documents and the retrieved documents and Precision@N means the top N retrieved documents were considered to calculated the precision.

Due to the fact that the users mostly check 1∼2 pages using the search engines and do not go deeper down to the lower rank items ([57] and Section 6.2.3), the calculation of the precision was done at rank 10 of each list.

Therefore, the system performance while using the TaskSieve (Condition 1) system was defined as follows.

$$Precision@10 = \frac{|Doc_{rel}|}{10} \tag{6.1}$$

Along with the traditional Precision@10, one additional measure was adopted. DCG (Discounted Cumulative Gain) is a measure devised to calculate the usefulness, or gain, of a document based on its position in the result list. The gain is accumulated cumulatively from the top of the result list to the bottom with the gain of each result discounted at lower ranks [58]. NDCG (Normalized DCG) normalizes DCG by an ideal DCG at position p. In this study, NDCG_{10} was used.

$$DCG_p = rel_1 + \sum_{i=2}^{p} \frac{rel_i}{log_2 i} \tag{6.2a}$$

$$NDCG_p = \frac{DCG_p}{IDCG_p} \tag{6.2b}$$

We could use these measures for Adaptive VIBE systems (Condition 2 & 3) too, because they generated the ranked lists internally and the top 10 highest stored documents were marked using different colors in the visualizations.

Adaptive VIBE supported a function called a marquee tool (Section 3.2.4.5) in order to select a set of documents from the visualization plane, and then could retrieve the documents in the list format so that the users could examine the document summaries, make notes from the summaries, and open the full-text of seemingly relevant documents.

Therefore, we could define a measure that was comparable to the system performance of the TaskSieve system. It was defined as follows.

$$P_{marquee} = \frac{|Doc_{rel}|}{|Doc_{marquee_selected}|} \tag{6.3}$$

Here, the marquee selected documents ($Doc_{marquee_selected}$) are the documents selected using the marquee-tool.

6.3.2 Precision of Marquee-selected Documents

The document list generated by the subjects' marquee-selection actions is one of the most ranked-list like elements of the experimental visualization systems. In fact, the format of the retrieved documents (document title and highlighted summaries) was exactly same with the ranked-lists of the baseline. Table 28 compares the precision of the retrieved documents of the baseline as the precision at rank 10 and the precision of the marquee-selected documents. There was no difference found among the three systems.

Even though the experimental systems did not show any improvement over the baseline, it cannot be an evidence that they were worse than the baseline. Actually, the comparisons between the different sets – ranked-lists and marquee-selection – were not quite fair, because of the following reasons.

Table 28: System performance comparisons – ranked-list versus marquee-selection

System	TaskSieve (Precision@10)	VIBE (Precision$_{marquee}$)	VIBE+NE (Precision$_{marquee}$)
Precision	0.873	0.823	0.817

- If the number of marquee-selected documents are more than 10, it increases the chance to contain non-relevant documents.
- Even though they are not more than 10, due to the nature of the visualization that usually displays hundreds of documents, the chance to contain noises becomes very high.
- The participants preferred to directly open documents, which was much more familiar to them (section 6.2.2.2).

It should be noted that the marquee selection could show almost similar performance compared to the ranked lists, even in these unfair conditions. The next section investigates another side of the system performance that can complement the analysis of the marquee selection precision, top-10 document precision.

6.3.3 Precision of Top 10 Documents

This section compares the quality of the ranked lists in order to compare the system performance. Even though the Adaptive VIBE systems replaced the ranked lists with the adaptive visualizations, their personalization engine still generates ranked lists internally before converting them to the visualizations. Therefore, we can compare the precision of the ranked lists first.

Precision@10 and NDCG$_{10}$ were used for comparing the ranked lists. Precision@10 was selected because the first page of the TaskSieve search contained 10 documents. NDCG$_{10}$ could measure the distribution of relevant documents higher in the top-10 document lists.

Table 29 and Figure 35 show the comparison of the scores across the three systems: TaskSieve (baseline), VIBE (Adaptive VIBE visualization) and VIBE+NE (Adaptive VIBE

Table 29: System performance comparisons

System	Precision@10	$NDCG_{10}$
TaskSieve	0.873	0.924
VIBE	0.919	0.966
VIBE+NE	0.895	0.950

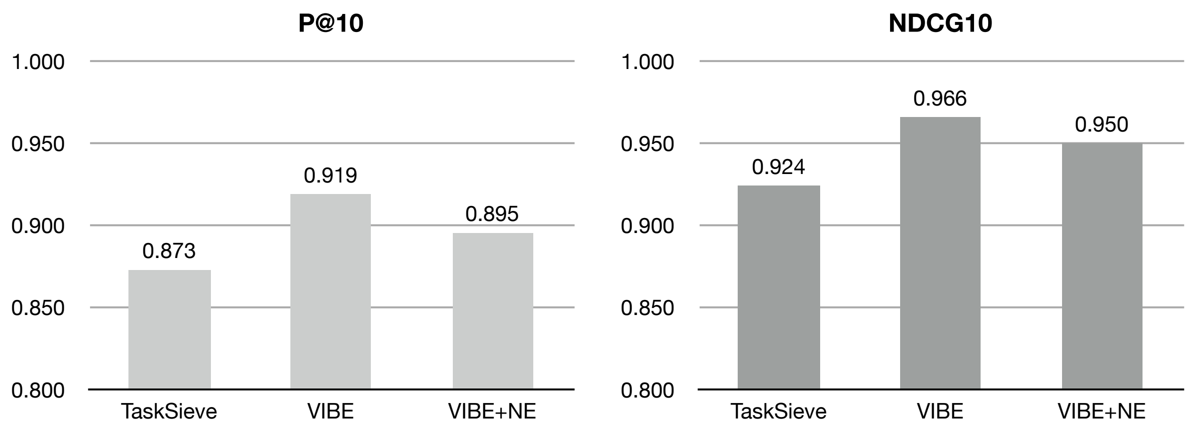

Figure 35: System performance comparisons

with NE-based user model). The statistical tests showed no difference in terms with Precision@10, whereas $NDCG_{10}$ showed significant differences between the baseline and VIBE (Wilcoxon signed rank test, $p = 0.0289$), and between the baseline and VIBE+NE (Wilcoxon signed rank test, $p = 0.0331$).

This result is encouraging because it suggests that the Adaptive VIBE visualizations were able to provide more relevant documents than the baseline system's ranked-lists. The source of this improvement could be the better quality of the user models, developed during the cycles of the user-system interactions.

The initial ranked lists with better quality would have led to better visualizations. At the same time, the Adaptive VIBE prototype shows the top 10 highly scored document icons in different shades. The document with the highest score is the darkest and the 10th document is the least dark, but still distinguished from the lower rank documents. Users could have used the document colors as reliable cues leading to the relevant information.

6.3.4 Summary of Findings

The first step of the objective log analysis was the system performance analysis. We intended to measure the performance of ranked lists generated by the text-based baseline system (TaskSieve) and the Adaptive VIBE systems. The visualization systems generated the ranked search results first and then used them in order to construct the visualizations.

We used the precision measure for the marquee-selected documents and the top 10 highly ranked documents. The marquee-selected document precision was almost the same with the traditional ranked lists even in the unfair conditions and we found that two Adaptive VIBE systems (user models with keywords or named-entities) provided more precise rankings than the baseline in terms of NDCG_{10}.

6.4 SYSTEM PERFORMANCE – ANALYSIS OF VISUALIZATION

6.4.1 Cluster Analysis

Using the documents retrieved by the personalized search engine, Adaptive VIBE can generate the adaptive visualizations. In the previous section, we showed that Adaptive VIBE could produce better search results than the baseline, in terms of the top-10 highest relevancy scored documents. The next question is the quality of the visualizations that were generated from these better results.

We can answer this question by examining the separation of the relevant and non-relevant document cluster in the visualization. At the same time, our interest is to check if the user models were still able to attract the relevant documents. These effects have already been observed in our pilot studies (Section 4.1 and 4.2).

The difference is that the pilot visualization experiments were simulations, using the log data extracted from the TaskSieve study, whereas this section presents the results from the

Table 30: Comparison of the x-coordinates of the relevant and non-relevant document cluster centroids

Cluster	Relevant	Non-relevant
Average of the x-coordinates	458.03	379.55

Table 31: Comparison of DB-index between systems

System	VIBE	VIBE+NE
DB-index	1.70	1.69

real user study.

Table 30 compares the x-coordinates in the visualizations generated during the user study. As in the pilot studies, the average location of the relevant documents was closer to the user model in the Adaptive VIBE visualizations. The average distance between the relevant and non-relevant document clusters was statistically significant (paired Wilcoxon rank sum test, $N = 743, p < 0.001$).

In order to test the second hypothesis of this dissertation – Adaptive VIBE with NE-based user models will generate better results – we calculated the DB-indices of the visualizations created by those two systems separately (Table 31). Even though the VIBE+NE showed less DB score (represents better clustering) and the difference was significant (Kruskal-Wallis rank sum test, $p < 0.001$), the size of the difference was rather small. Therefore, a follow-up analysis was conducted by separating the data by topics (Table 32 and Figure 36).

Among the three topics, the topic ID 40048 showed the worst performance with VIBE+NE and the difference was large and significant (Kruskal-Waillis rank sum test). Two systems showed an exactly opposite result with the topic 40009, where VIBE+NE was significantly better than the keyword-based Adaptive VIBE system. There was no difference with the topic 40021. This topic difference was already noted by the participants' subjective feedbacks, which made it clear that the topic 40048 was the easiest (Section 6.1.1).

Table 32: Comparison of DB-index between systems and topics

System	VIBE	VIBE+NE	p
40009	1.62	0.98	< 0.001
40021	2.35	2.38	0.1412
40048	0.68	1.88	< 0.001

