Knowledgescapes: A Probabilistic Model for Mining Tacit Knowledge for Information Retrieval

Heyning Adrian Cheng

Abstract

Most existing information retrieval systems attempt to analyze the content and structural properties of documents, without explicitly considering the actual information needs of users. However, a vast amount of task-specific knowledge is implicitly encoded in the behavior of users accessing an online information collection. In this paper we present Knowledgescapes, a novel probabilistic framework for supporting general-purpose information retrieval by mining this tacit knowledge from web server access logs. We formulate a Bayesian probabilistic model for reasoning about the short-term information needs of users and use this model to support the dynamic reranking of query results based on the user's recent browsing history. We discuss our experiences with a realistic prototype search engine based on this model that we developed for users of the Berkeley Digital Library document collection. We analyze the capabilities and limitations of the Knowledgescapes model and identify several avenues for future research on the problem of mining implicit knowledge to support information retrieval applications.

Acknowledgement: This work was supported in part by the Defense Advanced Research Projects Agency.

1. Introduction

People searching the universe of online content today have a wide range of search and recommendation tools to choose from. Many systems provide indices of all the words in a text document; others compute estimates of the importance of Web sites; while still others return the preferences of other individuals having similar profiles. However, few if any of today's general purpose systems are designed to provide recommendations tailored to fit the user's particular information need at the time of the search. It is a basic premise of our research that search engines are often used to satisfy short-term, task-specific information needs, and that these needs are often much more specific than most common keyword queries. For example, a user who issues the query "airlines" is likely to be searching for a flight between specific cities that meets other requirements such as time and cost. Thus, it is our hypothesis that a model of information retrieval that explicitly reifies and analyzes these immediate information needs can provide better support for information-intensive tasks than existing paradigms.

We propose a simple Bayesian network model describing the probabilistic causal relationship between a user's information need, query, and browsing behavior. Using this model, the probability distribution over information needs can be updated to support dynamic reranking of documents based on the user's recent browsing history. Since our techniques are not based on specialized artificial intelligence or content analysis techniques, they are highly versatile and can be readily applied to non-textual as well as textual content collections. The data required to train our model for a particular content collection can easily be extracted from standard Web server access logs. Furthermore, our techniques do not require explicit input from the user, either for training or for runtime query processing, but are based on the observation that human knowledge is tacitly encoded in the actions of individuals using a collection. The focus of our research is the development of algorithms to mine this tacit knowledge and leverage it to improve general-purpose information access. In order to evaluate the efficacy of this approach in a realistic setting, we also developed a working prototype of the Knowledgescapes query engine for the Berkeley Digital Library document collection and deployed it as an online service for users of this collection. Based on our experience with this experimental application, we identify several longer-term research problems that will need to be addressed before the full potential of tacit knowledge mining approaches to information retrieval can be realized.

The rest of this paper is organized as follows. Section 2 outlines previous work in the field and discusses the advantages of our approach over existing paradigms. Section 3 presents the key ideas in our research and formulates the Knowledgescapes model for information retrieval. Section 4 describes the design and implementation of the Knowledgescapes search engine prototype for the Berkeley Digital Library document collection. Section 5 covers the evaluation of Knowledgescapes, including the results of user studies involving two different test prototypes of the system. Section 6 discusses our findings and examines the key technical and non-technical challenges involved in developing a practical usage-based search engine. Section 7 explores possible directions for extending our work to support a range of information-intensive applications. Section 8 offers concluding remarks.

2. Previous Work

Existing information retrieval systems generally follow one of several distinct paradigms, each of which has certain advantages and disadvantages. The most popular approach by far is keyword indexing of documents. While it is simple and effective for some domains, this model suffers from several major limitations. Since relevance scores are computed solely on the basis of the static properties of documents without incorporating any human judgment, they are often poor measures of the quality and utility of documents and are easily manipulated by authors. Since they do not consider the semantic context of the user's query, keyword-based methods cannot narrow the set of recommended documents to better fit the user's needs. This is especially problematic because many users have difficulty synthesizing queries that accurately describe their specific needs, and instead resort to vague and simplistic queries; well-known statistics show that a large fraction of text queries consist of a single word. Furthermore, the utility of keyword indexing is limited to text media; non-textual content such as images cannot be indexed unless it has been manually labeled with keywords.

Collaborative filtering, on the other hand, attempts to leverage the collective judgments of other users in evaluating the items of interest. For example, a book recommendation service might rank books based on the average ratings they received from all other users, or alternatively from other users who have historically preferred the same books as the current user. Most such systems require users to provide explicit ratings for the items in question. This additional effort often discourages people from using such systems, especially if the existing user base is insufficient to provide useful recommendations; as a result, the initial bootstrapping of these systems is often a daunting obstacle [5, 13]. Alternatively, collaborative filtering systems can also derive implicit ratings of information objects from historical records of the user's interaction with the content. The duration of time that a user spends browsing a document can be used as an indicator of the user's interest in it; experiments by Konstan et al. [11] showed a strong correlation between reading time and subjective ratings for Internet newsgroup articles. Link-based Web search engines such as Google [6] are another example of tacit knowledge mining. The basic assumption here is that the inclusion of a hyperlink on a Web page reflects the author's judgment of the quality and value of the destination page; the function of the search engine is then to aggregate the distributed knowledge implicit in the graph structure of the Web to compute scores for Web pages based on their connectivity.

