A Visual Framework for Knowledge Discovery on the Web: An Empirical Study of Business Intelligence Exploration

WINGYAN CHUNG, HSINCHUN CHEN, AND
JAY F. NUNAMAKER JR.

WINGYAN CHUNG is an Assistant Professor of Computer Information Systems in the Department of Information and Decision Sciences at College of Business Administration of the University of Texas at El Paso. He received a Ph.D. in Management Information Systems from the University of Arizona, and an M.S. in Information and Technology Management and a B.S. in Business Administration from the Chinese University of Hong Kong. His research interests include knowledge management, Web analysis and mining, artificial intelligence, business intelligence applications, and human–computer interaction. His papers have appeared in *Communications of the ACM, IEEE Computer, International Journal of Human–Computer Studies, Decision Support Systems, Journal of the American Society for Information Science and Technology,* Hawaii International Conference on System Sciences, and ACM/IEEE-CS Joint Conference on Digital Libraries, among others. Prior to joining UTEP, he worked as a research associate in the Artificial Intelligence Lab at the University of Arizona, where he led several research projects and wrote research proposals leading to federal funding.

HSINCHUN CHEN is McClelland Professor of MIS at the Eller College of the University of Arizona, and Andersen Consulting Professor of the Year (1999). He received a Ph.D. in Information Systems from New York University in 1989, an MBA in Finance from SUNY–Buffalo in 1985, and a B.S. in Management Science from the National Chiao-Tung University in Taiwan. He is author of more than 120 articles covering medical informatics, homeland security, semantic retrieval, search algorithms, knowledge management, and Web computing in leading information technology publications. He serves on the editorial boards of *Journal of the American Society for Information Science and Technology, ACM Transactions on Information Systems,* and *Decision Support Systems.* He is Scientific Counselor of the Lister Hill Center of the National Library of Medicine. His work has been featured in various scientific and information technologies publications, as well as in the general press, including *Science, New York Times, Los Angeles Times, Newsweek, Business Week, Business 2.0, NCSA Access Magazine, WEBster,* and *HPCWire.*

JAY F. NUNAMAKER JR. is Regents and Soldwedel Professor of MIS, Computer Science and Communication, and Director of the Center for the Management of Information at the University of Arizona, Tucson. Dr. Nunamaker received the LEO Award from the Association of Information Systems at ICIS in Barcelona, Spain, December 2002. This award is given for a lifetime of exceptional achievement in information

Journal of Management Information Systems / Spring 2005, Vol. 21, No. 4, pp. 57–84.

0742–1222 / 2005 $9.50 + 0.00.

systems. He was elected as a fellow of the Association of Information Systems in 2000. Dr. Nunamaker has over 40 years of experience in examining, analyzing, designing, testing, evaluating, and developing information systems. He has served as a test engineer at the Shippingport Atomic Power facility, as a member of the ISDOS team at the University of Michigan, and as a member of the faculty at Purdue University, prior to joining the faculty at the University of Arizona in 1974. His research on group support systems addresses behavioral as well as engineering issues and focuses on theory as well as implementation. Dr. Nunamaker received his Ph.D. in systems engineering and operations research from Case Institute of Technology, an M.S. and B.S. in engineering from the University of Pittsburgh, and a B.S. from Carnegie Mellon University. He has been a licensed professional engineer since 1965.

ABSTRACT: Information overload often hinders knowledge discovery on the Web. Existing tools lack analysis and visualization capabilities. Search engine displays often overwhelm users with irrelevant information. This research proposes a visual framework for knowledge discovery on the Web. The framework incorporates Web mining, clustering, and visualization techniques to support effective exploration of knowledge. Two new browsing methods were developed and applied to the business intelligence domain: Web community uses a genetic algorithm to organize Web sites into a tree format; knowledge map uses a multidimensional scaling algorithm to place Web sites as points on a screen. Experimental results show that knowledge map outperformed Kartoo, a commercial search engine with graphical display, in terms of effectiveness and efficiency. Web community was found to be more effective, efficient, and usable than result list. Our visual framework thus helps to alleviate information overload on the Web and offers practical implications for search engine developers.

KEY WORDS AND PHRASES: business intelligence, genetic algorithm, knowledge map, multidimensional scaling, visualization, Web browsing, Web community.

A STUDY FOUND THAT THE WORLD PRODUCES between 635,000 and 2.12 million terabytes of unique information per year, most of which has been stored in computer hard drives or servers [25]. Many of these computing devices serve as the repository of the Internet, supporting convenient access of information, but also posing challenges of effective knowledge discovery from voluminous information. Such a problem is known as information overload, where users find it difficult to sift through a large amount of irrelevant information [3]. Since the Internet is one of the top five sources of business information [15], information overload on the Web is likely to hinder business analysis. In a typical analysis scenario, a business analyst in the database technology field might ask the following questions:

- Can we visualize the overall market situation of the database technology industry?
- What are the different subgroups inside the community of database technology companies?

- To which group in the entire competitive environment does our company belong?
- Which ten competitors in our field most resemble us?

Answers to these questions reveal *business intelligence,* which is defined as the result of "acquisition, interpretation, collation, assessment, and exploitation of information" [10, p. 313] in the business domain.

Web search engines are commonly used to locate information for business analysis. They typically produce lists of numerous results upon users' query. Although search engines may put a number of relevant results on top of a long textual list, many relevant results are still buried in the list. Analysts may not be able to summarize all the results visually or to find meaningful subgroups within the entire set of results.

In this paper, we propose a visual framework for knowledge discovery on the Web. The framework incorporates Web mining, clustering, and visualization techniques to support effective exploration of knowledge. A central thesis is whether the framework can enhance knowledge discovery on the Web effectively and efficiently, with particular application to the business intelligence domain. Based on the framework, we have developed *Business Intelligence Explorer* (BIE), a visualization system for browsing a large number of results related to business Web sites. We also have compared the system with existing browsing methods in terms of effectiveness and efficiency. Using the framework, we aimed to alleviate information overload on the Web and to facilitate understanding of a vast amount of information. Throughout the paper, we make a distinction among "data," "information," and "knowledge" [29]. We believe that new browsing methods are needed to facilitate exploration of knowledge on the Web.

Literature Review

BUSINESS PRACTITIONERS HAVE DEVELOPED automated tools to support better understanding and processing of information. In recent years, business intelligence tools have become important for analysis of information on the Web [14]. Researchers have also developed advanced analysis and visualization techniques to summarize and present vast amount of information. This section reviews related work in business intelligence tools and using visualization to facilitate browsing. We define "browsing" as an exploratory information-seeking process characterized by the absence of planning, with a view to forming a mental model of the content being browsed.

Business Intelligence Tools

Business intelligence tools enable organizations to understand their internal and external environments through the systematic acquisition, collation, analysis, interpretation, and exploitation of information. Fuld et al. [14] found that the global interest in intelligence technology has increased significantly over the past five years. They compared 16 business intelligence tools and found that these tools have become more open to the Web, through which businesses nowadays share information and perform

transactions. However, automated search capability in many tools can lead to information overload. Despite recent improvements in analysis capability [14], there is still a long way to go to assist qualitative analysis effectively. Most tools that claim to do analysis simply provide different views of collection of information (e.g., comparison between different products or companies). Due to limited analysis capability, these tools are weak at visually summarizing a large number of documents collected from the Web. Although search engines may help, their linear list display of numerous results may lead to information overload.

Approaches to Reducing Information Overload

Researchers have proposed frameworks and techniques to deal with information overload and a lack of analysis capability of search engines and business intelligence tools. Shneiderman proposed a task by data type taxonomy (TTT) to study the types of data and tasks involved in visual displays of textual information [34]. Traditional result-list display of hypertext belongs to the one-dimensional data type. Although still widely used in many Web search engines because of the convenience of presentation, result list allows only limited room for browsing (e.g., scrolling a long list of results). In contrast, data types such as two-dimensional data, tree data, and network data allow more browsing tasks to be done and support human visual capabilities more effectively. Four types of visual display format are identified in Lin [24]: hierarchical displays, network displays, scatter displays, and map displays. Among them, hierarchical (tree) displays were shown to be an effective information access tool, particularly for browsing [9]; scatter displays most faithfully reflected underlying structures of data among the first three displays; and map displays provided a view of the entire collection of items at a distance [24]. However, both hierarchical and map displays are not widely used in existing search engines. Other visualization techniques have been developed to enable better understanding of documents.

Document Visualization

Document visualization primarily concerns the task of getting insight into information obtained from one or more documents without users' having read those documents [38]. Most processes of document visualization involve three stages: *document analysis, algorithms,* and *visualization* [35]. We review below various techniques of these three stages.

Document Analysis

Web mining techniques have been applied to analysis of documents on the Web. Web content mining treats a Web page as a vector of weights of key terms [31]. Web structure mining treats the Web as a graph, where nodes (Web pages) are connected to each other through directed edges (hyperlinks). Researchers have tried to combine Web content mining and Web structure mining to improve the quality of analysis.