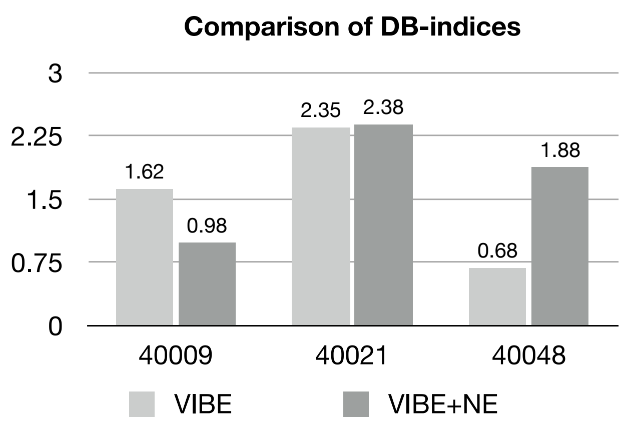

Figure 36: Comparison of DB-index between systems and topics

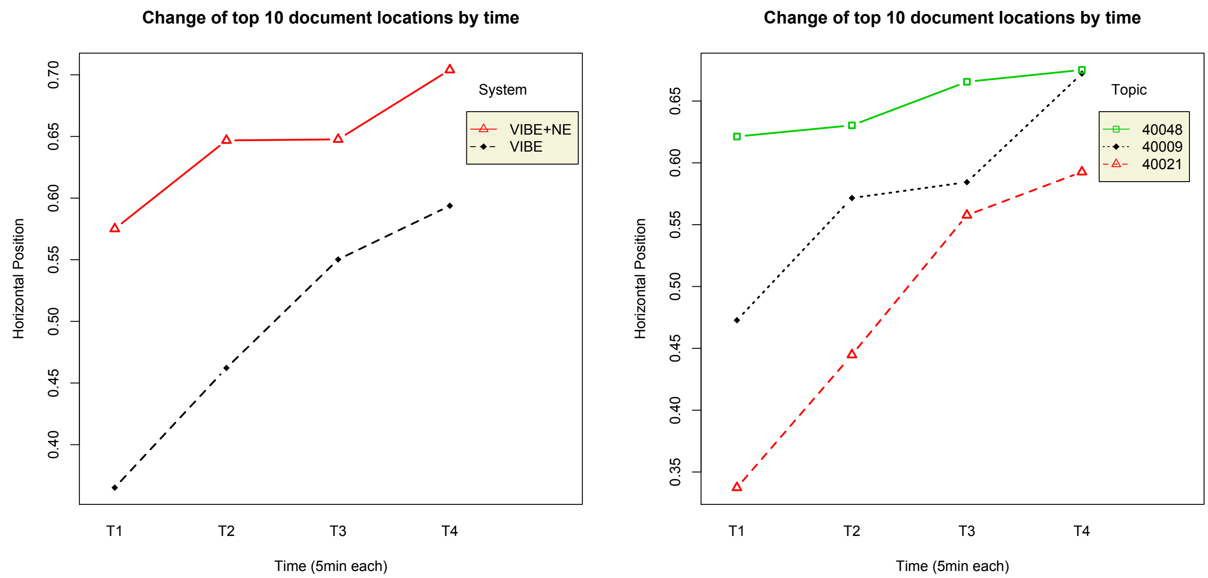

Figure 37: Movement of top 10 documents through search progress

6.4.2 Position of Top 10 Documents

In addition to the cluster analysis, we can examine the top 10 documents of the ranked list generation stage. We have already seen that Adaptive VIBE could produce those documents with high quality in Section 6.3.3. Because these documents are annotated with high intensity colors in the visualizations, it is important to check how they are actually represented in the visualizations, in terms of their positions.

Figure 37 shows the change of the top 10 document positions as the subjects continued their search processes. As can be seen from the graph, the horizontal position (y-axis) is changing regardless of the system (left) and the topic (right).

The horizontal position here is the relative position of the documents in the visualization, when the position of the left-most document is set to 0 and the right-most document is set to 1. Therefore, the position $= 0.5$ means that the document was exactly at the center of the document group displayed on the screen. Larger number represents it was closer to the user model, because the user model was placed at the right hand side of the visualization.

Table 33: System performance comparisons

	Before UM built		After UM built	
Position	Left-half	Right-half	Left-half	Right-half
Count	117	116	156	387
Ratio	0.50	0.50	0.29	0.71

This result means that the top 10 documents were moving toward the user models as time passed and more information was accumulated into the user models. For example, the top 10 documents presented by the Adaptive VIBE system (dashed line in Figure 37, left) were almost at the left-end side of the visualization, as the Horizontal Position was around 0.35 at the beginning of the experiment (T1). However, it rapidly moved toward the right-half side of the visualization through the middle of the experiment (T3). The NE user model-based Adaptive VIBE (solid line) showed even bigger inclination toward the user model.

Table 33 and Figure 38 show another example of the relationship between the top 10 documents and the user model. They compare the frequency of the top 10 documents according to the location – in the left-half or right-half of the visualization. When there was no user model, in the very early stage of the search, the position of the top 10 documents were exactly the same in terms of their horizontal locations. However, if we count the instances *after* the UM built, this ratio of the top 10 documents that were placed right-half side of the visualization increases beyond 70%. That is, it is very probable that the top-10 documents were placed closer to the user model.

The top 10 documents do not consider the position of the documents by itself. However, due to the nature of the user model in the personalized search, the top scored documents were aggregated closer to the user models in the visualization. In a sense, the combination of color (which was used for the top 10 documents) and the position (automatically moving toward the user model) may be a better method to guide the users to relevant documents.

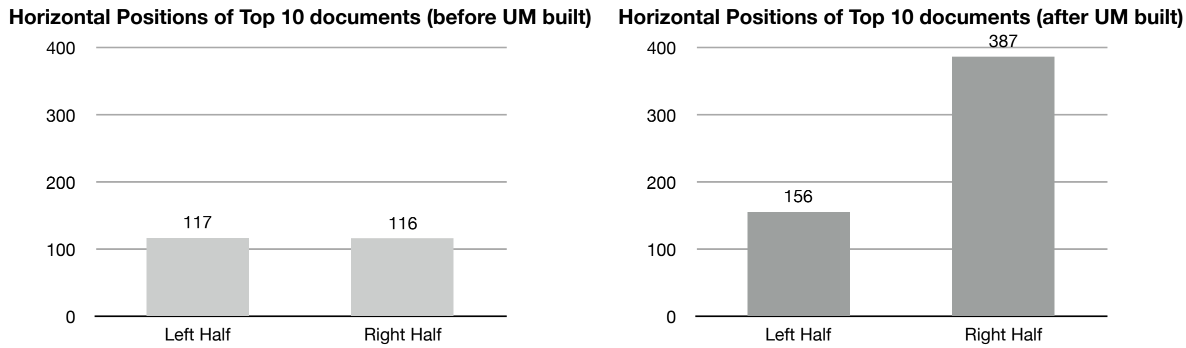

Figure 38: Movement of top 10 documents through search progress

6.4.3 Summary of Findings

To sum up, the visualizations generated during the user study could separated the relevant and non-relevant document clusters and the relevant document clusters were located closer to the user model.

When compared the keyword-based user models and the NE-based user models, we showed that the NE-based user models made significantly better separation of the document clusters, especially with the more difficult topic.

We also observed the top 10 documents were moving toward the user models in the adaptive visualizations. This result justifies the ability of Adaptive VIBE to guide the users to the relevant information.

6.5 USER PERFORMANCE – DOCUMENT ACCESS

When the documents are presented to the users through ranked-lists or visualization, the users examine the provided information and open a document to read the fulltext. They can check the visual cues such as the colors or the locations of the document icons, rank of the documents within the lists, or read summaries of the document in order to quickly decide which document they should open.

This "opening" action is done by the users but it originates from the system output. It is more like the interaction between the user and the system, rather than the result of independent user actions. Therefore, by analyzing the document access behaviors of the users, we can learn how well each system contributed to the user behaviors.

6.5.1 Measures

6.5.1.1 Open Precision The most common document access method is to open the document by clicking the document icons or titles and read the fulltext. We could measure the precision of the opened documents and was defined as Equation 6.4.

$$P_{open} = \frac{|Doc_{rel}|}{|Doc_{open}|} \tag{6.4}$$

If a system could provide a good ranked list or a visualization and they were accompanied by proper information – such as summaries, document icon colors, or document location closer to the user models in Adaptive VIBE visualization – users will have higher chances to open relevant documents without making errors.

6.5.1.2 Depth of Navigation and Diversity Recollecting the fact that users tend not to navigate deeper into the ranked lists and they mostly review 1 or 2 pages (10 to 20 documents if one page shows 10 documents), we could easily expect the relevant documents or passages found by the users using the baseline will mostly come from within rank 20. This tendency would be stronger because the baseline was a personalized search system, which tried to promote relevant documents to the top of the list.

However, it was still possible that some relevant documents were hiding at the bottom of the ranked lists, such as at rank 90 or rank 100. Our visualization-based approaches could alleviate this weakness of the ranked lists. They could show 100∼200 documents easily at the same time and would make it much easier for users to locate the low-ranked, possibly relevant documents. This could be measured by comparing the ranks of the relevant documents or passages found by the users.

$$NavDepth_{doc} = Rank(doc) \tag{6.5}$$

A similar aspect is the diversity of findings. The precision measures in the previous sections considered the correctness of the findings only. They did not care about what fraction of the information in the pool was discovered. Diversity is a measure adopted to reflect this perspective. It was defined as follows.

$$Diversity_{topic} = \frac{|Subtopics_{notebook}|}{|Subtopics_{groundtruth}|} \tag{6.6}$$

It looks similar with the traditional recall measure, but it is topic-based, rather than document-based. The LDA (Latent Dirichlet Allocation) [15] algorithm was used for finding out the sub-topics. Detailed procedures and results are presented below.

6.5.2 Open Precision

Table 34 and Figure 39 compare the precision of the opened documents by the subjects. In order to make the comparison clearer under the situation when the user model is working, the document-opening actions after the user models built were counted. The best was the baseline (TaskSieve), which showed a very high average precision score. The experimental systems showed slightly less precision scores, but there was no significant difference between the baseline and the experimental systems (Wilcoxon signed rank test).

Table 34: Precision of opened documents by the three systems (after UM built)

System	TaskSieve	VIBE	VIBE+NE
Open Precision	0.950	0.875	0.875

There are two issues about this result. First, what is notable is the very high performance of the baseline (TaskSieve). The mean score of TaskSieve was 0.950 and there were a lot of

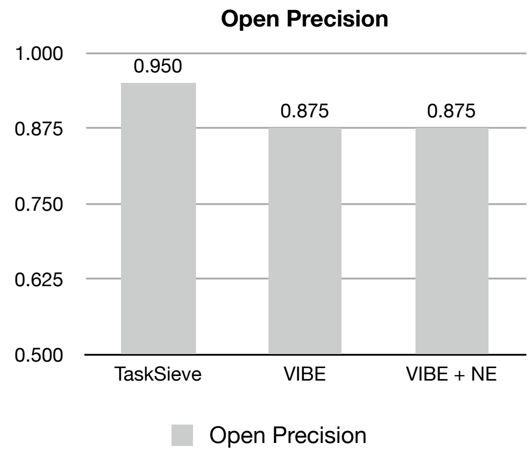

Figure 39: Precision of opened documents by the three systems (After UM built)

perfect precision scores (1.0) depending on the subjects[4]. It was a challenging task to beat the already perfect performance of the baseline system.

The second issue is the condition where the document-opening actions were made. The reason why the baseline was able to record this good performance was the existence of the document summaries. The subjects of the baseline system mostly explored the first page of the ranked-lists (Table 27) and the average precision of the first pages was at least 0.85 (section 6.3.3). On top of this high quality system output, the users were able to examine the document contents *before* they open the full-text. This additional effort of the subjects could have boosted the performance of the baseline system.

Of course, the experimental systems (Adaptive VIBE with/without the NE-based user model) were under the similar condition. They provided the same document summaries and the precision of the top-10 documents were even better than the first page of the baseline system.

However, we should note that the subjects of the visualization-based system had more freedom to explore the document space and they had more chances to make mistakes, even though they were provided with quality results from the system. In fact, there were hundreds

[4]This score is almost consistent with the average precision found in the previous TaskSieve study, which was 0.96 [5].

of documents spread in the visualization and a single click on a non-relevant document by mistake could lead to the large drop of open precision.

6.5.3 Depth of Document Navigation

Even though the open precision could not show any improvement of the experimental systems, it was not necessarily discouraging. It gave us a hint that the increased degree of freedom could benefit the experimental systems. One possible benefit was the depth of the navigation.

The previous analyses only showed whether the findings of the system or the subjects were correct or not. They did not show other perspectives such as the coverage or the diversity of the results.

Table 35: Average ranks of opened documents

	Open Rank		Noted Rank	
	Overall	Relevant	Overall	Relevant
TaskSieve	4.00	3.96	3.63	3.60
VIBE	21.76	13.26	10.73	9.44
VIBE+NE	24.40	10.77	11.43	11.49

Table 35 and Figure 40 illustrate the rank of the opened documents under various conditions. Here, we could clearly see that there were differences between the baseline (TaskSieve) and the experimental systems (VIBE, VIBE+NE). The average rank of opened documents of the baseline was around 4, whereas the visualization-based experimental systems went down to around rank 20.