In the Knowledgescapes model, we employ a different kind of tacit knowledge mining that leverages the collective knowledge of the *users*, rather than the authors, of a document collection. This practical knowledge can be mined from usage data such as is available from Web server access logs. The advantages of mining usage data are several. Usage data is far more abundant than link data; detailed user data is more likely to be available for individual pages in a Web site, whereas incoming hyperlinks tend to point to the site's home page. Usage data mining is also more versatile since it does not require a rich hyperlink structure and thus can be applied to large document collections with flat organizational structures such as digital libraries. Usage data is more specifically contextualized; the history of a user's information access behavior over a short period of time can be viewed in the context of an individual pursuing a specific information need or performing a particular task. Finally, usage information more directly addresses the problem of evaluating the utility of documents relative to real user needs, as authors cannot usually anticipate all of the possible uses for their documents.

Several experimental information systems have been designed to leverage usage data. Lau and Horvitz [12] apply a Bayesian network approach to infer informational goals from Web queries issued by users. Their model is based on a fixed, manually defined hierarchy of information needs and does not consider the interaction of users with the actual documents returned in response to queries. Horvitz et al. [8] use Bayesian inference to derive the probability distribution over the real-time user needs and computes the expected utilites of various courses of action. Their system is intended for providing assistance for users of particular software applications and not for supporting general information access; furthermore, it requires a substantial amount of manual knowledge engineering of a highly domain-specific nature. Pennock and Horvitz [14] present a movie recommendation service that uses Bayesian inference to match the current user's personality with the profiles of previous users, treating each user's long-term profile as an information need; users must provide explicit ratings of the movies to establish profiles. Fu et al. [4] track a user's Web navigation history and apply data mining techniques based on associative rules to extract tacit knowledge contained in the history. They do not track the amount of time spent browsing each page and consider only long-term user behavior. Our Knowledgescapes paradigm represents an improvement over existing systems in several important respects. We apply a Bayesian model to infer the short-term, real-time needs of users, which allows them to derive immediate benefits from the system without establishing a long-term profile or history. Our model allows for an unrestricted, extensible set of information needs and does not require a manually constructed knowledge base. Knowledgescapes also does not require users to provide explicit profiles or ratings; user preferences are implicitly obtained by tracking all interactions with the content collection, including both browsing navigation and queries.

3. Probabilistic Model for Mining Usage Data

In this section we formulate the Knowledgescapes model for information retrieval. Section 3.1 provides a overview of the data structures; section 3.2 discusses the fundamental assumptions on which the model is based and attempts to dissect the meaning of information needs; section 3.3 formulates the probabilistic algorithm used to infer the user's current information need; and section 3.4 describes heuristics used for computing document scores. Scalability issues, including the time and space complexity of the Knowledgescapes model, are discussed in Section 7.2.

3.1. Overview

Knowledgescapes uses a Bayesian probabilistic model to infer the user's current information need and generate document recommendations accordingly. This model is used both for computing initial search results for ad-hoc queries, and for dynamically recomputing search results based on the user's subsequent browsing behavior. The basic entities in this model are queries, information needs, and documents. More intuitively, Knowledgescapes can be represented as a tripartite graph containing nodes for queries, needs, and documents, as shown in Figure 1. Let Q, N, and D denote the sets of query strings, information needs, and documents in this graph, respectively. In this graph, a query node is linked with equal weight to all information needs for which the particular query string was submitted; each information need node is in turn linked to all the documents visited during that need, with the link weights reflecting the estimated utility of the document for satisfying the need. In this discussion, we assume that we have a database containing the following data structures, all of which can easily be extracted from the historical web server logs for the collection of interest. These include the sets Q, N, and D, along with data structures which define the historical relationships among these entities. The latter includes a weighted matrix $\boldsymbol{T}$ where $\boldsymbol{T_{ij}}$ represents the value of document d_j with respect to information need n_i; and a boolean matrix $\boldsymbol{Q}$ where $\boldsymbol{Q_{ij}} = 1$ if query string qi is associated with information need nj, and $\boldsymbol{Q_{ij}} = 0$ otherwise.

3.2. Reading Times and Information Needs

It is a fundamental assumption of the model that the time spent reading a document is an indication of the perceived utility of the document. Thus, reading time is used as a proxy for the document valuations in $\boldsymbol{T}$. In practice, reading time depends on many factors including the difficulty of finding the desired information within the document, the bandwidth of the user's Internet connection, and the constraints of the user's work environment [5]. However, we believe that our assumption is generally valid for reasonably long reading times. From an economic perspective, reading time could also be thought of as a measure of the value of the document in terms of the amount of time the reader is willing to invest. In practice, reading time is likely to be a highly inexact measure of utility, since the very nature of the browsing process is such that people generally do not know what information is contained within a document until they have browsed it.