For example, using a similarity metric that incorporated textual information, hyperlink structure, and cocitation relations, He et al. [20] proposed an unsupervised clustering method that was shown to identify relevant topics effectively. The clustering method employed a graph-partitioning method based on a normalized cut criterion. Bharat and Henzinger [1] augmented a connectivity analysis-based algorithm with content analysis and achieved an improvement of precision by at least 45 percent over purely connectivity analysis. For the purpose of finding Web pages that form communities, the approach used by He et al. has the advantage of combining both Web content information and Web structure information for clustering.

In addition to the analysis of documents, meta-searching has been shown to be a highly effective method of resource discovery and collection on the Web. By sending queries to multiple search engines and collating the set of top-ranked results from each search engine, meta-search engines can greatly reduce bias in search results and improve the coverage. Chen et al. [8] showed that the approach of integrating meta-searching with textual clustering tools achieved high precision in searching the Web. Mowshowitz and Kawaguchi [26] concluded from their study that the only realistic way to counter the adverse effects of search engine bias is to perform meta-searching. In addition, several commercial meta-search engines allow the searching of various large search engines and provide added functionality. MetaCrawler (www.metacrawler .com) provides analysis of relevance rankings from source search engines and elimination of duplicates [32]. Vivisimo (www.vivisimo.com) automatically clusters the search results into different groups [30].

Algorithms

Algorithms have been used to cluster and project a high-dimensional structure onto a two- or three-dimensional space. For example, cluster algorithms and multidimensional scaling algorithms are frequently used in visualization.

Cluster Algorithms. Cluster algorithms classify objects into meaningful disjoint subsets or partitions. Two categories of cluster algorithms are used in previous research: hierarchical and partitional [18]. Hierarchical clustering is a procedure for transforming a proximity matrix into a sequence of nested partitions. Partitional clustering assigns objects into groups such that objects in a cluster are more similar to each other than to objects in different clusters. Typically, a clustering criterion is adopted to guide the search for optimal grouping. A graph-theoretic criterion, called normalized cut, treats clustering as graph partitioning and computes the normalized cost of cutting a graph. Using this criterion in image segmentation [33] and Web page clustering [20] has achieved good results. Although partitional clustering tries to achieve optimal results, it is usually difficult to evaluate all partitions because the number of possible partitions is extremely large. Therefore, heuristics are needed to find good values to the criterion selected. Examples of such heuristics include genetic algorithms, taboo search, scatter search, and simulated annealing [2]. Among these techniques, genetic algorithms generally performs well when the search space is very

large because of its global searching capability. Whereas other techniques also support searching and optimization in different domains, genetic algorithms have been successfully used in information retrieval to find the best document description [17] to guide the spidering of Web pages [7], to learn from user interests in Web searching [28], and to partition graphs [4]. Yet genetic algorithms have not been widely applied to the business intelligence domain.

Multidimensional Scaling. Multidimensional scaling (MDS) algorithms consist of a family of techniques that portray a data structure in a spatial fashion, where the coordinates of data points (x_{ia}) are calculated by a dimensionality reduction procedure [40]. The (unweighted) distances (d_{ij}) among entities can be calculated as follows [40]:

$$d_{ij} = \left[\sum_{a=1}^{r} \left| x_{ia} - x_{ja} \right|^{p} \right]^{1/p} \quad (p \geq 1), x_{ia} \neq x_{ja},$$

where p is referred to as the Minkowski exponent and may take any value not less than 1, r is the coordinate of point i on dimension a, and x_i is an r-element row vector from the ith row of a n-by-r matrix containing x_{ia} of all n points on all r dimensions. The MDS procedure constructs a geometric representation of the data (such as a similarity matrix), usually in a Euclidean space of low dimensionality (i.e., $p = 2$).

MDS has been applied in many different domains. He and Hui [21] applied MDS to displaying author cluster maps in their author cocitation analysis. Eom and Farris [11] applied MDS to author cocitation in decision support systems (DSS) literature over 1971 through 1990 in order to find contributing fields to DSS. Kanai and Hakozaki [22] used MDS to lay out three-dimensional thumbnail objects in an information space to visualize user preferences. Kealy [23] applied MDS to studying changes in knowledge maps of groups over time to determine the influence of a computer-based collaborative learning environment on conceptual understanding. Although much has been done to visualize relationships of objects in different domains using MDS, no attempts to apply it to discovering business intelligence on the Web have been found. In addition, no existing search engine uses MDS to facilitate Web browsing.

Visualization

Visualization is the process of displaying encoded data in a visual format that can be perceived by human eyes. The output often takes the form of a knowledge map, which is a two-dimensional picture showing the relationships among entities representing knowledge sources such as Web page content, newsgroup messages, business market trends, newspaper articles, and other textual and numerical information.

As the Web was introduced in the 1990s and has become a major knowledge repository, automatically generated knowledge maps have been proposed. CYBERMAP provides a personalized overview of a subset of all documents of interest to a user, based on his or her profile [16]. A three-dimensional colored display called Theme-

scape generates a landscape showing a set of news articles on a map with contour lines [38]. Themescape was implemented as Cartia's NewsMap (www.cartia.com) to show articles related to the financial industry. However, when many articles are placed close to a point, it is difficult for users to distinguish them without viewing details of all articles. Kartoo (www.kartoo.com), a commercial search engine, presents search results as interconnected objects on a map. Each page shows ten results, represented as circles of varying sizes corresponding to their relevance to the search query. The circles are interconnected by lines showing common keywords for the results. Different details (such as summary of results, related key words) are provided while users are moving the mouse on the screen. However, the Kartoo Web site does not explain the meaning of placement of results on the screen.

Research Framework and Testbed

Research Questions

THE LITERATURE REVIEW REVEALS THREE RESEARCH GAPS. First, existing business intelligence tools suffer from a lack of analysis and visualization capabilities. There is a need to develop better methods to enable visualization of voluminous information and discovery of communities from the Web. Second, hierarchical and map displays were shown to be effective ways of accessing and browsing information. However, they have not been widely used to discover business intelligence on the Web. Third, none of the existing search engines allows users to visualize the relationships among the search results in terms of their relative closeness. We state our three research questions as follows:

1. How can a visual framework be used to facilitate knowledge discovery on the Web?
2. How can document analysis techniques and hierarchical and map displays be used in such a framework to assist in the business intelligence analysis process?
3. In terms of effectiveness, efficiency, and usability, how do hierarchical and map displays compare with a textual result list and a graphical result map used by search engines?

To address our research questions, we have developed a visual framework for knowledge discovery on the Web. Figure 1 shows the components of the framework, consisting of three main phases: data collection; parsing, indexing, and analysis; and visualization. Each phase includes a number of steps described in detail below. Results from each step are input to the next step as shown by the annotated arrows. In the figure, the similarity graph (networked, colored nodes) produced in co-occurrence analysis is used as input to both Web community identification and knowledge map creation. In other words, we used two browsing methods to visualize the same set of Web pages. We started by selecting appropriate algorithms and techniques, proceeded to system development, and, finally, evaluated our proof-of-concept prototype empirically. Based on the framework, we developed the Business Intelligence Explorer,

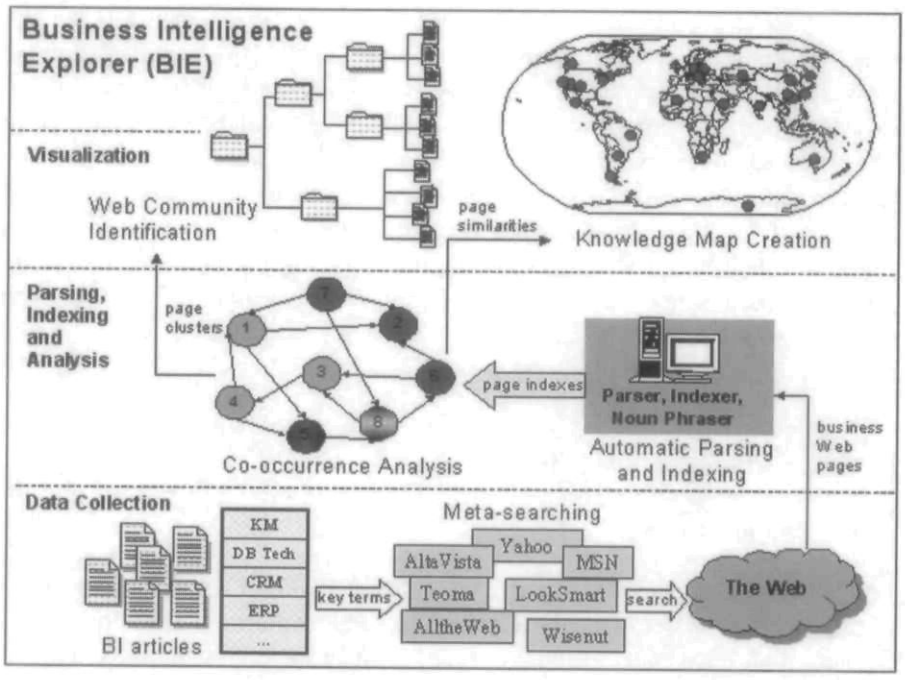

Figure 1. A Visual Framework for Knowledge Discovery on the Web

a visualization system for browsing a large number of results related to business Web sites. Specific steps of our framework are described below in the context of BIE development.

Data Collection

Data were collected in two steps: identifying key terms and meta-searching. Key topics identified in the first step were used as input queries in meta-searching.