Even when the relevant documents were solely counted ("Relevant" columns), the visualization systems marked around rank 10. Of course, there is no visible rank below 10 (only the top 10s were highlighted) in the visualization-based systems but that may be one of the reasons the subjects were able to find low-ranked but relevant documents. They could have more freely examine low-ranked documents and then have chosen relevant ones. The rank of

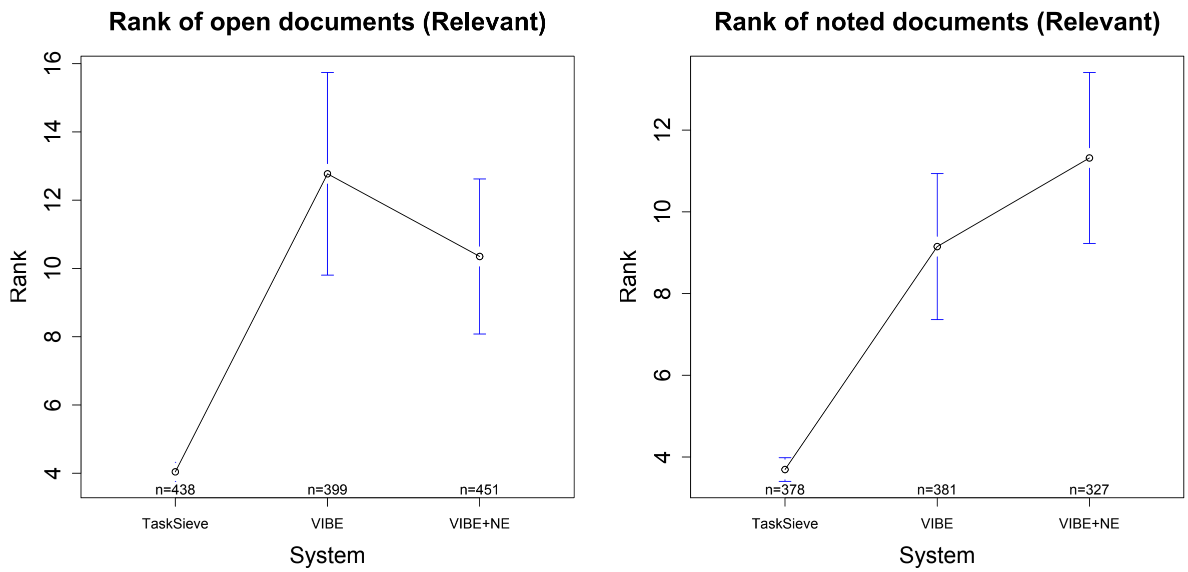

Figure 40: Average ranks of opened documents

the noted documents – from where the user notes were annotated – showed similar results ("Noted Rank" column).

6.5.4 Diversity of Opened Documents

It can be understood that the visualization-based systems' users were able to find more diverse set of documents, according to the rank-depth analysis in the previous section. At the same time, they were not only random documents that were not visible in the higher ranks. Rather, the subjects were still able to find relevant documents in the lower ranks that may not be able to be found by using the baseline system – as can be seen in the "Noted Rank" data.

In this section, this analysis is expanded even further by examining sub-topics (they were not the task topics: 40009, 40048, and 40021 in this experiment) found by the subjects. The aim of the analysis was to calculate the portion of the discovered sub-topics out of the entire set of relevant sub-topics from the groundtruth information.

The concept may look similar to the recall of information retrieval evaluation, which is the fraction of number of retrieved documents and the total number of relevant documents. However, the analysis here is intended to be done in the topic level, rather than the traditional document level recall.

The analysis procedure is as follows:

1. Locate the relevant documents from the groundtruth per each topic (40009, 40048, and 40021)
2. Find out the sub-topics from the relevant document sets.
3. Assign each opened document to the sub-topics found in the previous step.
4. Compare the number of the discovered (assigned) sub-topics.

For the step 2 and 3, Latent Dirichlet Allocation (LDA) [15] was used. It is a generative probabilistic model for collections of discrete data such as text corpora and can be used for detecting latent topics from the text and cluster the documents to them. Therefore, we could first apply the LDA algorithm to the relevant documents and extract the sub-topics from them (step 2). In the next step, the list of opened documents by the subjects were applied to each sub-topic.

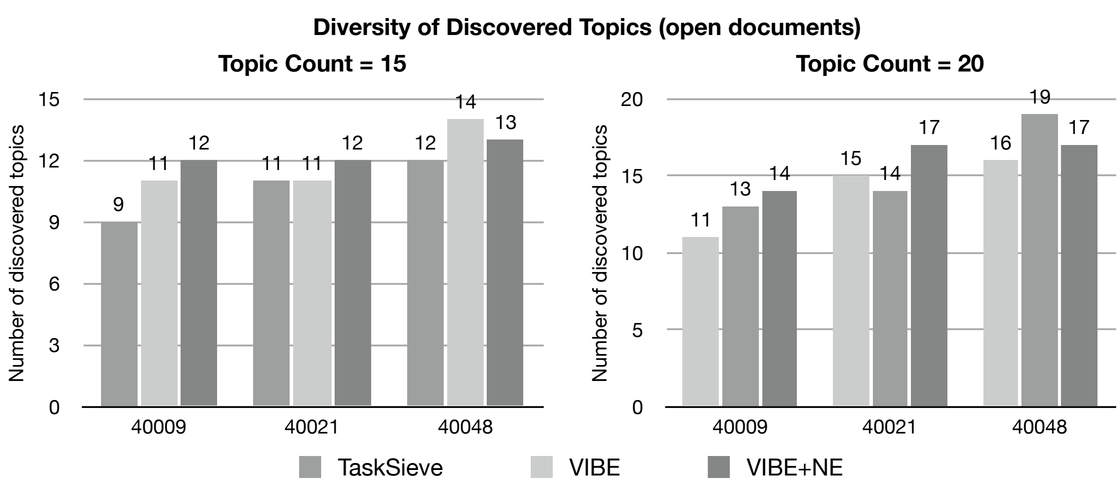

Figure 41: Number of sub-topics discovered by opening documents

Table 36: Number of sub-topics discovered by opening documents

		Task Topic		
Sub-topic Count	System	40009	40021	40048
15	TaskSieve	9	11	12
	VIBE	11	11	14
	VIBE+NE	12	12	13
20	TaskSieve	11	15	16
	VIBE	13	14	19
	VIBE+NE	14	17	17

Because LDA requires to specify the number of sub-topics to be estimated from the corpus and the real values are not known, two values were tried for each topic, k=15 and 20. We chose these values assuming that the number of sub-topics will be greater than the number of questions per topic. Because the topic 40021 had the maximum number of questions ($N = 13$), we tried $k = 15$ and $k = 20$. However, it should be noted that different topics can have different number of sub-topics and more sub-topics could benefit from the VIBE visualizations. This nature of LDA might limit the accuracy of the analysis.

Table 36 and Figure 41 list the number of sub-topics discovered by the subjects in terms of their document opening actions. It can be seen that experimental systems (either VIBE or VIBE+NE) always found more sub-topics than the baseline. The difference can look rather smaller (less than 4) but a single difference of sub-topic discovery can mean a lot to the given problem domain.

6.5.5 Summary of Findings

In this section, we analyzed the precision and the diversity of users' document access (open) behaviors. There was no improvements in terms of the average precision of the accessed documents with Adaptive VIBE.

However, according to the analysis of the depth of navigation and the diversity of sub-topics discovered by the participants, the Adaptive VIBE systems were able to help the users to find out more relevant information than the baseline. The Adaptive VIBE users were able to overcome the nature of ranked lists that bias their users only on top of the lists.

Using the visualization systems (Adaptive VIBE or Adaptive VIBE+NE), the subjects navigated much deeper down to the ranked lists compared to the baseline. They examined more documents residing in the lower ranks (while still relevant) and were able to discover more diverse information.

The sub-topic analysis result coincides this navigational characteristics. The Adaptive VIBE visualization systems could find more sub-topics than the baseline.

6.6 USER PERFORMANCE – NOTE ANNOTATION

This section analyses the quality of the notes collected by the users. After receiving the search results from the system and checking the documents included in the output of the systems, the participants found the answer to the questions of the tasks. They were asked to save them into the notebook as the final report of their search process. By analyzing the participants' annotations saved in the notebook, we can evaluate the ultimate results of the three systems.

6.6.1 Measures

Two measures were employed to evaluate the quality of the user annotations. First one is the precision of each annotation. Unlike the document precision where the relevance of individual document access action was regarded as 1 or 0, note precision can give precision value $0 \sim 1$ per each note.

The second measure is the note diversity. Note precision only considers correctness of user annotations. However, it is an important issue to check whether the user annotations covered as much answers as possible. For this purpose the LDA-based sub-topic analysis was repeated, in the level of user annotations.

6.6.1.1 Note Precision The precision of a user annotation is defined as follows.

$$\frac{\sum_{i \in passage \cap groundtruth} overlap\_length_i \times weight_i}{\sum_{i \in passage \cap groundtruth} overlap\_length_i \times weight_i + \sum_{i \in passage - passage \cap groundtruth} miss\_length_i \times 0.5} \tag{6.7}$$

It is the same formula used in the TaskSieve study [5, 50], which was derived from (Allan, 2003). The calculation of passage precision takes advantage of the fact that two human annotators generated the ground truth. It calculates the precision of a passage against the ground truth, where *overlap_length* is the character length of the common text chunk between a user's selection and the ground truth; weight is the weight of the ground truth combining the two annotators mark-ups, where the weight can be one of five levels: 0, 0.25, 0.5, 1, 1.25, 2; *miss_length* is the character length of the part of the passage that has no overlap with the ground truth. Here the 0.5 associated with *miss_length* is the penalty.

6.6.1.2 Note Diversity The same procedure used for measuring the document access level diversity was used in the note level. Please refer to Section 6.5.1.2 and Section 6.5.4.

6.6.1.3 Productivity If the diversity increases, it could lead to higher productivity. The one who could found some hidden gems would be able to record higher stacks than others. The productivity is defined as the number of found answers (and saved to the notebook) in a given time. Because every participant was allowed to use just 20 minutes per topic, the simple count can be used as a productivity measure.

There is a risk of duplicate entries in the notebook that will inflate the measure. However, the coordinator asked the participants to arrange the notebooks and mark the note numbers on the answer sheet after finishing each session. Therefore, unnecessary duplications were filtered out by the participants themselves.

$$Productivity = \frac{|Notes|}{Time} \tag{6.8}$$

6.6.2 Note Precision Analysis

Table 37 summarizes the mean note precision by the three systems. First of all, there were significant differences among the three systems (Kruskal Wallis rank sum test, $p < 0.001$). According to the pairwise comparison results (Table 38), there was a significance difference between the baseline (TaskSieve) and Adaptive VIBE with NE-based user models (VIBE+NE). However, no difference was found between the baseline (TaskSieve) and the Adaptive VIBE (VIBE).

Table 37: Note precision

System	TaskSieve	VIBE	VIBE+NE
Note Precision	0.795	0.817	0.866

Table 38: Pairwise comparisons by Wilcoxon rank sum test

System	TaskSieve	VIBE
VIBE	$p = 0.296$	-
VIBE+NE	$p = 0.011$	$p = 0.110$

In order to understand the nature of this difference, the notes were split by the topics and their precisions were compared. Table 39 and Figure 42 show the results. There was no difference among systems with topic 40009 and 4008, but a significant difference was found with topic 40021 (Kruskal-Wallis rank sum test, $p < 0.001$). The follow-up test shows that there were significant differences between the VIBE+NE system (precision = 0.888) and two other systems. There was no difference between TaskSieve (precision = 0.790) and VIBE (precision = 0.798, Table 40).

Table 39: Note precision analysis by topic

Topic	40009	40021	40048
TaskSieve	0.698	0.790	0.901
VIBE	0.769	0.798	0.865
VIBE + NE	0.784	0.888	0.931

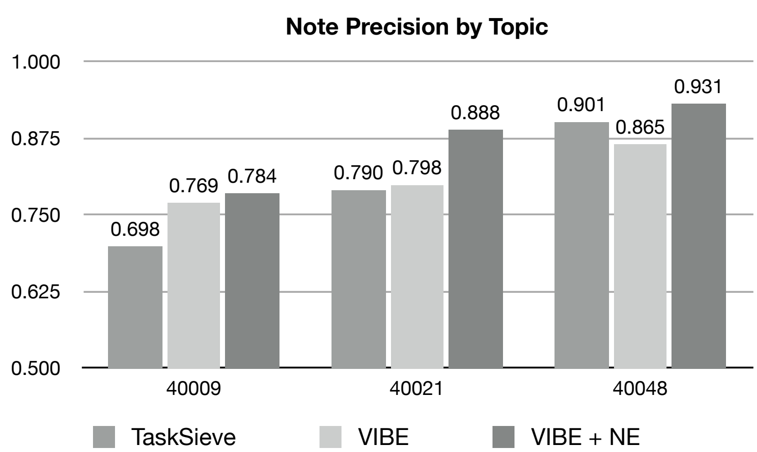

Figure 42: Note precision by topic

Table 40: Pairwise comparisons using Wilcoxon rank sum test (Topic 40021)

System	TaskSieve	VIBE
VIBE	$p = 0.41$	-
VIBE+NE	$p < 0.001$	$p < 0.001$

Even though the differences were found with only one topic 40021, the result still embeds meaning because 40021 was considered as a difficult one. Topic 40048 was the easiest and the higher average note precision scores of three systems directly reveal the lowest difficulty, where no cross-system difference was found. In terms of the topic 40009, the average note precision of the experimental systems were over 0.75 whereas that of the baseline was below 0.7. However, no statistically significant difference was found either (Kruskal Wallis rank sum test, $p = 0.2589$).

6.6.3 Note Diversity Analysis

Along with the precision of the user notes, it is important to check the number of notes found by the subjects. As the analysis results form the previous section suggest, the experimental systems were able to help users to examine more diverse range of documents.

Table 41 and Figure 43 are the latent sub-topic analysis done for the documents from where the notes were saved by the subjects. Note that the same analysis in the previous section was done for the "document opening" actions. This analysis was to examine whether the clusters found by the opening action were actually used for the final product, user notes.

Two things can be noticed from this data. First of all, the experimental systems annotated more sub-topics than the baseline. It is a similar trend with the sub-topic discovery analysis of the previous section. Either the Adaptive VIBE or Adaptive VIBE with NE user modeling annotated more sub-topic than the baseline, except just a single setting (sub-topic count = 15 and the topic id = 40009). However, the baseline could not beat the experimental systems even in this case. They all annotated same 9 sub-topics.

Table 41: Number of sub-topics annotated by saving notes

		Task Topic		
Sub-topic Count	System	40009	40021	40048
15	TaskSieve	9	9	12
	VIBE	9	10	13
	VIBE+NE	9	11	12
20	TaskSieve	9	11	15
	VIBE	12	10	17
	VIBE+NE	11	16	13

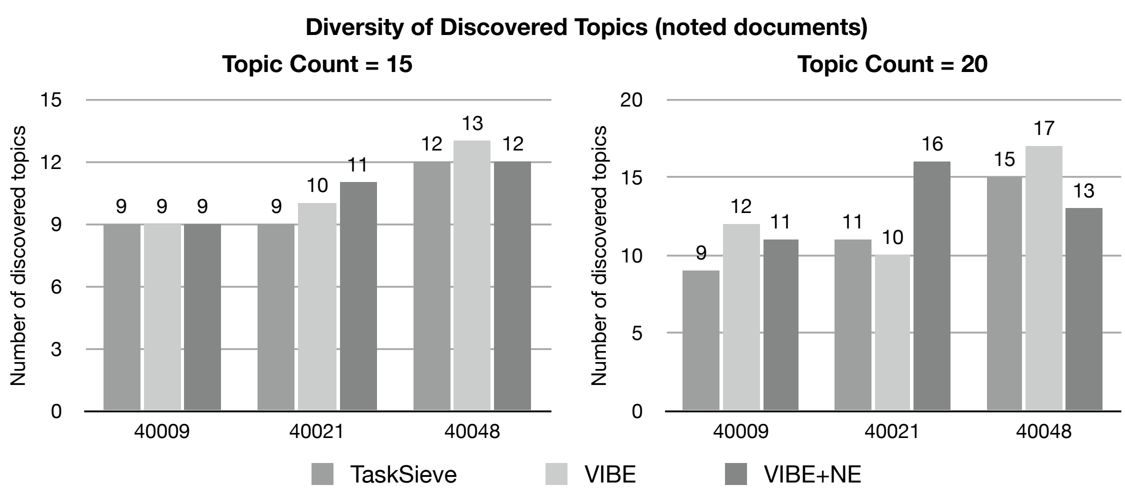

Figure 43: Number of sub-topics annotated by saving notes

Therefore, we can assume that the diverse topic discovery done at the level of document opening repeated in the annotation stage too.

6.6.4 Note Productivity Analysis

There is another evidence we can check in the similar context with the diversity – *note productivity*. The analysis in the previous section just tells us that which system collected more precise information. However, collecting just precise notes is not enough. We need to collect *a lot of* precise notes. The count matters too.

Luckily, we can expect the experimental systems could provide with more diverse outcomes than the baseline, with the increased diversity. Table 42 shows the number of notes annotated by the subjects using three different systems.

Table 42: Note count and precision

System	TaskSieve	VIBE	VIBE+NE
Overall	451	476	423
High-precision notes ($precision > 0.9$)	343	379	355

It shows that the Adaptive VIBE system collected more notes (476) than the other two systems (First row). The NE-based user model system (VIBE+NE) was the worst. However, these numbers consider the quantity only and do not consider the quality this time. The number of high-precision notes, which counts the notes with precision greater than 0.9, shows slightly different outcomes.

The baseline became the worst and VIBE+NE became the second best. This result reflects the analysis result of the previous section, where VIBE+NE recorded the best precision than the others. Even though VIBE+NE did not collect more notes than the baseline, it collected more high-quality data. This difference resulted in the different mean precision.

At the same time, the Adaptive VIBE (VIBE) system's mean note precision could not beat the baseline, but it collected more data. Even though the high-precision (precision > 0.9) note count reflects this quantity+quality phenomenon, we need deeper analysis in order to understand what really happened.

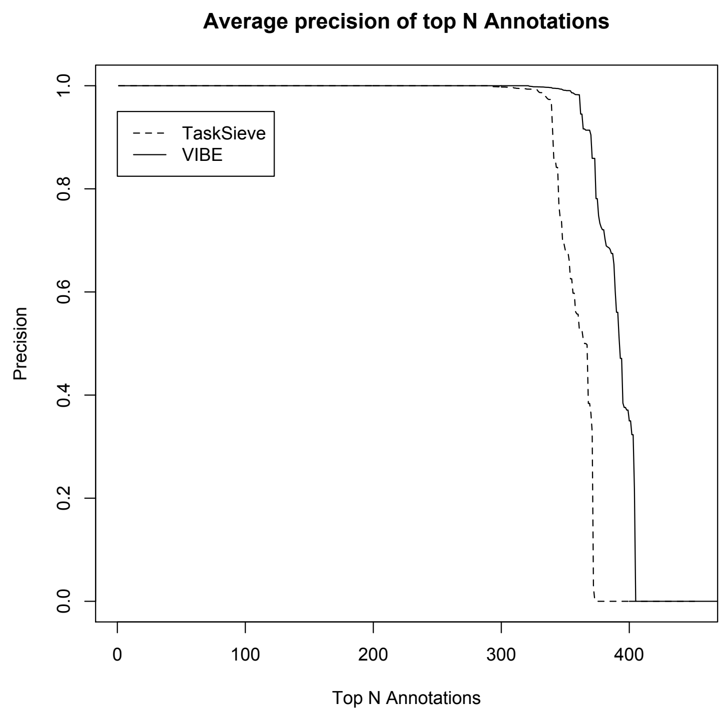

Figure 44: Average precision of top N notes

In order to combine the quality and quantity of notes of the baseline (TaskSieve) and Adaptive VIBE (VIBE), the notes collected by the two systems were sorted in the decreasing order and plotted on the same graph (Figure 44). It shows the change of average precision of top N annotations and compares the average precision scores of the two systems when the subjects made top N high precision annotations. Both are dropping as they collect more passages but the baseline drops more sharply and the experimental system shows higher precision until the N reaches 405. The right end point N=476 is the total number of annotations made using the experimental system and it is where the overall average precision in Table 37 was calculated.

Subjects with the baseline made 451 annotations, so a new average precision at N=451 (where the note counts of both systems are same) was calculated (Table 43). The experimental system showed significantly higher average precision (Wilcoxon signed rank test,

Table 43: Combination of note quantity plus quality analysis

	TaskSieve	VIBE	p
Note precision (Top N=451)	0.795	0.862	< 0.001

Table 44: Combination of note quantity plus quality analysis

	TaskSieve	VIBE+NE	p
Note precision (Top N=423)	0.847	0.866	0.1918

Table 45: Combination of note quantity plus quality analysis

	VIBE+NE	VIBE	p
Note precision (Top N=423)	0.866	0.919	0.1596

$p < 0.001$).

The same analyses were repeated for the remaining pairs (Table 44, 45). However, the no significant difference was found (Wilcoxon signed rank test, $p = 0.1918$ and $p = 0.1596$ respectively).

6.6.5 Summary of Findings

The user annotated notes are the final product of the user study. In this section, we analyzed the precision, diversity, and the productivity of the note annotations.

We compared the note precision across the three systems and found significant differences between Adaptive VIBE with NE user model and two other systems. We also found the topic difference. With the most difficult topic, Adaptive VIBE with NE user model showed the best results.

According to the sub-topic discovery analysis in the note level, the Adaptive VIBE systems found more sub-topics than the baseline. Along with the productivity analysis that counted the number of high quality annotations, we could summarize the lessons of this section as follows.

1. Adaptive VIBE could collect *more* high *quality* notes, while maintaining the equivalent level of note precisions compared with TaskSieve.
2. Adaptive VIBE+NE could collect higher *quality* notes, even though the number of notes did not exceed that of Adaptive VIBE.

7.0 DISCUSSION OF RESULTS

This chapter summarizes the results presented in the previous chapter and attempts to connect each individual result, in order to provide a structure and discuss the implications of the findings.

The goal of this dissertation is testing two key ideas: (1) adaptive visualization (Adaptive VIBE) and (2) conceptual user modeling (Adaptive VIBE+NE). We defined two broad stages according to the systems' working process and the result analysis: (1) system performance and (2) user performance. By combining those two dimensions, we defined four hypotheses.

We tried to prove them from various perspectives. The following sections discuss each perspective one by one. For an overview, Table 46 summarizes the hypotheses and the measures used for proving each of them.

7.1 ADAPTIVE VISUALIZATION VERSUS TEXT-BASED PERSONALIZED SEARCH

7.1.1 Higher Precision of Search Results in the System Level

In the first hypothesis, we compared TaskSieve (Condition 1) and Adaptive VIBE (Condition 2) as the baseline and the experimental system. Each represents the traditional text-based personalized search and the adaptive visualization method. First of all, we assumed that Adaptive VIBE would be able to generate the personalized ranked lists with higher precision.