Identifying Key Terms

The purpose of this step was to identify key terms that could be used as initial queries to search for Web pages. We chose example queries that were related to "business intelligence" to demonstrate the capability of our framework in discovering business intelligence on the Web. To identify the queries, we first entered the term "business intelligence" into the INSPEC literature indexing system, a leading English-language bibliographic information service providing access to the world's scientific and technical literature. The system returned 281 article abstracts published between 1969 and 2002, with a majority of the articles (230 articles) published in the past 5 years. Based on the key words appearing in the titles and abstracts, we identified the following nine key terms: knowledge management, information management, database tech-

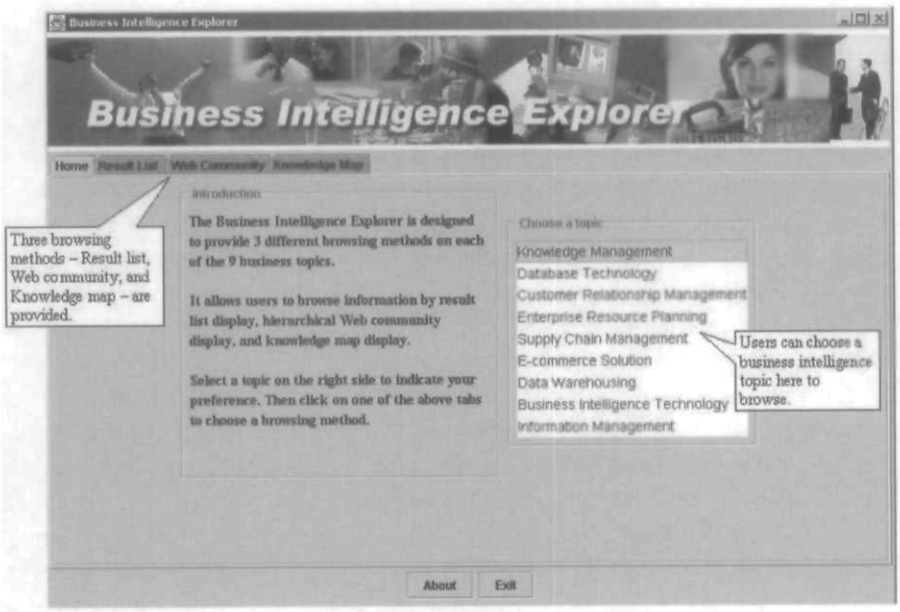

Figure 2. User Interface of the Business Intelligence Explorer

nology, customer relationship management, enterprise resource planning, supply chain management, e-commerce solutions, data warehousing, and business intelligence technology. These became the nine key topics on business intelligence and are shown on the front page of the system's user interface in Figure 2. Alternatively, these topics can be identified by other methods such as expert judgment.

Meta-Searching

Using the nine business topics, we performed meta-searching on seven major search engines: AltaVista, AlltheWeb, Yahoo, MSN, LookSmart, Teoma, and Wisenut. They are the major search engines also used by Kartoo, which was compared with the knowledge map of our system. Kartoo is a new meta-search engine that presents results in a map format. We tried to create a collection that is comparable to the one used by Kartoo. From each of the seven search engines, we collected the top 100 results. As page redirection is used in the front page of many Web sites, our spider automatically followed these URLs to fetch the pages that were redirected. Since we are interested in only business Web sites, URLs from educational, government, and military domains (with host domain "edu," "gov," and "mil," respectively) were removed. Further filtering was applied to remove non-English Web sites, academic Web sites that do not use the "edu" domain name, Web directories and search engines, online magazines, newsletters, general news articles, discussion forums, case studies, and so on. In total, we collected 3,149 Web pages from 2,860 distinct Web sites, or about 350 Web pages for each of the nine topics. Each Web page represented one Web site.

Automatic Parsing and Indexing

Since Web pages contain both textual content and HTML tag information, we needed to parse out this information to facilitate further analysis. We used a parser to automatically extract key words and hyperlinks from the Web pages collected in the previous step. A stop word list of 444 words was used to remove nonsemantic-bearing words (e.g., "the," "a," "of," "and"). Using HTML tags (such as <TITLE>, <H1>, <IMG SRC= 'car.gif' alt= 'Ford'>), the parser also identified types of words and indexed the words appearing on each Web page. Four types of words were identified (in descending order of importance): title, heading, content text, and image alternate text. If a word belonged to more than one type, then the most important type was used to represent that term on the page. The word-type information was used in the co-occurrence analysis step (discussed below). Then we used Arizona Noun Phraser (AZNP) to extract and index all the noun phrases from each Web page automatically based on part-of-speech tagging and linguistic rules [36]. For example, the phrase *strategic knowledge management* will be considered a valid noun phrase because it matches the noun-phrase rule: adjective + noun + noun. We then treated each key word or noun phrase as a subject descriptor. Based on a revised automatic indexing technique [31], we measured the term's level of importance by term frequency and inverse Web page frequency. Term frequency measures how often a particular term occurs in a Web page. Inverse Web page frequency indicates the specificity of the term and allows terms to acquire different strengths or levels of importance based on their specificity. A term could be a one-, two-, or three-word phrase.

Co-occurrence Analysis

Co-occurrence analysis converted raw data (indices and weights) obtained from the previous step into a matrix that showed the similarity between every pair of Web sites. The similarity between every pair of Web sites contained its content and structural (connectivity) information. He et al. [20] computed the similarity between every pair of Web pages by a combination of hyperlink structure, textual information, and cocitation. However, their algorithm placed a stronger emphasis on cocitation than hyperlink and textual information. When no hyperlink existed between a pair of Web pages, the similarity weight included only cocitation weight, even if their textual content were very similar. The same situation appeared when no common word appeared in the pair of Web pages, even if many hyperlinks existed between them. In order to impose a more flexible weighting in the three types of information, we modified He et al.'s algorithm to compute the similarity. The following shows the formulas used in our co-occurrence analysis.

Similarity between site i and site j is

$$W_{ij} = \alpha \frac{A_{ij}}{\|A\|_2} + \beta \frac{S_{ij}}{\|S\|_2} + (1-\alpha-\beta)\frac{C_{ij}}{\|C\|_2}, \tag{1}$$

where A, S, and C are matrices for A_{ij}, S_{ij}, and C_{ij}, respectively, α and β are parameters between 0 and 1, and $0 \leq \alpha + \beta \leq 1$ ($A_{ij} = 1$ if site i has a hyperlink to site j, $A_{ij} = 0$ otherwise; S_{ij} = asymmetric similarity score between site i and site j [5]).

$$S_{ij} = sim\left(D_i, D_j\right) = \frac{\sum_{k=1}^{p} d_{ki} d_{kj}}{\sum_{k=1}^{n} d_{di}^2} \quad S_{ji} = sim\left(D_j, D_i\right) = \frac{\sum_{k=1}^{p} d_{ki} d_{kj}}{\sum_{k=1}^{m} d_{kj}^2},$$

where n is total number of terms in D_i; m is total number of terms in D_j; p is total number of terms that appear in both D_i and D_j; d_{ij} is $tf_{ij} \times \log(N/df_j \times w_j) \times (Termtype\ factor)_j$; tf_{ij} is number of occurrences of term j in Web page i; df_j is number of Web pages containing term j; w_j is number of words in term j; $(Termtype\ factor)_j = 1 + ((10 - 2 \times type_j)/10)$, where $type_j = \min\{1$ if term j appears in title; 2 if term j appears in heading, 3 if term j appears in content text, 4 if term j appears in image alternate text$\}$; C_{ij} is number of Web sites pointing to both site i and site j (cocitation matrix).

We normalized each of the three parts of Equation (1) in computing the similarity and assigned a weighting factor to each of them independently. We computed the similarity of textual information (S_{ij}) by asymmetric similarity function, which was shown to perform better than cosine function [5]. When computing the term importance value (d_{ij}), we included a term type factor that reflected the importance of the term inside a Web page. Using the formulas shown above, a similarity matrix for every pair of Web sites in each of the nine business intelligence topics was generated.

Web Community Identification

We define a Web community as a group of Web sites that exhibit high similarity in their textual and structural information. The term *Web community* originally was used by researchers in Web structure mining to refer to a set of Web sites that are closely related through the existence of links among them [13]. Similarly, researchers in Web content mining use the term "cluster" to refer to a set of Web pages that are closely related through the co-occurrence of key words or phrases among them [6]. We chose to use "Web community" to refer to our groups of similar Web sites because it connotes the meanings of common interests and mutual references (but not just a group of similar objects as connoted by "cluster"), and our similarity calculation has reflected content, hyperlink, and cocitation similarities (see previous subsection).