The analysis was consistent with our assumption. The marquee tool that was designed to help users to select documents spatially showed the almost equivalent precision level

Table 46: Comparison of hypothesis and the measures

	System Performance	User Performance
TaskSieve vs. Adaptive VIBE	**H1-1** System Precision Visual Cluster Quality	**H1-2** Open Diversity Navigation Depth Note Diversity Productivity (more high quality notes)
Adaptive VIBE vs. Adaptive VIBE + NE	**H2-1** Visual Cluster Quality	**H2-2** Note Precision

compared with ranked lists.

The top 10 documents retrieved by Adaptive VIBE had better quality than the baseline. $NDCG_{10}$ was used for measuring the quality of the retrieved documents and Adaptive VIBE generated higher NDCG top 10 documents than the baseline. NDCG gives more scores to the high ranked relevant documents. Therefore, the higher NDCG score within top 10 ranks means that Adaptive VIBE promoted relevant documents even more within the higher rank area.

The source of this performance improvement of Adaptive VIBE could be the better user models. The baseline search ability of Adaptive VIBE was identical to the baseline TaskSieve personalized search system and the key component that retrieved and ranked the documents was the combination of the query and the user model. It means that within the cycle of personalized searching and user interaction using the adaptive visualization, the user model was built better than the baseline system.

7.1.2 User Model Effects of Adaptive VIBE

The adaptive visualization generated by Adaptive VIBE are the direct interface that the users see and manipulate during their search process. It is the second component of the

Adaptive VIBE system on top of the personalized search results. We need to examine the quality of the visualizations as the second stage of system performance analysis.

Even though we did not include this effect in the hypothesis definitions, we have conducted pilot studies (Chapter 4) and expected the result of the pilot studies would duplicate in the user study too. During the pilot experiments, we found two effects:

1. Adaptive VIBE visualizations could separate the relevant and non-relevant documents visually.
2. The relevant document clusters gathered closer to the user model in the visualization.

Because these results were induced by the experiments using the log data of the TaskSieve personalized search study, we needed to check if the same effects would be observed during the user study. If we could confirm it, we would be able to guarantee the quality of the visualizations too.

The visualizations generated during the user study were analyzed using a clustering validity score and the location of the clusters were examined. The same separation of relevant and non-relevant document clusters were observed and the relevant document clusters tended to gather closer to the user models in the Adaptive VIBE visualization.

7.1.3 Helping Users to Locate Relevant Documents with the Visual User Model

On top of the result of the pilot study – relevant documents gathering near the user model – we formulated a scenario for users to pick up those relevant information, before the user study. We provided a marquee tool that could spatially select a group of documents and let them know the pilot study result as the background information to use the marquee tool for examining the documents around the user model.

Even though the tool was unfamiliar to the participants and there was greater risk to including noises during the selecting behavior, the marquee selection tool showed almost the same precision level with the traditional ranked lists.

We also observed the positions of the top 10 documents in the visualizations and found that those documents were moving towards the user models as the user-system interactions were accumulating and as the user models became more enriched.

Because the high rank documents were visible to the users (they were annotated in higher intensity colors), the participants were able to be naturally guided to the area closer to the user model, where the relevant information was more abundant. Combined with the fact that these high rank documents were very probable to be relevant, this characteristic could form a basis of the better user performance found in the next stage.

Of course, the visibility of the top-10 documents could have biased the subjects regardless of their proximity to the user model. That is, the subjects could have paid more attention to the top-10 documents regardless of the fact that they were closer to the user model or not. A complementary study that randomizes the location of the highlighted documents in the visualization and examines the user reaction will be able to answer this question.

7.1.4 User Performance – Diversity and Productivity beyond Over-fitting

The discussion in the previous section focused more on the system's role to prepare a good set of information. That is, the good system serves the requested information that the users can immediately consume. The good text-based system would place a number of documents on the very top of the ranked lists. The Adaptive VIBE system working perfectly as our scenario would place the relevant documents just beside the user models.

However, users' information need is not that simple. Particularly, the tasks required to solve in this user study was designed to navigate through complicated paths of exploration. Those who try to solve these tasks usually try different queries, study initial results returned from the system, learn more about the problem, try another hint against the system, etc. They may need to solve a question in order to solve the next when the structure of the task is multi-faceted.

Therefore, any system that is optimized too much for a specific task or a sub-task may occur the "over-fitting" problem. This is one of the problems that the personalized information access systems can have. We have already experienced before, when we provided users with an editable open user model [4]. There, the user model over-fitted by the user intervention and eventually deteriorated the system performance.

A personalized information retrieval system with an over-fitted user model can bring up a small number of highly precise results in the high rank. Nevertheless, there is always a risk to miss valuable information too.

Two variables were investigated in order to examine the perspectives that the simple precisions cannot cover: *Diversity* and *Productivity*.

7.1.4.1 Diversity Diversity was defined as how much the relevant information for the task was discovered by the participants. We first checked the depth of navigation in the ranked lists. The results clearly showed that the participants navigated much more low rank documents – *that were* ***relevant*** – than the text-based system. This result contrasts with the findings in [63], which found that the search engine users were easily biased toward the top of the ranked lists.

The second measure used for checking the diversity was the discovered sub-topic counts. A statistical topic detection algorithm was applied to the test corpus and we found that the visualization systems could discover more sub-topics in the annotation level as well as the document access level than the baseline.

One of our design principles of Adaptive VIBE was to employ the exploratory searching scheme. We wanted a system that could encourage its users explore the search space and learn within the process, rather than a simple look-up search system. The ability to discover diverse documents, annotations, and topics using Adaptive VIBE was an evidence that our expectation was met.

7.1.4.2 Productivity Even though the note precision of Adaptive VIBE was not found to be better than that of the baseline, we were able to find that Adaptive VIBE could make more high precision annotations.

In terms of the annotations that are the final products of the task-solving, the productivity can better reflect the reality. That is, how *many* high *quality* information was collected during the given time. It is like a combination of quality and quantity of the user annotations.

We found that the users of Adaptive VIBE could make more precise annotations than the baseline. Along with the diversity, the higher productivity of Adaptive VIBE could tell us that it was able to help users better to interact and work together with the system in order to achieve their goals.

7.2 KEYWORD-BASED VERSUS NE-BASED USER MODELING

The second hypothesis compared the keyword and the named-entity (NE) based user models of Adaptive VIBE. In the second pilot study conducted using the log data of TaskSieve, we found that the mixture of the keywords and the NEs within the user models were able to better separate the relevant and the non-relevant document clusters than the keyword or NE only user models.

Therefore, we constructed the user models mixing the keywords and the NEs (Condition 3) and compared their performances with the keyword only user models (Condition 2) in the user study. The same criteria for the result analysis was applied as in the first hypothesis testing: (1) system performance analysis and (2) user performance analysis.

7.2.1 System Performance Analysis

In the ranked list retrieval stage, we could not find any difference between the keyword and the NEs. However, the quality of the visualizations showed a marginal difference between two models. In one of the three topics given to the participants, the NE-based user model could construct better visualizations in terms of the separation of the relevant and non-relevant document clusters.

7.2.2 User Performance Analysis

We could not find any difference in terms of the users' document access to open the fulltexts, diversity and productivity of user annotations. However, we found that the users were able to make more precise annotations compared to the keyword-based user models or text-based personalized search system.

Given the almost equivalent level of diversity and productivity of user annotations, this improvement is worth noting. When we compared Adaptive VIBE and the text-based TaskSieve, we found the improvements were mostly found in the diversity or the productivity level, especially in the user performance. However, the NE-based user models contributed to the opposite dimension, the user annotation precision.

This difference could be originated from the difference of the two approaches, the adaptive visualization and the concept-based user modeling. Adaptive visualization was able to provide users with the ability to explore and discover the search space visually, which led to the higher diversity and productivity. NE-based user models were able to assist users to produce more accurate annotations, which lead to the higher precision notes.

Therefore, we can conclude that the second hypothesis was confirmed marginally, in that it could promote the precision of the system and the user performances.

8.0 CONCLUSION

This dissertation presented an adaptive visualization method called Adaptive VIBE and tested its effectiveness and analyzed the characteristics through a user study. Adaptive VIBE was also extended to include named-entity based user models in order to better present the semantics and concepts. The extension was compared with the keyword-based user models too.

Before the user study, we conducted two pilot experiments that attempted to construct the Adaptive VIBE visualizations with and without the NE-based user models. The experimental results showed that the Adaptive VIBE could better separate the relevant documents from non-relevant documents in the visualization. Moreover, the relevant information was more probable to be found near the user models, which were visually shown to the users. The NE-based user models could improve the quality of the visualizations when they were mixed with keywords in the user models.

8.1 MAIN FINDINGS

The result of the user study was presented in two stages: (1) system performance and (2) user performance. According to the analysis, Adaptive VIBE was able to construct higher precision document sets compared to the baseline text-based personalized search system. The results of the pilot study was replicated during the user study. That is, the visual separation of relevant information was observed again.

Based on the higher system performance, users of Adaptive VIBE could produce better results. They were able to access more diverse information. They discovered hidden documents in the lower rank of the retrieval lists, explored more sub-topics, and produced more diverse final reports compared to the baseline.

The effect of the NE-based user modeling appeared differently. With the almost same level of system performance and diversity, the users were able to produce the final results that were more precise than the keyword-based Adaptive VIBE.

This difference between Adaptive VIBE and Adaptive VIBE plus NE-based user models suggests that they can contribute to the different areas of advanced information access – diversity of information exploration and higher precision of information discovery.

Despite the positive results toward the Adaptive VIBE systems, we should note that some users showed negative reactions too. That was due to the unfamiliarity and the complexity of the adaptive visualization systems. Even though we imposed a good lengths of training, a single day experience in the new system was not enough to change their subjective impressions. We make it clear that Adaptive VIBE is not intended to replace the more familiar ranked lists. Rather, it is a complementary tool that can remedy some shortcomings of the ranked list based systems.

8.2 LIMITATIONS AND DELIMITATIONS

This study presented an adaptive visualization approach, Adaptive VIBE, and the results of the evaluations. However, the visual information exploration tool incorporated in Adaptive VIBE here cannot be stated as an easy-to-use tool for information retrieval. Rather, it was designed to assist users for in-depth exploration of the data during the retrieval process.

Therefore, it is not correct to state that the result of this study is applicable to the general audiences who are everyday Web search engine users. More specifically, the audiences who can benefit from the information exploration idea can be characterized as similar to information analysts. They are ready to invest time and effort to explore the massive pool of information, beyond the simple search attempts including just one or two queries in very short search iterations.

Two questions can be raised when comparing the participants of this study with real information analysts: (1) Is their knowledge enough in order to perform the task? (2) Do they have enough experience using the information access system? In order to address the first issue, we provided enough background information for the tasks used in the experimental sessions. For addressing the second issue, we provided enough time and effort for training the subjects. This process did not only include the how-to-use issue of the system, but also included real task-solving experience by using a training task exactly equivalent with the main tasks.

In terms of the text corpus, this study made use of an artificial news collection (TDT4). Therefore, the system was more optimized for the news texts than other type of documents. This can also limit the direct applicability of the result to more general and broader scope of information because different documents compared to news texts can have different properties. For example, finding out a simple and short Web page that contains a clear fragment of information will not required the exploratory analysis describe above. Rather, it will be recommended to use the traditional look up search instead.

The choice of NEs as the semantic representation for user models could be justified because it was easy to extract good NEs from the news corpus and the extracted NEs are appropriate for representing the concepts of the news stories. News stories naturally contain a lot of elements such as name, time, and events. They play significant roles in the news articles and could be effectively discovered by the NE annotator. However, we have no evidence that this performance can be generalized to other types of texts. Therefore, the potential of NEs as semantic elements should not be over-generalized beyond the news articles or the texts with similar characteristics.

8.3 FUTURE WORK

From the subjective feedbacks of the participants (Section 6.1.2), we found a great deviation of user preference toward the visualization-based systems. That is, it depended on each individual whether they liked the visualization-based information exploration or not. We tried to find out if there was any demographic or cognitive factor that affects this difference as Koshman [66], who found that the VIBE visualization was more appropriate for smart or expert users than novices. However, we could not find any clear evidence that determines individual differences in this user study. It may be because we had limited number of participants and tried to impose the participants the role of information analysts.

We expect that we will be able to find out the individual factors and learning effects in a long-term multi-session study such as MILCs (Multi-dimensional In-depth Long-term Case studies) [113]. A more generalized Adaptive VIBE version that is attached to a more accessible search environment could be used for this purpose.

Also, as discussed in Section 7.1.3, we found that the top-10 documents were moving towards the user model in the visualizations and the movements could have acted as a guidance for subjects to locate relevant documents. However, this feature also has the weakness to bias the users to the top-10 documents regardless of their spatial proximity to the user model. Therefore, we can plan a future study that randomizes the location of the top-10 documents and observes the user reactions in order to learn the exact factor that motivates the choice of relevant documents.

Another plan is to extend the concept extraction method beyond NEs. We need to explore various concept extraction techniques relevant to different document types or domains, such as noun phrase extraction [119], part-of-speech tagging [81], or probability-based N-gram extraction methods [71, 86].

APPENDIX A

ENTRY QUESTIONNAIRE

Searcher #___________

UNIVERSITY OF PITTSBURGH
PERSONALIZED INFORMATION ACCESS STUDY
ENTRY QUESTIONNAIRE

1. Your program and University: _____________________________________

2. Gender: ___ Male ___Female

3. Age: ___________________

4. First language:_________________

5. Highest degree level attained:
   ___Bachelor's ___Master's ___PhD ____Post-Doctorate

6. Have you taken any information retrieval class?

   ___ Yes ___ No ___Don't know

7. On average, the amount of time spent per day using a computer
   ___Less than 1 hr.
   ___1 hr – less than 2 hrs.
   ___2 hrs. – less than 3 hrs.
   ___3 hrs. – less than 4 hrs.
   ___More than 4 hrs.

8. Which Web browser do you use most often?
   ___Apple Safari
   ___Internet Explorer
   ___Mozilla Firefox
   ___Netscape
   ___Opera
   ___Other __________________

9. Which of the following search engines have you used before?
   ___AltaVista
   ___Aliweb
   ___AlltheWeb
   ___Ask.com
   ___Baidu
   ___Dogpile
   ___Excite
   ___Google
   ___Infoseek
   ___Inktomi
   ___Lycos
   ___Monkey Look
   ___MSN Search
   ___Northern Light

___Quaero
___Teoma
___Trumalia
___Vivisimo
___WebCrawler
___Yahoo! Search
___Other_________________
___Don't know

10. How confident are you in your abilities to locate specific information using a search engine?

Not at all confident *Very Confident*
1 2 3 4 5 6 7 8 9 10

11. On average, how much time do you spend per day reading, listening, and watching news?
    ___Less than 1 hr.
    ___1 hr – less than 2 hrs.
    ___2 hrs. – less than 3 hrs.
    ___3 hrs. – less than 4 hrs.
    ___More than 4 hrs.

12. What news sources (news programs, online newspapers, print newspapers, or online news databases) do you use to learn information about news events?

APPENDIX B

POST QUESTIONNAIRES

Searcher #___________
System ___TaskSieve___
Topic # ___________

UNIVERSITY OF PITTSBURGH
STUDY OF WEB INFORMATION EXTRACTION
POST-SEARCH QUESTIONNAIRE

Please answer the following questions, as they relate to this specific topic.

	Not at all		**Somew hat**		**Extrem ely**
1. Were you familiar with this topic before the search?	1	2	3	4	5
2. Did the passages and their documents provide you sufficient information for your summary?	1	2	3	4	5
3. When choosing to view a full document, was it mostly because you found useful information in the passage?	1	2	3	4	5
4. Do you think the task model helped finding out relevant information than plain searching?	1	2	3	4	5
5. Do you think the mediation between the query and the user model helped solving the problem?	1	2	3	4	5
6. Overall, did you have a positive experience with this system?	1	2	3	4	5
7. Overall, what is the difficulty of the **topic**? (1-easy, 5-hard)	1	2	3	4	5
8. How confident are you that you found all information in the database?	1	2	3	4	5

Please list important names (person, location, organization, etc.) found during the task

Please write down any other comments. Thank you!

Searcher #___________
System ___VIBE___
Topic # ___________

UNIVERSITY OF PITTSBURGH
STUDY OF WEB INFORMATION EXTRACTION
POST-SEARCH QUESTIONNAIRE

Please answer the following questions, as they relate to this specific topic.

	Not at all		**Somewhat**		**Extremely**
1. Were you familiar with this topic before the search?	1	2	3	4	5
2. Did the passages and their documents provide you sufficient information for your summary?	1	2	3	4	5
3. When choosing to view a full document, was it mostly because you found useful information in the passage?	1	2	3	4	5
4. Were you confident in the **visualization**'s ability to find useful information on this topic?	1	2	3	4	5
5. Did you find the separation of **the query and the user model POI** helpful in finding useful information?	1	2	3	4	5
6. Did you find the support of **different POI location** presets useful?	1	2	3	4	5
7. Did you find the visual cue about the system's **relevance expectation** (size & color of document icons) useful?	1	2	3	4	5
8. Did you find the **POI clustering** useful to complete your task?	1	2	3	4	5
9. Did you find the visualization easy to use?	1	2	3	4	5
10. Overall, did you have a positive experience with this system?	1	2	3	4	5
11. Overall, what is the difficulty of the **topic**? (1-easy, 5-hard)	1	2	3	4	5
12. How confident are you that you found all information in the database?	1	2	3	4	5

Please list important names (person, location, organization, etc.) found during the task

Please write down any other comments. Thank you!

Searcher #___________
System ___NEVIBE___
Topic # ___________

UNIVERSITY OF PITTSBURGH
STUDY OF WEB INFORMATION EXTRACTION
POST-SEARCH QUESTIONNAIRE

Please answer the following questions, as they relate to this specific topic.

	Not at all		**Somewhat**		**Extremely**
1. Were you familiar with this topic before the search?	1	2	3	4	5
2. Did the passages and their documents provide you sufficient information for your summary?	1	2	3	4	5
3. When choosing to view a full document, was it mostly because you found useful information in the passage?	1	2	3	4	5
4. Were you confident in the **visualization**'s ability to find useful information on this topic?	1	2	3	4	5
5. Did you find the **named-entities** were useful to complete your task?	1	2	3	4	5
6. Did you find the **named-entities** were reasonably extracted?	1	2	3	4	5
7. Did you find the **combination** of the **visualization** and **named-entities** was useful to complete your task?	1	2	3	4	5
8. Overall, did you have a positive experience with this system?	1	2	3	4	5
9. Overall, what is the difficulty of the **topic**? (1-easy, 5-hard)	1	2	3	4	5
10. How confident are you that you found all information in the database?	1	2	3	4	5

Please list important names (person, location, organization, etc.) found during the task

Please write down any other comments. Thank you!

APPENDIX C

TASK ASSIGNMENTS

5 TDT Topic 40009 - "US Senate Proposes Easing Cuban Trade Embargo" (JZ)

Background

1. Washington, DC is in US.

Short description of the task:

US Senate proposes a bill to easy trade with Cuba and Clinton signed it later. The task is to find out supports and objections as well as consequences of this new legislation, and give recommendations to changes in related policies, if any.

From the documents, find snippets of text that contain answers to each of the following questions:

1. What is main content of the law?
2. What is the name of issuer of the law?
3. What are the names of the affected parties of the law?
4. Where is the law?
5. What are the previous related laws?
6. What are the supports and objections of the current legislation?
7. What is the possible future impact?

15 TDT Topic 40048 - “Train Fire in Austrian Tunnel” (PB)

Background

1. Kaprun is an alpine resort in Austria
2. Austria is a country in Europe

Short description of the task

After a fire in a cable car traveling through a tunnel near Austrian village of Karpun, officials are working on identifying victims and salvaging the bodies. The task is to identify the number of US victims and survivors, determine and dispatch the transportation means to bring the rescued US citizens and the bodies to the nearest US Hospital/base, determine the person in charge to approve the transportation.

From the documents, find snippets of text that contain answers to each of the following questions:

1. what was the number of victims, and that of survivors
2. what was the number of US victims, and that of US survivors
3. what was the location of the accident
4. the type of vehicle used by rescuers
5. who was the responsible party for approving transportation
6. when did the accident start?
7. what were the actions taken by rescuers?
8. who was the responsible party for rescue operation?

13 TDT Topic 40021 – "Earthquake in El Salvador" (PB)

Background:

1. El Salvador and Guatemala are two countries in western Central America.
2. El Salvador is bordered to the west by Guatemala
3. An earthquake measuring 7.6 on the Richter scale happened 65 miles off the coast of El Salvador in the North Pacific Ocean.

Short description of the task:

Rescuers dig for survivors and causalities are being counted in El Salvador and Guatemala after an earthquake. The task is to identify a site with largest causalities and plan a rescue/help operation at that site. Determine the kind and the amount of help required and requested (i.e., tents, rescue equipment) and identify authorities in charge to cooperate with.

From the documents, find snippets of text that contain answers to each of the following questions:

1. How many victims in the earthquake? How many injured in the earthquake? How many survivors in the earthquake?
2. How many bodies were recovered in the earthquake?
3. Where is the site with large causalities?
4. What is the population of site with large causalities?
5. When was the earthquake?
6. What kinds of action had been taken by the rescuers?
7. Who was the person in charge of rescue operation?
8. How about police authorities?
9. What kinds of equipment were required for rescue operation?
10. What was the type of disaster?
11. How about the international help?
12. What help had been requested?
13. How about death toll?

BIBLIOGRAPHY

[1] J. Ahn and P. Brusilovsky. Adaptive visualization of search results: Bringing user models to visual analytics. *Information Visualization*, 8(3):167–179, 2009.

[2] J. Ahn and P. Brusilovsky. Can concept-based user modeling improve adaptive visualization? In *18th International Conference on User Modeling, Adaptation, and Personalization (UMAP 2010)*, pages 4–15, 2010.

[3] J. Ahn, P. Brusilovsky, J. Grady, D. He, and R. Florian. Semantic annotation based exploratory search for information analysts. *Information Processing and Management*, 46(4):383–402, 7 2010.

[4] J. Ahn, P. Brusilovsky, J. Grady, D. He, and S. Y. Syn. Open user profiles for adaptive news systems: help or harm? In *WWW '07: Proceedings of the 16th international conference on World Wide Web*, pages 11–20, New York, NY, USA, 2007. ACM Press.

[5] J. Ahn, P. Brusilovsky, D. He, J. Grady, and Q. Li. Personalized web exploration with task models. In *WWW '08: Proceedings of the 17th international conference on World Wide Web*, pages 1–10, New York, NY, USA, 2008. ACM.

[6] J. Ahn, P. Brusilovsky, and S. Sosnovsky. Quizvibe: Accessing educational objects with adaptive relevance-based visualization. In T. Reeves and S. Yamashita, editors, *Proceedings of World Conference on E-Learning in Corporate, Government, Healthcare, and Higher Education 2006*, pages 2707–2714, Honolulu, Hawaii, USA, October 2006. AACE.