To identify Web communities for each business intelligence topic, we modeled the set of Web sites as a graph consisting of nodes (Web sites) and edges (similarities). Based on previous research, hierarchical and partitional clustering had been found applicable for different purposes, but it was not likely that any one method was the best [19]. Here, we decided to choose a combination of hierarchical and partitional clustering so as to obtain the benefits of both methods. We used a partitional cluster method to partition the Web graph recursively in order to create a hierarchy of clusters. This way,

we could obtain the clustering quality of the partitional method while being able to show the results in a visual dendrogram. However, partitional clustering is computationally intensive; search heuristics were required to find good solutions. To obtain high quality of clustering using partitional methods, we used an optimization technique that finds the "best" partition point in each cluster task. Being a global search technique, genetic algorithms can perform a more thorough space search than some other optimization techniques (such as taboo search and simulated annealing). Genetic algorithms are also suitable for large search space such as the Web graph. Therefore, we selected genetic algorithms as the optimization technique in our graph partitioning. During each iteration, the algorithm tries to find a way to bipartition the graph such that a certain criterion (the fitness function) is optimized. Based on previous work on Web page clustering [20] and image segmentation [33], we used a normalized cut criterion as the genetic algorithm's fitness function, which measures both the total dissimilarity between different partitions as well as the total similarity within the partitions. It has been shown to outperform the minimum cut criterion, which favors cutting small sets of isolated nodes in the graphs [39].

Figure 3 shows the formulas used to compute the normalized cut in the partitioning of a graph into two parts: A and B. We modeled the set of Web sites as a graph. Each node represents a Web page from a site and each directed link represents a hyperlink pointing from a site to another. A cut on a graph $G = (V, E)$ is the removal of a set of edges such that the graph is split into two disconnected subgraphs.

$$cut(x) = cut(A, B) = \sum_{i \in A, j \in B} W_{ij},$$

where W_{ij} is similarity between node i and node j; A, B are two distinct partitions in G; x is a binary vector showing which nodes belong to which partitions. For example, in Figure 3, the number on each node represents the position of the digit in x. When the digit is 0, then the kth node belongs to partition A. When the digit is 1, then the kth node belongs to partition B. In the example shown in Figure 3, $x = 01001101$.

The normalized cut is a fraction of the total edge connections to all the nodes in Figure 3.

$$Ncut(x) = \frac{cut(A,B)}{assoc(A,V)} + \left(\frac{cut(A,B)}{assoc(B,V)}\right),$$

where $assoc(A,V) = \Sigma_{u \in A, t \in V} W_{ut}$ is the total connection from nodes in A to all nodes in the graph and $assoc(B,V) = \Sigma_{u \in B, t \in V} W_{ut}$ is similarly defined (see Figure 3). It was also shown in Shi and Malik [33] that $Ncut(x) = 2 - Nasso(x)$, where $Nasso(x) = (assoc(A,A))/(assoc(A,V)) + (assoc(B,B))/(assoc(B,V))$ reflects how tightly on average nodes within the group are connected to each other. Because the sum of $Ncut(x)$ and $Nasso(x)$ is a constant, minimizing the dissimilarity between the two partitions (i.e., minimizing $Ncut(x)$) and maximizing the similarity within the partitions (i.e., maximizing $Nasso(x)$) are identical.

When partitioning the Web graph into Web communities, the genetic algorithms tried to maximize the value of $Nasso(x)$, which reveals how closely the sites exist

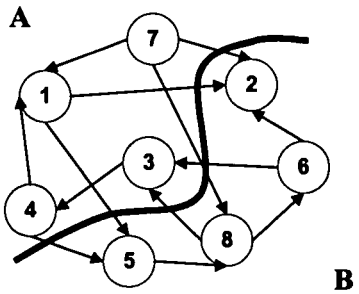

Figure 3. An Example of Graph Partitioning

within a Web community. Figure 4 shows the steps in the genetic algorithms, and Figure 5 illustrates how the genetic algorithms worked by a simplified example, in which we set the maximum number of levels in the hierarchy to be 2 and the maximum number of nodes in the bottom level to be 5. Ten nodes were initially partitioned into graph A and graph B, forming the first level of the hierarchy. Since graph B contained fewer than 5 nodes, the partitioning stopped there. Graph A continued to be partitioned to create graphs C and D, which were on the second level of the hierarchy. Then the whole procedure stopped, because the maximum number of levels and the maximum number of nodes in the bottom level had been reached. The graphs (A, B, C, D) partitioned in the process were considered to be Web communities. In our actual Web site partitioning, the maximum number of levels is 5 and the maximum number of nodes in the bottom level was 30. These numbers were determined by considering the visual cluttering that may result from having too many levels and nodes shown on the screen. The partitioned Web communities were labeled by the top 10 phrases having the highest term importance value (d_{ij} shown in the previous subsection). Manual selection among these 10 phrases was used to obtain the one that best described the community of Web sites.

Knowledge Map Creation

In this step, we used MDS to transform a high-dimension similarity matrix into a two-dimensional representation of points and displayed them on a map. As described in our literature review, MDS has been applied in different domains for visualizing the underlying structure of data. We used Torgerson's classical MDS procedure, which does not require iterative improvement [37]. The procedure was shown to work with non-Euclidean distance matrixes (such as the one we used here) by giving approximation of the coordinates [40]. We detail the MDS procedure as follows:

1. Convert the similarity matrix into a dissimilarity matrix by subtracting each element by the maximum value in the original matrix. Call the new dissimilarity matrix D.
2. Calculate matrix B, which is the scalar products, by using the cosine law. Each element in B is given by

Step 0. Set up parameters for convergence:
- GA convergence criteria: population size, maximum number of generations, crossover, and mutation probability.
- Hierarchical clustering convergence criteria: maximum number of levels in hierarchy, minimum number of nodes.

Step 1. A chromosome is represented by a binary vector that has the same number of digits as the number of nodes in the graph to be bipartitioned. Each gene represents a Web site on a partition. The chromosome is equivalent to the binary vector x in Figure 5.

Step 2. Initialize the population randomly.

Step 3. Evaluate the fitness of each chromosome using Nasso measure.

Step 4. Spin the roulette wheel to select individual chromosomes for reproduction based on the probability of their fitness relative to the total fitness value of all chromosomes.

Step 5. Apply crossover and mutation operators.

Step 6. Evaluate the fitness and find the highest fitness value in this generation.

Step 7. Check if genetic algorithms convergence criteria are met. If not, go back to step 3. Otherwise, use the best chromosome to guide the graph partitioning, store the results in a tree structure, and proceed to step 8.

Step 8. Check if the hierarchical clustering convergence criteria are met. If not, go back to step 2 and apply genetic algorithms to partition each of the two graphs partitioned in step 7. Otherwise, stop the whole procedure and return the results.

Figure 4. Steps in Using a Genetic Algorithm for Recursive Web Graph Partitioning

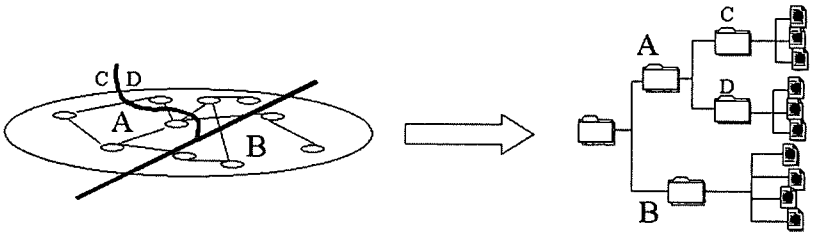

Figure 5. A Simplified Example of Genetic Algorithm Graph Partitioning

$$b_{ij} = -\frac{1}{2}\left(d_{ij}^2 - \frac{1}{n}\sum_{k=1}^{n} d_{ik}^2 - \frac{1}{n}\sum_{k=1}^{n} d_{kj}^2 + \frac{1}{n^2}\sum_{g=1}^{n}\sum_{h=1}^{n} d_{gh}^2\right),$$

where d_{ij} is an element in D, n = number of nodes in the Web graph.

3. Perform a singular value decomposition on B and use the following formulas to find out the coordinates of points.

$$B = U \times V \times U' \tag{2}$$

$$X = U \times V^{1/2}, \tag{3}$$

where U has eigenvectors in its columns and V has eigenvectors on its diagonal. Combining (2) and (3), we have $B = X \times X'$.

We then used the first two column vectors of X to obtain the two-dimensional coordinates of points, which were used to place the Web sites on our knowledge maps.

Evaluation Methodology

THIS SECTION DESCRIBES THE EVALUATION METHODOLOGY used to compare different browsing methods. Details of the evaluation objectives, variables involved, experimental tasks, hypotheses, subject profile, and performance measurement are described.

Objectives

The objectives of the evaluation were to understand the effectiveness, efficiency, and usability of the two browsing methods we developed—Web community and knowledge map—and to compare them with a textual result list display and a powerful commercial searching and browsing tool called Kartoo (www.kartoo.com). We chose Kartoo because it displays results in a graphical map format that is most nearly comparable to our knowledge map. Our textual result list display resembles the display of results presented by typical search engines such as Google. We used the same relevancy rankings as returned by meta-searching the seven selected search engines and combined the ranked lists according to the order in which the results were fetched by our spidering program. More advanced compilation of results was not performed because it was not the focus of our study. To make the comparison with other browsing methods fairer, we did not provide a search function, which also was not our focus of study. Kartoo displays search results in a map format, with circles representing Web sites and lines linking the Web sites through common key words. The independent variables studied in the experiment were four browsing methods (result list, Web community, knowledge map, and Kartoo map). The dependent variables were three system attributes (effectiveness, efficiency, and usability). Figures 6 through 9 show the screenshots of the four browsing methods.

Experimental Tasks

Two experimental tasks related to one of the nine business intelligence topics were designed for each browsing method. Task 1 incorporated the close-ended tasks used in the Text REtrieval Conferences (TREC), and an exact answer to each task was expected. In the task, subjects were given the names of two companies and were asked to find their URLs and major business areas in four minutes. An example was "Use the Web community display, select the topic *Supply Chain Management,* find the URL and the major business areas of *Gensym Corporation.*" Task 2 was designed according to the open-ended tasks used in TREC, and topics relevant to each task

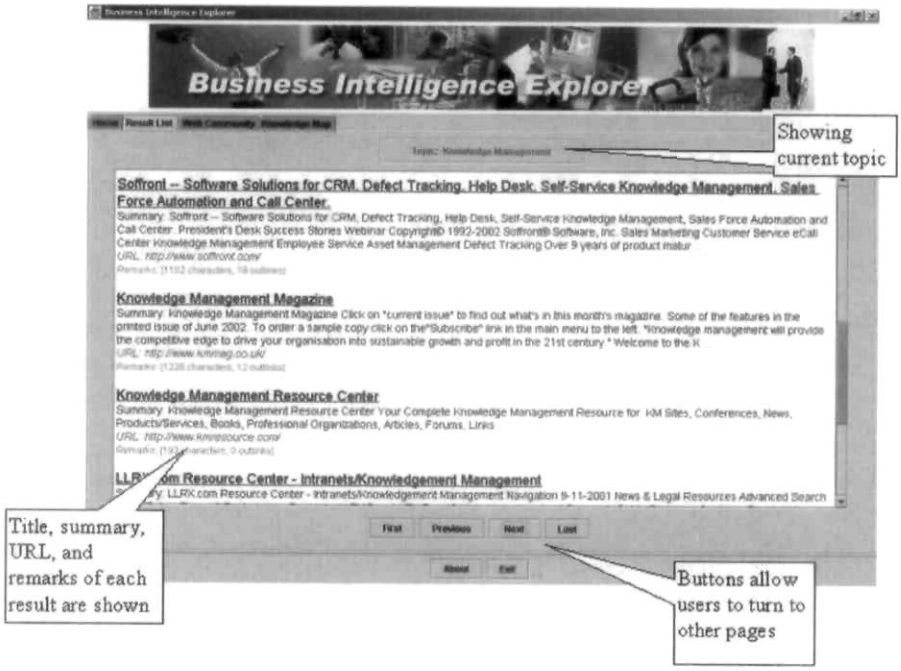

Figure 6. Result List Browsing Method

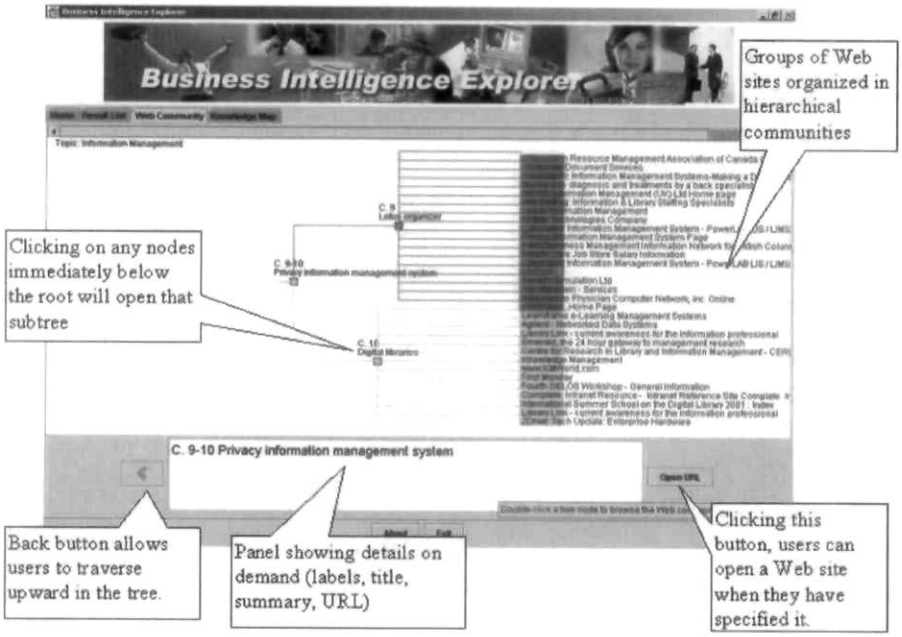

Figure 7. Web Community Browsing Method

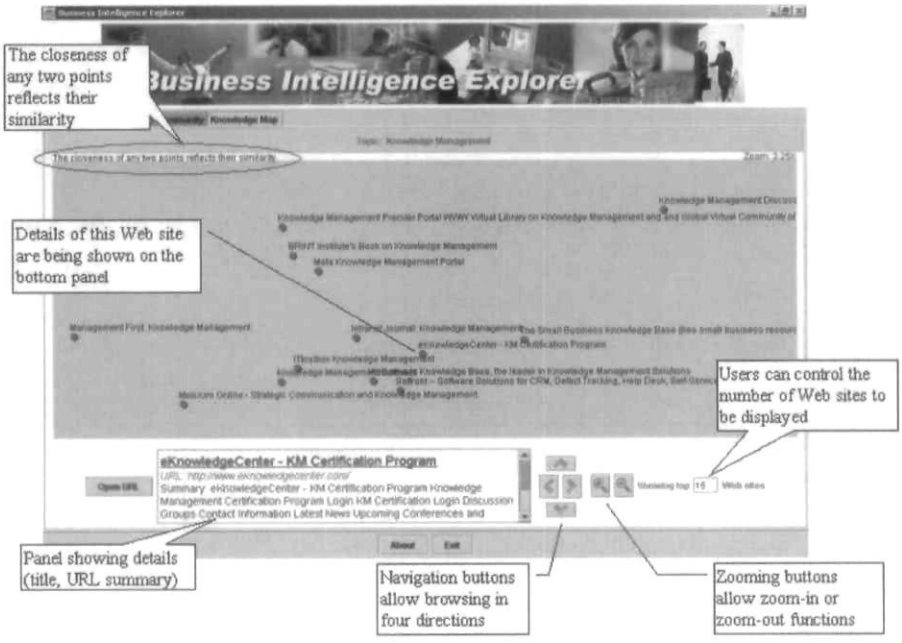

Figure 8. Knowledge Map Browsing Method

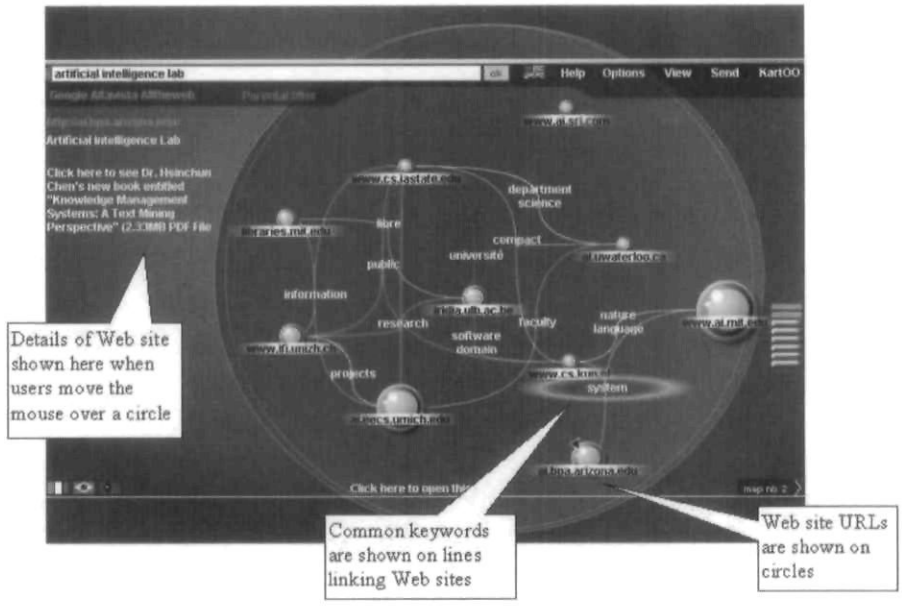

Figure 9. Kartoo Map Browsing Method

were expected to be found. In the task, subjects were given a broad issue in a business intelligence topic and were asked to find the titles and URLs of Web sites related to that issue in eight minutes. They could find as many Web sites as they wanted. An example was "Use the result list display, select the topic *Customer Relationship Management,* find the titles and URLs of Web sites that are related to *CRM benchmarking.*" Thirty University of Arizona students participated voluntarily in the experiment. Half of them were female. Most were younger than thirty and were attending business school.

Experimental Design and Hypotheses

Three comparisons were performed in our experiment. Web community was compared with result list because we wanted to determine whether clustering and hierarchical display helped subjects visualize the business Web sites. Knowledge map was compared with Web community because we wanted to study the visual effects and accuracy of the different browsing methods. Knowledge map was compared with Kartoo map to examine point placement and the visual effects of using different browsing methods. Other pairwise comparisons could be performed, but we were mainly interested in these three.