[7] R. Arezki, P. Poncelet, G. Dray, and D. Pearson. Web information retrieval based on user profile. In *Adaptive Hypermedia and Adaptive Web-Based Systems*, pages 275–278. Springer, 2004.

[8] F. Bakalov, B. König-Ries, A. Nauerz, and M. Welsch. Introspectiveviews: An interface for scrutinizing semantic user models. *18th International Conference on User Modeling, Adaptation, and Personalization (UMAP 2010)*, pages 219–230, 2010.

[9] D. Bawden and L. Robinson. The dark side of information: overload, anxiety and other paradoxes and pathologies. *Journal of Information Science*, 35(2):180–191, April 2009.

[10] N. J. Belkin and W. B. Croft. Information filtering and information retrieval: two sides of the same coin? *Communications of ACM*, 35(12):29–38, December 1992.

[11] S. Benford, D. Snowdon, C. Greenhalgh, R. Ingram, I. Knox, and C. Brown. Vr-vibe: A virtual environment for co-operative information retrieval. *Computer Graphics Forum*, 14(3):349–360, 1995.

[12] E. Bier, E. Ishak, and E. Chi. Entity Workspace: an evidence file that aids memory, inference, and reading. *Intelligence and Security Informatics*, pages 466–472, 2006.

[13] D. Billsus and M. Pazzani. A hybrid user model for news story classification. In *Proceedings of the Seventh International Conference on User Modeling (UM'99)*, pages 99–108, 1999.

[14] E. Bitton. A spatial model for collaborative filtering of comments in an online discussion forum. In *Proceedings of the third ACM conference on Recommender systems*, pages 393–396. ACM, 2009.

[15] D. Blei, A. Ng, and M. Jordan. Latent dirichlet allocation. *The Journal of Machine Learning Research*, 3:993–1022, 2003.

[16] M. Braun, K. Dellschaft, T. Franz, D. Hering, P. Jungen, H. Metzler, E. Müller, A. Rostilov, and C. Saathoff. Personalized search and exploration with mytag. In *Proceeding of the 17th international conference on World Wide Web*, pages 1031–1032. ACM, 2008.

[17] P. Brusilovsky, J. Ahn, T. Dumitriu, and M. Yudelson. Adaptive knowledge-based visualization for accessing educational examples. In *Tenth International Conference on Information Visualisation (IV'06)*, pages 142 – 150, Location, England, July 2006.

[18] P. Brusilovsky and E. Millán. The adaptive web. chapter User models for adaptive hypermedia and adaptive educational systems, pages 3–53. Springer-Verlag, Berlin, Heidelberg, 2007.

[19] P. Brusilovsky and R. Rizzo. Map-based horizontal navigation in educational hypertext. In *Proceedings of the thirteenth ACM conference on Hypertext and hypermedia*, volume 3 of *1*, page 10. ACM, 2002.

[20] P. Brusilovsky and R. Rizzo. Using maps and landmarks for navigation between closed and open corpus hyperspace in web-based education. *New Review of Hypermedia and Multimedia*, 8(1):59–82, 2002.

[21] A. Bunt, G. Carenini, and C. Conati. The adaptive web. chapter Adaptive content presentation for the web, pages 409–432. Springer-Verlag, Berlin, Heidelberg, 2007.

[22] J. Callan, W. Croft, and S. Harding. The INQUERY retrieval system. In *Proceedings of the third international conference on database and expert systems applications*, pages 78–83, 1992.

[23] G. Cao, J.-Y. Nie, and J. Bai. Integrating word relationships into language models. In *SIGIR '05: Proceedings of the 28th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 298–305, New York, NY, USA, 2005. ACM.

[24] R. G. Capra and G. Marchionini. The relation browser tool for faceted exploratory search. In *JCDL '08: Proceedings of the 8th ACM/IEEE-CS joint conference on Digital libraries*, pages 420–420, New York, NY, USA, 2008. ACM.

[25] S. Card, J. Mackinlay, and B. Shneiderman. *Readings in information visualization: using vision to think*. Morgan Kaufmann, 1999.

[26] J. Chaffee and S. Gauch. Personal ontologies for web navigation. In *Proceedings of the ninth international conference on Information and knowledge management*, pages 227–234. ACM, 2000.

[27] M. Chalmers and P. Chitson. Bead: explorations in information visualization. In *SIGIR '92: Proceedings of the 15th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 330–337, New York, NY, USA, 1992. ACM.

[28] C. C. Chen, M. C. Chen, and Y. Sun. Pva: a self-adaptive personal view agent system. In *KDD '01: Proceedings of the seventh ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 257–262, New York, NY, USA, 2001. ACM.

[29] H. Chen, A. L. Houston, R. R. Sewell, and B. R. Schatz. Internet browsing and searching: User evaluations of category map and concept space techniques. *Journal of the American Society for Information Science*, 49(7):582–603, 1998.

[30] P. A. Chirita, C. S. Firan, and W. Nejdl. Personalized query expansion for the web. In *SIGIR '07: Proceedings of the 30th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 7–14, New York, NY, USA, 2007. ACM.

[31] M. Christel and D. Martin. Information visualization within a digital video library. *Journal of Intelligent Information Systems*, 11(3):235–257, 1998.

[32] C. Cleverdon, J. Mills, and M. Keen. Factors determining the performance of indexing systems. 1966.

[33] J. Cohen. *Statistical Power Analysis for the Behavioral Sciences*. Routledge Academic, 2 edition, January 1988.

[34] W. Croft and D. Harper. Using probabilistic models of document retrieval without relevance information. *Journal of Documentation*, 35(4):285–295, 1979.

[35] M. Daoud, L. Tamine, and M. Boughanem. A Personalized Graph-Based Document Ranking Model Using a Semantic User Profile. *18th International Conference on User Modeling, Adaptation, and Personalization (UMAP 2010)*, pages 171–182, 2010.

[36] D. L. Davies and D. W. Bouldin. A cluster separation measure. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PAMI(2):224–227, January 2009.

[37] A. Don, E. Zheleva, M. Gregory, S. Tarkan, L. Auvil, T. Clement, B. Shneiderman, and C. Plaisant. Discovering interesting usage patterns in text collections: integrating text mining with visualization. In *CIKM '07: Proceedings of the sixteenth ACM conference on Conference on information and knowledge management*, pages 213–222, New York, NY, USA, 2007. ACM.

[38] D. Dubin. *Structure in document browsing spaces*. PhD thesis, University of Pittsburgh, 1996.

[39] S. Dumais. Latent semantic analysis. *Annual Review of Information Science and Technology (ARIST)*, 38:189–230, 2004.

[40] P. Eades. A heuristic for graph drawing. *Congressus Numerantium*, 42:149–160, 1984.

[41] R. Florian, H. Hassan, A. Ittycheriah, H. Jing, N. Kambhatla, X. Luo, N. Nicolov, S. Roukos, and T. Zhang. A statistical model for multilingual entity detection and tracking, 2004.

[42] E. Gansner, Y. Hu, S. Kobourov, and C. Volinsky. Putting recommendations on the map: visualizing clusters and relations. In *Proceedings of the third ACM conference on Recommender systems*, pages 345–348. ACM, 2009.

[43] J. Gao, J.-Y. Nie, G. Wu, and G. Cao. Dependence language model for information retrieval. In *SIGIR '04: Proceedings of the 27th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 170–177, New York, NY, USA, 2004. ACM Press.

[44] S. Gauch, M. Speretta, A. Chandramouli, and A. Micarelli. The adaptive web. chapter User profiles for personalized information access, pages 54–89. Springer-Verlag, Berlin, Heidelberg, 2007.

[45] G. Gentili, A. Micarelli, and F. Sciarrone. Infoweb: An adaptive information filtering system for the cultural heritage domain. *Applied Artificial Intelligence*, pages 715–744, 2003.

[46] D. Gotz, M. Zhou, and V. Aggarwal. Interactive visual synthesis of analytic knowledge. In *IEEE Symposium on Visual Analytics Science and Technology (VAST 2006)*, pages 51–58, Baltimore, MD, USA, 2006. IEEE Computer Society.

[47] B. Grawemeyer. Evaluation of erst–an external representation selection tutor. *Diagrammatic Representation and Inference*, pages 154–167, 2006.

[48] B. Grawemeyer and R. Cox. A bayesian approach to modelling users'information display preferences. *User Modeling 2005*, pages 225–230, 2005.

[49] B. Grawemeyer and R. Cox. Graphical data displays and database queries: Helping users select the right display for the task. In *Smart Graphics*, pages 53–64. Springer, 2005.

[50] D. He, P. Brusilovsky, J. Ahn, J. Grady, R. Farzan, Y. Peng, Y. Yang, and M. Rogati. An evaluation of adaptive filtering in the context of realistic task-based information exploration. *Information Processing & Management*, 44(2):511–533, March 2008.

[51] M. A. Hearst. Tilebars: visualization of term distribution information in full text information access. In *CHI '95: Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 59–66, New York, NY, USA, 1995. ACM Press/Addison-Wesley Publishing Co.

[52] M. A. Hearst. Clustering versus faceted categories for information exploration. *Communications of ACM*, 49(4):59–61, April 2006.

[53] M. A. Hearst and J. O. Pedersen. Reexamining the cluster hypothesis: scatter/gather on retrieval results. In *SIGIR '96: Proceedings of the 19th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 76–84, New York, NY, USA, 1996. ACM Press.

[54] M. Hemmje, C. Kunkel, and A. Willett. Lyberworld - a visualization user interface supporting fulltext retrieval. In *SIGIR '94: Proceedings of the 17th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 249–259, New York, NY, USA, 1994. Springer-Verlag New York, Inc.

[55] T. Hofmann. Probabilistic latent semantic indexing. In *SIGIR '99: Proceedings of the 22nd annual international ACM SIGIR conference on Research and development in information retrieval*, pages 50–57, New York, NY, USA, 1999. ACM.

[56] E. Ide. New experiments in relevance feedback. *The SMART retrieval system: experiments in automatic document processing*, pages 337–354, 1971.

[57] B. J. Jansen, A. Spink, and T. Saracevic. Real life, real users, and real needs: a study and analysis of user queries on the web. *Information Processing & Management*, 36(2):207–227, March 2000.

[58] K. Järvelin and J. Kekäläinen. Cumulated gain-based evaluation of ir techniques. *ACM Transactions on Information Systems (TOIS)*, 20(4):422–446, 2002.

[59] B. Johnson and B. Shneiderman. Tree-maps: a space-filling approach to the visualization of hierarchical information structures. In *VIS '91: Proceedings of the 2nd conference on Visualization '91*, pages 284–291, Los Alamitos, CA, USA, 1991. IEEE Computer Society Press.

[60] S. Jokela, M. Turpeinen, T. Kurki, E. Savia, and R. Sulonen. The role of structured content in a personalized news service. In *HICSS '01: Proceedings of the 34th Annual Hawaii International Conference on System Sciences (HICSS-34)-Volume 7*, page 7044, Washington, DC, USA, 2001. IEEE Computer Society.

[61] S. Jung, K. Harris, J. Webster, and J. L. Herlocker. Serf: integrating human recommendations with search. In *CIKM '04: Proceedings of the thirteenth ACM international conference on Information and knowledge management*, pages 571–580, New York, NY, USA, 2004. ACM Press.

[62] S. Kaski, T. Honkela, K. Lagus, and T. Kohonen. Websom - self-organizing maps of document collections. *Neurocomputing*, 21(1-3):101–117, November 1998.