As each subject was asked to perform the same set of tasks using the four browsing methods, a one-factor repeated-measures design was used, because it gives greater precision than designs that employ only between-subjects factors [27, p. 280]. The whole experiment took about an hour and was divided into four sections. In each section, subjects used one of the four browsing methods to perform two tasks as described above. The task contents were different for different browsing methods but their natures were the same (i.e., close-ended for Task 1 and open-ended for Task 2). The functionalities of each browsing method were explained to subjects before they used it. During the experiment, the experimenter also provided necessary assistance to subjects without showing any value judgment and recorded their behavior, verbal protocols, and other observational data. In the following, we describe each system performance factor and state our hypotheses.

System Performance Factors

The system performance factors studied in the experiment were effectiveness, efficiency, and usability. The effectiveness of a browsing method has three components: accuracy, precision, and recall. Accuracy refers to how well the browsing methods helps users find exact answers to close-ended tasks (Task 1). Precision measures how well the browsing method helps users find relevant results and avoid irrelevant results in open-ended tasks (Task 2). Recall measures how well a browsing method helps users find all the relevant results in open-ended tasks (Task 2). A single measure, called *F-value,* was used to combine recall and precision. An expert who had a master's degree in library and information science and five years of content management and Internet searching experience was recruited to provide answers to the experimental

tasks for judging the effectiveness. She spent one hour searching and browsing the Web, using our tool, Kartoo, and other search tools such as Google, to manually identify all relevant results. This way, her judgment of relevance was based on a wide range of information sources (but not just on our collection). The formulas to obtain the above measurements are stated below:

$$Accuracy = \frac{Number\ of\ correctly\ answered\ questions}{Total\ number\ of\ questions}$$

$$Precision = \frac{Number\ of\ relevant\ results\ identified\ by\ the\ subject}{Number\ of\ all\ results\ identified\ by\ the\ subject}$$

$$Recall = \frac{Number\ of\ relevant\ results\ identified\ by\ the\ subject}{Number\ of\ relevant\ results\ identified\ by\ the\ expert}$$

$$F\text{-}value = \frac{2 \times Recall \times Precision}{Recall + Precision}.$$

Efficiency refers to the amount of time users required to use a browsing method to finish the tasks. Usability refers to how satisfied users are with using a browsing method. This was obtained from subjects' overall ratings of the browsing method on a five-point Likert scale, where 1 means "poor" and 5 means "excellent." One feature of each browsing method was selected for rating (using the same five-point scale): textual list of result list, labels of Web community, placement of points in knowledge map, and placement of circles in Kartoo map. Subjects were encouraged to speak during the experiment so the experimenter could record the reasons behind their behaviors and feelings. They also could type their comments regarding the strengths and weaknesses of the methods into a computer file. All verbal comments were analyzed using protocol analysis [12]. Moreover, we asked subjects to compare knowledge map with Kartoo map in the following aspects: user friendliness, graphical user interface, quality of results, and meaning of placement of points. For each aspect, subjects provided verbal comments and indicated whether knowledge map or Kartoo map performed better.

Hypotheses

Nine hypotheses, grouped by the three system attributes, were tested and are listed below.

H1: Web community is more effective than result list.

H2: Knowledge map is more effective than Web community.

H3: Knowledge map is more effective than Kartoo map.

H4: Web community is more efficient than result list.

H5: Knowledge map is more efficient than Web community.

H6: Knowledge map is more efficient than Kartoo map.

H7: Web community obtains higher users' ratings than result list.

H8: Knowledge map obtains higher users' ratings than Web community.

H9: Knowledge map's placement of Web sites is more meaningful than Kartoo map's placement.

In H1, H4, and H7, we believed that Web community would perform better than result list in terms of the three system attributes because of Web community's visual effect and accurate clustering.

In H2, H5, and H8, knowledge map was compared against Web community. As subjects could compare the physical distances among different points in knowledge map, but were not provided clues to distinguish among Web sites within the same community in Web community, we believed that knowledge map was more flexible and would perform better than Web community in terms of the three system attributes. In H3, H6, and H9, knowledge map was compared against Kartoo map. We believed that the relative closeness of Web sites in, and the cleaner interface of, knowledge map would enable subjects to find relevant results more quickly than they did in Kartoo map, in which the placement of circles did not explicitly correspond with Web sites' similarity. We thus considered Kartoo map to be less intuitive and believed that knowledge map would perform better in terms of the three system attributes.

Experimental Results and Discussions

THIS SECTION DESCRIBES THE RESULTS of our user study comparing four browsing methods. Table 1 summarizes the means, standard deviations, and rankings of different performance measures for different browsing methods. Table 2 shows the p-values of various t-tests in testing the hypotheses.

Comparison Between Web Community and Result List

From the testing results of H1, H4, and H7, we found that Web community performed significantly better than result list in terms of all performance measures. We believe that the advantages of clustering and visualization were the main contributing factors. In terms of effectiveness, the main reason for the superior performance of Web community was its ability to group similar Web sites in the same community accurately, whereas result list's one-dimensional list did not.

In terms of efficiency, we believe that Web community's hierarchical structure allowed subjects to visualize the landscape of the entire collection of Web sites. Subjects using result list spent much time opening the result pages sequentially. In contrast, subjects relying on Web community's visual effects could quickly locate the key labels related to the topics they were searching for, thus saving time.

Table 1. Summary of Key Statistics

Browsing method	Performance measure	Average	Standard deviation	Ranking
Result list	Accuracy (percent)	36.67	26.04	Second
	Precision (percent)	83.81	28.47	Second
	Recall (percent)	5.69	6.48	Fourth
	F-value (percent)	10.06	9.79	Fourth
	Total time (minutes)	12.00	0.00	Fourth
	Rating on result listing	2.33	0.96	—
	Overall rating	2.70	0.92	Third
Web community	Accuracy (percent)	**100.00**	0.00	First
	Precision (percent)	**92.05**	5.66	First
	Recall (percent)	64.58	13.96	Second
	F-value (percent)	**74.89**	11.10	First
	Total time (minutes)	9.30	1.67	Second
	Rating on node labeling	4.70	0.54	—
	Overall rating	**4.20**	0.62	First
Knowledge map	Accuracy (percent)	**100.00**	0.00	First
	Precision (percent)	66.13	16.63	Third
	Recall (percent)	70.00	18.45	First
	F-value (percent)	66.85	15.26	Second
	Total time (minutes)	**9.29**	0.61	First
	Rating on point placement	4.02	0.91	—
	Overall rating	3.83	0.95	Second
Kartoo map	Accuracy (percent)	21.67	25.20	Third
	Precision (percent)	38.00	26.05	Fourth
	Recall (percent)	12.86	6.55	Third
	F-value	18.29	8.89	Third
	Total time (minutes)	11.91	0.45	Third
	Placement rating	2.22	1.03	—

Notes: All ratings are on a five-point discrete Likert scale, where 1 means "poor" or "not helpful," and 5 means "excellent" or "very helpful." Bold figures indicate the best performance achieved among all of the browsing methods.

In terms of usability, subjects rated Web community significantly higher than result list, probably because of Web community's visualization effects, clustering into hierarchical groups, labeling, and providing details on demand. As subject 19 said, "Once I spot the label, I can move to the relevant topics very easily . . . (Web community) save(s) time, (I) don't need to read all the summaries and Web pages to decide which are relevant." Fifteen subjects had similar comments. Regarding visualization effects, subject 10 said that "visualization helps to navigate faster and easier." Eight subjects said that the tree structure was helpful, and seven subjects said that labels were clear. We therefore believe that clustering and visualization of Web community contributed to its higher rating.

In addition, subjects also commented on the strengths and weaknesses of result list and Web community. They were familiar with result list's display of results because it was similar to typical search engines' displays. However, result list display provided

Table 2. p-Values of Various t-Tests

Effectiveness					
Hypothesis	Accuracy	Recall	Precision	F-value	Conclusion
H1: WC > RL	0.000 (>)[1]	0.000 (>)	0.154 (>*)[2]	0.000 (>)	Confirmed
H2: KM > WC	Undefined[3]	0.114 (>*)	0.000 (>)	0.007 (<)	Not confirmed
H3: KM > KT	0.000 (>)	0.000 (>)	0.000 (>)	0.000 (>)	Confirmed
Efficiency					
Hypothesis	Time for task 1	Time for task 2	Total time		Conclusion
H4: WC > RL[4]	0.000 (>)	0.000 (>)	0.000 (>)		Confirmed
H5: KM > WC	0.000 (>)	0.000 (<)	0.970 (>*)		Not confirmed
H6: KM > KT	0.000 (>)	0.191 (>*)	0.000 (>)		Confirmed
Usability					
Hypothesis	Rating on feature	Overall rating	Combined rating		Conclusion
H7: WC > RL	0.000 (>)	0.000 (>)	0.000 (>)		Confirmed
H8: KM > WC	0.000 (<)	0.030 (<)	0.000 (<)		Not confirmed[5]
		Rating on placement of Web sites			
H9: KM > KT		0.000 (>)			Confirmed

[1] A greater-than sign (>) means that the method on the left of the hypothesis yields a higher mean than the method on the right. A less-than sign (<) means the reverse. [2] An asterisk (*) means that t-test result was not significant at a 95 percent confidence level. [3] The correlation and t cannot be computed because the standard error of the difference is 0. [4] For H4, H5, and H6, a greater-than sign (>) means that the method on the left yields a higher efficiency (or smaller amount of time) than the method on the right. A less-than sign (<) means the reverse. [5] H8 was not confirmed, but the opposite of H8 was confirmed. WC—Web community; RL—result listing; KM—knowledge map; KT—Kartoo map.

too much information and might have created information overload. Subject 4 said that result list required "too much reading at one time" and that it was "hard to search for a specific word or phrase." Eight subjects said that it was hard to find specific words, four subjects said too much information was provided, and three subjects complained that they could not use the search function to search for a specific key word. Regarding the weaknesses of Web community, ten subjects said that when they browsed at the root level the labels were overlapped and looked crowded. *To summarize, we found that Web community performed significantly better than result list in terms of effectiveness, efficiency, and usability.*

Comparison Between Web Community and Knowledge Map

Regarding the results of testing H2, H5, and H8, surprisingly, we found that all three of these hypotheses were not confirmed, and the opposite of H8 was confirmed. In other words, knowledge map performed very similarly to Web community in terms of effectiveness and efficiency. We suggest three reasons for such results. First, both knowledge map and Web community facilitated subjects' browsing by visualization. Second, both knowledge map and Web community employed the same Web site similarity data to perform further analysis. Third, both methods provided a visual summary of the entire collection while allowing details on demand.

In terms of usability (H8), Web community was rated significantly higher than knowledge map, contrary to our expectation. We believe that knowledge map's inadequate zooming function contributed to the result. By default, the top ten results were displayed on knowledge map. When subjects performed the tasks, they could increase the number of Web sites shown on screen. This would also increase the chance that titles overlapped. Unfortunately, knowledge map's zooming function was restricted to between one and five times the original size. Thus, if subjects chose to display more than 50 results, they were presented with too many overlapping labels. On the other hand, Web community's hierarchical display could automatically resize the subtree being displayed. As subjects zoomed in on lower levels close to the bottom, they could get a clearer picture of the results. Therefore, the overlapping-label problem did not appear in those lower levels. In contrast, when subjects used knowledge map, they got more and more information as they increased the number of Web sites shown on screen, thus increasing the chance of information overload. As subject 30 commented on knowledge map: "(It is) easy to forget the Web sites I searched. It makes me do some repeating work." *To summarize, we found that knowledge map has effectiveness and efficiency similar to those of Web community because of their similar functionality. But knowledge map was rated less usable than Web community because of its inadequate zooming function.*

Comparison Between Knowledge Map and Kartoo Map

The results of testing H3, H6, and H9 show that knowledge map performed significantly better than Kartoo map in terms of effectiveness, efficiency, and users' ratings

Table 3. Number of Subjects Who Expressed a Preference for Knowledge Map or Kartoo

System attributes	Knowledge map is better	Kartoo is better	Similar/ undecided
User-friendliness	20	7	3
Graphical user interface	2	27	1
Quality of results	26	0	4
Meaning of placement of Web sites	25	0	5

on the meaning conveyed by placement of points. We suggest that the main reasons contributing to the superior performance of knowledge map were its clean interface, intuitive communication of the meaning of placement of points, and provision of details on demand. As subject 9 summarized succinctly: "This is an intelligent tool and has features superior to any other search, as it gives a visual picture of the topic and all topics closely related to the one under search. The map is intuitive and helps steer the user to the right topics or the ones that are close." Twenty subjects agreed that the closeness of points and their relationships with similarity helped in browsing. Five subjects said that finding results was quick using knowledge map, and four subjects said that the display was clear.

Subjects' verbal comments revealed that knowledge map had better user-friendliness, quality of results, and placement of Web sites on the screen, whereas Kartoo map had a better graphical user interface. Table 3 shows their preferences on the four aspects. Subjects pointed out the inaccuracy of results and problems of user interface of Kartoo map. Unlike knowledge map, which displayed the titles of the search results, Kartoo map displayed Web site URLs that often did not bear semantic meaning. When a URL did not contain the search terms a subject was looking for, she could not decide whether the URL was relevant unless she opened the Web site to browse its content or moved the mouse over the circle (representing the Web site) to see the title and summary. But when the pointer was moved away from the circle, the title and summary could no longer be seen, because Kartoo map displayed only the details of the Web site to which the pointer had been moved.

In addition, Kartoo provided much information that might have confused subjects. As subject 18 pointed out: "Too many lines in Kartoo confused the users when we used it." In general, subjects had difficulty adapting to the complicated display of Kartoo. However, Kartoo, being a commercial search engine, had a more professional graphical user interface that most subjects preferred. *To summarize, we found that knowledge map performed significantly better than Kartoo in terms of effectiveness, efficiency, and users' rating on placement of Web sites because of knowledge map's accurate placement of Web sites and its clean interface. From users' verbal comments, we found that knowledge map was more user-friendly, produced quality results, and conveyed better meaning of the placement of Web sites because of accurate placement and clear presentation. Kartoo map was considered to have a better graphical user interface because of its professional graphical design.*

Discussion

The experimental results supported our belief that Web community and knowledge map could help to reduce information overload and discover business intelligence on the Web. We concluded that appropriate use of visualization was the main contributor to superior performance. We can explain it by using Shneiderman's taxonomy for information visualizations [34]. Among the seven data types (1D, 2D, 3D, temporal, network, tree, and multidimensional), knowledge map represents a two-dimensional/network data type, whereas Web community represents a two-dimensional/tree data type. Both methods employ techniques to perform some of the seven visualization tasks (overview, zoom, filter, detail on demand, extract, related, and history) appropriately in order to reduce information overload. Web community's clustering provides an *overview* of the entire collection of Web sites, and its tree structure provides *details on demand.* Knowledge map's intelligent placement of Web sites as points on a map also provides an *overview* of the Web sites. Its intuitive depiction of the relationship between physical distance on map and Web site similarity helps users *relate* easily to different Web sites. Moreover, both Web community and knowledge map *extract* information (title, summary, URL) from the Web sites that users click on. Knowledge map allows users to *zoom* into finer levels; Web community allows users successively to open nodes containing communities of Web sites. Knowledge map *filters* information by allowing users to change the number of Web sites displayed; Web community *filters* information by not displaying the communities that users do not select. However, neither Web community nor knowledge map provides a *history* of actions performed by users. Although this did not affect the significance of results due to the short durations of tasks, providing the history might have helped users' browsing, as reflected by a comment from subject 30, "(It is) easy to forget the Web sites I searched. It makes me do some repeating work."

Conclusion

IN THIS PAPER, WE DESCRIBE A VISUAL FRAMEWORK for knowledge discovery on the Web. This generic framework integrated techniques in content collection, text mining, and document visualization to address the problem of information overload on the Web. We have incorporated two new browsing methods—Web community and knowledge map—into the framework upon which we built *Business Intelligence Explorer,* a visualization system that helps users explore and understand a large number of business Web sites. A user study was conducted to evaluate the browsing methods and to compare them with a result list display and Kartoo map display, a commercial search engine.

Web community was found to perform significantly better than result list in terms of effectiveness, efficiency, and usability because of its accurate clustering and appealing visualization. Knowledge map was found to perform similarly to Web community in terms of effectiveness and efficiency due to their similar visualization features and their use of the same similarity matrix in analysis. Contrary to our expectation,

Web community obtained significantly higher users' ratings than knowledge map because of knowledge map's inadequate zooming function. When comparing knowledge map against Kartoo map, we found that knowledge map performed significantly better in terms of effectiveness, efficiency, and users' ratings on placement of Web sites because of knowledge map's accurate placement of Web sites and clean interface. Subjects' comments indicated that knowledge map had better user-friendliness, higher-quality results, and more meaningful placement of Web sites, whereas Kartoo map had a better graphical user interface.

The contributions of this research are threefold. First, our framework is promising for alleviating information overload and discovering business intelligence on the Web. Knowledge was discovered through mining the Web pages returned by multiple search engines. It is potentially useful to business analysts for whom it could be used to effectively and efficiently extract knowledge, in the form of patterns, from large amounts of information. Incorporating the framework into an enterprise generally involves participation from business analysts and managers who have a basic knowledge of the business environment. Second, the Web community and knowledge map browsing methods are particularly suitable for visually summarizing a large number of Web sites or search results. They helped to transform raw data on Web pages into business intelligence that shows relationships among business Web sites, thus augmenting existing business intelligence tools. Search engine developers can use them as alternatives to result list display for Web analysis. Third, findings from our user study have practical implications for search engine developers and human–computer interaction researchers, who can use the findings to develop new techniques to facilitate Web browsing.

Three future directions will be explored. First, other algorithms can be applied in the different steps of the framework. Currently, BIE is limited by high computational intensity in co-occurrence analysis, which requires $O(n^2)$ computational time, and cumbersome identification of Web communities. Faster algorithms might be used to improve performance. In addition, the calculation of similarity between Web sites can include more attributes specific to the domain and Web pages. Placement algorithms other than MDS might be used to compute the coordinates of points on the knowledge map.