[63] M. T. Keane, M. O'Brien, and B. Smyth. Are people biased in their use of search engines? *Communications of ACM*, 51(2):49–52, 2008.

[64] D. Kelly and J. Teevan. Implicit feedback for inferring user preference: a bibliography. *SIGIR Forum*, 37(2):18–28, 2003.

[65] P. Korpipää and J. Remes. Website navigation and profiling based on self-organising maps. In *Proceedings of the IJCAI'03 Workshop on Artificial Intelligence, Information Access, and Mobile Computing*, pages 59–64, August 2003.

[66] S. Koshman. Testing user interaction with a prototype visualization-based information retrieval system. *Journal of the American Society for Information Science and Technology*, 56(8):824–833, 2005.

[67] R. Krovetz. Viewing morphology as an inference process. In *Proceedings of the 16th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 191–202. ACM, 1993.

[68] G. Kumaran and J. Allan. Text classification and named entities for new event detection. In *SIGIR '04: Proceedings of the 27th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 297–304, New York, NY, USA, 2004. ACM.

[69] B. Kump, C. Seifert, G. Beham, S. Lindstaedt, and T. Ley. MyExperiences: Visualizing Evidence in an Open Learner Model. In *Adjunct Proceedings of the 18th International Conference on User Modeling, Adaptation, and Personalization (Posters and Demonstrations)*, pages 16–18, 2010.

[70] W. Lai, X. Huang, Q. Nguyen, and M. Huang. Applying graph layout techniques to web information visualization and navigation. In *Computer Graphics, Imaging and Visualisation, 2007. CGIV'07*, pages 447–453, 2007.

[71] V. Lavrenko and W. B. Croft. Relevance based language models. In *SIGIR '01: Proceedings of the 24th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 120–127, New York, NY, USA, 2001. ACM.

[72] S. Lehmann, U. Schwanecke, and R. Dörner. Interactive visualization for opportunistic exploration of large document collections. *Information Systems*, 35(2):260–269, 2010.

[73] A. Leuski and J. Allan. Improving interactive retrieval by combining ranked lists and clustering. In *Proceedings of RIAO*, volume 2000, pages 665–681, 2000.

[74] A. Leuski and J. Allan. Lighthouse: showing the way to relevant information. In *Information Visualization, 2000. InfoVis 2000. IEEE Symposium on*, pages 125–129, 2000.

[75] A. Leuski and J. Allan. Interactive information retrieval using clustering and spatial proximity. *User Modeling and User-Adapted Interaction*, 14(2):259–288, 2004.

[76] X. Lin, D. Soergel, and G. Marchionini. A self-organizing semantic map for information retrieval. In *SIGIR '91: Proceedings of the 14th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 262–269, New York, NY, USA, 1991. ACM Press.

[77] F. Liu, C. Yu, and W. Meng. Personalized web search for improving retrieval effectiveness. *IEEE Transactions on knowledge and data engineering*, 16(1):28–40, 2004.

[78] H. Luhn. Kwic—keyword-in-context index for technical literature. *American Documentation*, 13(4):359–366, 1960.

[79] B. Magnini and C. Strapparava. User modelling for news web sites with word sense based techniques. *User Modeling and User-Adapted Interaction*, 14(2):239–257, June 2004.

[80] T. Mandl and C. W. Hacker. The effect of named entities on effectiveness in cross-language information retrieval evaluation. In *SAC '05: Proceedings of the 2005 ACM symposium on Applied computing*, pages 1059–1064, New York, NY, USA, 2005. ACM.

[81] C. Manning and H. Schütze. *Foundations of statistical natural language processing*. MIT Press, 1999.

[82] C. D. Manning, P. Raghavan, and H. Schütze. *Introduction to Information Retrieval*. Cambridge University Press, New York, 2008.

[83] G. Marchionini. Exploratory search: from finding to understanding. *Communications of ACM*, 49(4):41–46, April 2006.

[84] G. Marchionini, B. M. Wildemuth, and G. Geisler. The Open Video Digital Library: A Mobius strip of research and practice. *Journal of the American Society for Information Science and Technology*, 57(12):1629–1643, 2006.

[85] J. F. McCarthy, K. A. Marx, P. E. Hoffman, A. G. Gee, P. O'Neil, M. L. Ujwal, and J. Hotchkiss. Applications of machine learning and high-dimensional visualization in cancer detection, diagnosis, and management. *Annals of the New York Academy of Sciences*, 1020:239–262, May 2004.

[86] D. Metzler and W. Croft. Combining the language model and inference network approaches to retrieval. *Information processing & management*, 40(5):735–750, 2004.

[87] D. Metzler Jr. *Beyond bags of words: effectively modeling dependence and features in information retrieval*. PhD thesis, University of Massachusetts Amherst, 2007.

[88] A. Micarelli, F. Gasparetti, F. Sciarrone, and S. Gauch. The adaptive web. chapter Personalized search on the world wide web, pages 195–230. Springer-Verlag, Berlin, Heidelberg, 2007.

[89] R. Mihalcea and D. L. Moldovan. Document indexing using named entities. *Studies in Informatics and Control*, 10(1):21–28, 2001.

[90] H. Njike, T. Artieres, P. Gallinari, J. Blanchard, and G. Letellier. Automatically building domain model in hypermedia applications. 2005.

[91] K. A. Olsen, R. R. Korfhage, K. M. Sochats, M. B. Spring, and J. G. Williams. Visualization of a document collection: the vibe system. *Information Processing & Management*, 29(1):69–81, 1993.

[92] T. O'Reilly. What is web 2.0: Design patterns and business models for the next generation of software, 2005.

[93] D. Petkova and B. W. Croft. Proximity-based document representation for named entity retrieval. In *CIKM '07: Proceedings of the sixteenth ACM conference on Conference on information and knowledge management*, pages 731–740, New York, NY, USA, 2007. ACM.

[94] P. Pirolli and S. Card. The sensemaking process and leverage points for analyst technology as identified through cognitive task analysis. In *Proceedings of International Conference on Intelligence Analysis*, pages 2–4, 2005.

[95] J. Pitkow, H. Schütze, T. Cass, R. Cooley, D. Turnbull, A. Edmonds, E. Adar, and T. Breuel. Personalized search. *Communications of ACM*, 45(9):50–55, 2002.

[96] J. M. Ponte and W. B. Croft. A language modeling approach to information retrieval. In *SIGIR '98: Proceedings of the 21st annual international ACM SIGIR conference on Research and development in information retrieval*, pages 275–281, New York, NY, USA, 1998. ACM.

[97] M. Porter. An algorithm for suffix stripping. *Program*, 14(3):130–137, 1980.

[98] A. Pretschner and S. Gauch. Ontology based personalized search. In *Tools with Artificial Intelligence, 1999. Proceedings. 11th IEEE International Conference on*, pages 391–398. Published by the IEEE Computer Society, 1999.

[99] W. Rivadeneira and B. B. Bederson. A study of search result clustering interfaces: Comparing textual and zoomable user interfaces. Technical report, University of Maryland HCIL Technical Report HCIL-2003-36, 2003.

[100] S. E. Robertson. The probability ranking principle in IR. *Journal of documentation*, 33(4):294–304, 1993.

[101] S. E. Robertson and K. S. Jones. Relevance weighting of search terms. *Journal of the American Society for information Science*, 27(3):129–146, 1976.

[102] J. J. Rocchio. Relevance feedback in information retrieval. [106], pages 313–323.

[103] D. Roussinov and M. Ramsey. Information forage through adaptive visualization. In *DL '98: Proceedings of the third ACM conference on Digital libraries*, pages 303–304, New York, NY, USA, 1998. ACM.

[104] I. Ruthven and M. Lalmas. A survey on the use of relevance feedback for information access systems. *The Knowledge Engineering Review*, 18(02):95–145, June 2003.

[105] G. Salton. *Automatic Text Processing: The Transformation, Analysis, and Retrieval of Information by Computer*. Addison Wesley, Reading, MA, 1989.

[106] G. Salton et al. *The SMART retrieval system*. Prentice-Hall Englewood Cliffs, NJ, 1971.

[107] S. Sekine. Named entity: History and future. *Project notes, New York University*, 2004.

[108] G. Semeraro, M. Degemmis, P. Lops, and I. Palmisano. WordNet-based user profiles for semantic personalization. In *PIA 2005 Workshop on New Technologies for Personalized Information Access*, pages 74–83, 2005.

[109] X. Shen, B. Tan, and C. Zhai. Implicit user modeling for personalized search. In *CIKM '05: Proceedings of the 14th ACM international conference on Information and knowledge management*, pages 824–831, New York, NY, USA, 2005. ACM.

[110] B. Shneiderman. The eyes have it: A task by data type taxonomy for information visualizations. *Visual Languages, IEEE Symposium on*, 0:336–343, 1996.

[111] B. Shneiderman. Supporting creativity with advanced information-abundant user interfaces. *Frontiers of human-centered computing, online communities and virtual environments*, pages 469–480, 2001.

[112] B. Shneiderman, B. B. Bederson, and S. M. Drucker. Find that photo!: interface strategies to annotate, browse, and share. *Communications of ACM*, 49(4):69–71, April 2006.

[113] B. Shneiderman and C. Plaisant. Strategies for evaluating information visualization tools: multi-dimensional in-depth long-term case studies. In *BELIV '06: Proceedings of the 2006 AVI workshop on BEyond time and errors*, pages 1–7, New York, NY, USA, 2006. ACM.

[114] M. Speretta and S. Gauch. Personalized search based on user search histories. In *The 2005 IEEE/WIC/ACM International Conference on Web Intelligence, 2005. Proceedings*, pages 622–628, 2005.

[115] A. Spoerri. Infocrystal: a visual tool for information retrieval & management. In *CIKM '93: Proceedings of the second international conference on Information and knowledge management*, pages 11–20, New York, NY, USA, 1993. ACM.

[116] K. Sugiyama, K. Hatano, and M. Yoshikawa. Adaptive web search based on user profile constructed without any effort from users. In *WWW '04: Proceedings of the 13th international conference on World Wide Web*, pages 675–684, New York, NY, USA, 2004. ACM.

[117] J. Teevan, S. T. Dumais, and E. Horvitz. Personalizing search via automated analysis of interests and activities. In *SIGIR '05: Proceedings of the 28th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 449–456, New York, NY, USA, 2005. ACM.

[118] T. Teraoka and M. Maruyama. Adaptive information visualization based on the user's multiple viewpoints-interactive 3d visualization of the www. In *Proceedings of the 1997 IEEE Symposium on Information Visualization (InfoVis' 97)*, pages 25–28. IEEE Computer Society, 1997.

[119] K. Tolle and H. Chen. Comparing noun phrasing techniques for use with medical digital library tools. *Journal of the American Society for Information Science*, 51(4):352–370, 2000.

[120] C. J. Van Rijsbergen. *Information Retrieval, 2nd edition.* Dept. of Computer Science, University of Glasgow, 1979.

[121] J. Vassileva. User modeling and user-adapted interaction; a task-centered approach for user modeling in a hypermedia office documentation system. 6(2):185–223, 1996.

[122] E. M. Voorhees. Using wordnet to disambiguate word senses for text retrieval. In *SIGIR '93: Proceedings of the 16th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 171–180, New York, NY, USA, 1993. ACM.

[123] A. Wærn. User involvement in automatic filtering: An experimental study. *User Modeling and User-Adapted Interaction*, V14(2):201–237, June 2004.

[124] R. White, B. Kules, and B. Bederson. Exploratory search interfaces: Categorization, clustering and beyond. In *SIGIR FORUM*, volume 39, page 52, 2005.