Second, new visualization metaphors can be developed for Web browsing. Metaphors that exploit the nature of the Web as well as features of the domain may bring a more satisfactory and pleasurable browsing experience to users. The map display and hierarchical display used in this research are only two of the many potentially available visualization metaphors. Others such as three-dimensional displays and animations could be further studied.

Third, other domain areas can be explored to create knowledge maps and to discover communities. Cross-regional and temporal analysis of business intelligence on the Web is an interesting area to which our framework could be applied. The critical factors and cost/benefit analysis of incorporating our framework into an enterprise is another interesting area of study. Our framework also might be applied to areas where intelligence analysis is required to discover knowledge from a large amount of data.

Acknowledgments: This research was partly supported by two research grants: NSF Digital Library Initiative-2, "High Performance Digital Library Classification Systems: From Information Retrieval to Knowledge Management," IIS-9817473, April 1999–March 2002, and NSF Digital Government Program, "COPLINK Center: Information and Knowledge Management for Law Enforcement," 9983304, July 2000–June 2003. They authors thank Olivia R. Liu Sheng, Chunju Tseng, Jennifer Jie Xu, Barbara Sears, and Ann Lally for their help with different parts of this research. They also thank all the subjects who participated in the user study.

REFERENCES

1. Bharat, K., and Henzinger, M.R. Improved algorithms for topic distillation in hyperlinked environments. In B. Croft, A. Moffat, C.J. van Rijsbergen, R. Wilkinson, and J. Zobel (eds.), *Proceedings of the Twenty-First International ACM SIGIR Conference on Research and Development in Information Retrieval.* New York: ACM Press, 1998, pp. 104–111.

2. Blum, C., and Roli, A. Metaheuristics in combinatorial optimization: Overview and conceptual comparison. *ACM Computing Surveys, 35,* 3 (2003), 268–308.

3. Bowman, C.M.; Danzig, P.B.; Manber, U.; and Schwartz, F. Scalable Internet resource discovery: Research problems and approaches. *Communications of the ACM, 37,* 8 (1994), 98–107.

4. Bui, T.N., and Moon, B.R. Genetic algorithm and graph partitioning. *IEEE Transactions on Computers, 45,* 7 (1996), 841–855.

5. Chen, H., and Lynch, K.J. Automatic construction of networks of concepts characterizing document databases. *IEEE Transactions on Systems, Man, and Cybernetics, 22,* 5 (1992), 885–902.

6. Chen, H.; Schuffels, C.; and Orwig, R. Internet categorization and search: A self-organizing approach. *Journal of Visual Communication and Image Representation, 7,* 1 (1996), 88–102.

7. Chen, H.; Chung, Y.; Ramsey, M.; and Yang, C. A smart itsy bitsy spider for the Web. *Journal of the American Society for Information Science, 49,* 7 (1998), 604–618.

8. Chen, H.; Fan, H.; Chau, M.; and Zeng, D. MetaSpider: Meta-searching and categorization on the Web. *Journal of the American Society for Information Science and Technology, 52,* 13 (2001), 1134–1147.

9. Cutting, D.R.; Karger, D.R.; Pederson, J.O.; and Tukey, J.W. Scatter/gather: A cluster-based approach to browsing large document collections. In E. Fox, N. Belkin, P. Ingwerson, and A.M. Pejtersen (eds.), *Proceedings of the Fifteenth Annual International ACM SIGIR Conference on Research and Development in Information Retrieval.* New York: ACM Press, 1992, pp. 318–329.

10. Davies, P.H.J. Intelligence, information technology, and information warfare. *Annual Review of Information Science and Technology, 36* (2002), 313–352.

11. Eom, S.B., and Farris, R.S. The contributions of organizational science to the development of decision support systems research subspecialties. *Journal of the American Society for Information Science, 47,* 12 (1996), 941–952.

12. Ericsson, K.A., and Simon, H.A. *Protocol Analysis: Verbal Reports as Data.* Cambridge, MA: MIT Press, 1993.

13. Flake, G.W.; Lawrence, S.; Giles, C.L.; and Coetzee, F.M. Self-organization and identification of Web communities. *IEEE Computer, 25,* 3 (2002), 66–71.

14. Fuld, L.M.; Singh, A.; Rothwell, K.; and Kim, J. *Intelligence Software Report™ 2003: Leveraging the Web.* Cambridge, MA: Fuld & Company, 2003.

15. Futures-Group Ostriches & Eagles. The Futures Group Articles, Washington, DC, 1997 (available at www.futuresgroup.com).

16. Gloor, P.A. CYBERMAP: Yet another way of navigating in hyperspace. In *Proceedings of the Third Annual ACM Conference on Hypertext.* New York: ACM Press, 1991, pp. 107–121.

17. Gordon, M. Probabilistic and genetic algorithms for document retrieval. *Communications of the ACM, 31,* 10 (1988), 1208–1218.

18. Grabmeier, J., and Rudolph, A. Techniques of cluster algorithms in data mining. *Data Mining and Knowledge Discovery, 6,* 4 (2002), 303–360.

19. Hartigan, J.A. Statistical theory in clustering. *Journal of Classification, 2* (1985), 63–76.

20. He, X.; Ding, C.; Zha, H.; and Simon, H. Automatic topic identification using Webpage clustering. In X. Wu, N. Cercone, T.Y. Lin, J. Gehrke, C. Clifton, R. Kotagiri, N. Zhong, and X. Hu (eds.), *Proceedings of the 2001 IEEE International Conference on Data Mining.* Los Alamitos, CA: IEEE Computer Society Press, 2001, pp. 195–202.

21. He, Y., and Hui, S.C. Mining a Web citation database for author co-citation analysis. *Information Processing and Management, 38,* 4 (2002), 491–508.

22. Kanai, H., and Hakozaki, K. A browsing system for a database using visualization of user preferences. In E. Banissi (ed.), *Proceedings of IEEE International Conference on Information Visualization.* Los Alamitos, CA: IEEE Computer Society Press, 2000, pp. 277–282.

23. Kealy, W.A. Knowledge maps and their use in computer-based collaborative learning. *Journal of Educational Computing Research, 25,* 4 (2001), 325–349.

24. Lin, X. Map displays for information retrieval. *Journal of the American Society for Information Science, 48,* 1 (1997), 40–54.

25. Lyman, P., and Varian, H. How much information. University of California, Berkeley, 2000 (available at www.sims.berkeley.edu/how-much-info).

26. Mowshowitz, A., and Kawaguchi, A. Bias on the Web. *Communications of the ACM, 45,* 9 (2002), 56–60.

27. Myers, J., and Well, A. *Research Design and Statistical Analysis.* Hillsdale, NJ: Lawrence Erlbaum, 1995.

28. Nick, Z., and Themis, P. Web search using a genetic algorithm. *IEEE Internet Computing, 16,* 2 (2001), 18–26.

29. Nunamaker, J.F.; Romano, N.C.; and Briggs, R.O. A framework for collaboration and knowledge management. In R.H. Sprague Jr. (ed.), *Proceedings of the Thirty-Fourth Annual Hawaii International Conference on System Sciences.* Los Alamitos, CA: IEEE Computer Society Press, 2001, pp. 461–472 (available at www.hicss.hawaii.edu/hicss_34/apahome34.htm).

30. Palmer, C.; Pesenti, J.; Valdes-Perez, R.; Christel, M.; Hauptmann, A.; Ng, D.; and Wactlar, H. Demonstration of hierarchical document clustering of digital library retrieval results. In E. Fox and C.L. Borgman (eds.), *Proceedings of the First ACM/IEEE Joint Conference on Digital Libraries.* New York: ACM Press, 2001, p. 451.

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

32. Selberg, E., and Etzioni, O. The MetaCrawler architecture for resource aggregation on the Web. *IEEE Expert, 12,* 1 (1997), 8–14.

33. Shi, J., and Malik, J. Normalized cuts and image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 22,* 8 (2000), 888–905.

34. Shneiderman, B. The eyes have it: A task by data type taxonomy for information visualizations. In *Proceedings of IEEE Symposium on Visual Languages.* Los Alamitos, CA: IEEE Computer Society Press, 1996, pp. 336–343.

35. Spence, R. *Information Visualization.* New York: ACM Press, 2001.

36. Tolle, K.M., and Chen, H. Comparing noun phrasing techniques for use with medical digital library tools. *Journal of the American Society for Information Science (Special Issue on Digital Libraries), 51,* 4 (2000), 352–370.

37. Torgerson, W.S. Multidimensional scaling: I. Theory and method. *Psychometrika, 17,* 4 (1952), 401–419.

38. Wise, J.A.; Thoma, J.J.; Pennock, K.; Lantrip, D.; Pottier, M.; Schur, A.; and Crow, V. Visualizing the non-visual: Spatial analysis and interaction with information from text documents. In *Proceedings of IEEE Symposium on Information Visualization.* Los Alamitos, CA: IEEE Computer Society Press, 1995, pp. 51–58.

39. Wu, Z., and Leahy, R. An optimal graph theoretic approach to data clustering: Theory and its application to image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 15,* 11 (1993), 1101–1113.

40. Young, F.W. *Multidimensional Scaling: History, Theory, and Applications,* ed. R.M. Hamer. Hillsdale, NJ: Lawrence Erlbaum, 1987.