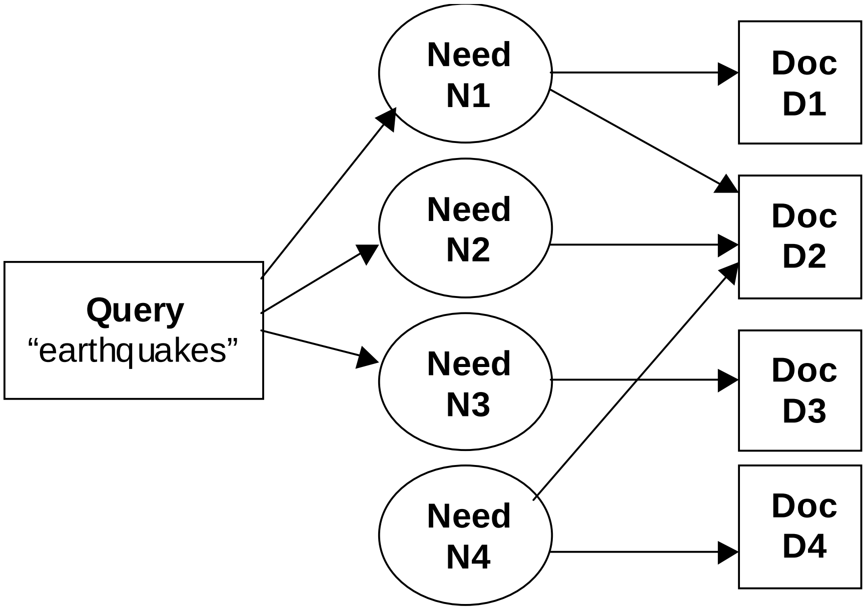

Figure 1. Knowledgescapes as a tripartite graph.

In order to infer the user’s information need based on his recent document access patterns, we must also make some assumptions about the mathematical form of the probability distribution of the reading time for a document D_j given a particular information need N_i. Since web browsing is a potentially complex behavioral process whose characteristics depend largely on the nature of the particular information need, such assumptions are necessarily based on empirical observations rather than on provable axioms or principles. We partitioned the Web server logs for the UC Berkeley Digital Library document collection into information needs and profiled the distribution of need-document reading times. It is impossible to determine from these logs which information need instances are in fact semantically equivalent and which instances are different. Furthermore, if we were assume that each need instance is unique, we would not be able to compile any meaningful statistics about the distribution $P(d_j|n_i)$, since only a single reading time observation is available for any given need instance and document pair. Therefore, we can only provide approximate validation of our mathematical model by profiling the distribution of reading times averaged over multiple information need instances. We observed that document reading times overall followed a heavy-tailed distribution with a mean of 287 seconds, a median of 115 seconds, and a standard deviation of 547 seconds. The log of the reading times, however, followed a roughly Gaussian distribution with a mean of L_{mean} = 4.78, a median of 4.74, and a standard deviation of σ_{mean} = 1.37. This distribution is shown in Figure 2. The peak at x=4.7 is an artifact resulting from the truncation of individual page reading times at 115 seconds during preprocessing of the Web server logs; while the peak at x=2.2 is the result of rounding very short reading times up to the minimum of 5 seconds. We then examined the distribution of reading times observed for specific documents in response to specific query strings, computing a separate distribution for each of the most commonly submitted queries in the server logs. In most of these cases, we found a heavy-tailed distribution of reading times that was approximately log-normal; the width of the normal distribution in log space averaged 1.25. Also, some of the query strings specified particular keywords for full-text search, while other queries were far more vague, identifying only a general category of documents. We repeated the previous experiment, considering reading times for documents in response to specific keyword queries only, and found that the width of the distribution decreased to 1.13. As we intuitively expected, the probability distribution becomes narrower as we consider more specific sets of information needs. Thus, in our model we assume that $ln\ P(d_j|n_i)$ follows a normal distribution with mean $ln\ (t_{ij})$ and standard deviation σ_d = 1.1, where t_{ij} is the observed reading time for document d_j by need n_i according to the server logs.

The reification of information needs in this model is an open research topic. The potential range of particular needs that users may have is nearly unlimited, and we must devise heuristics for partitioning the chronological log of a user's document accesses into contiguous segments corresponding to coherent information needs. This is a domain and application specific problem, as user practices may vary considerably depending on the nature of the tasks involved, the content of the collection, and the design of the collection interface. As a general rule of thumb, we assume that a new information need begins whenever a user enters a query that is different from his previous query, or whenever a sufficiently long time has passed since his previous access to the collection. Thus, an information need may represent either a sequence of accesses in response to a particular query, or simply a temporally localized period of browsing. This model of user behavior is far from perfect, as users often generalize or specialize their queries while attempting to fill particular needs [12]. The determination of the maximum length and/or idle time allowed for a single information need also necessitates an inherent tradeoff between the semantic coherence of each need and the richness of the data for a given need.

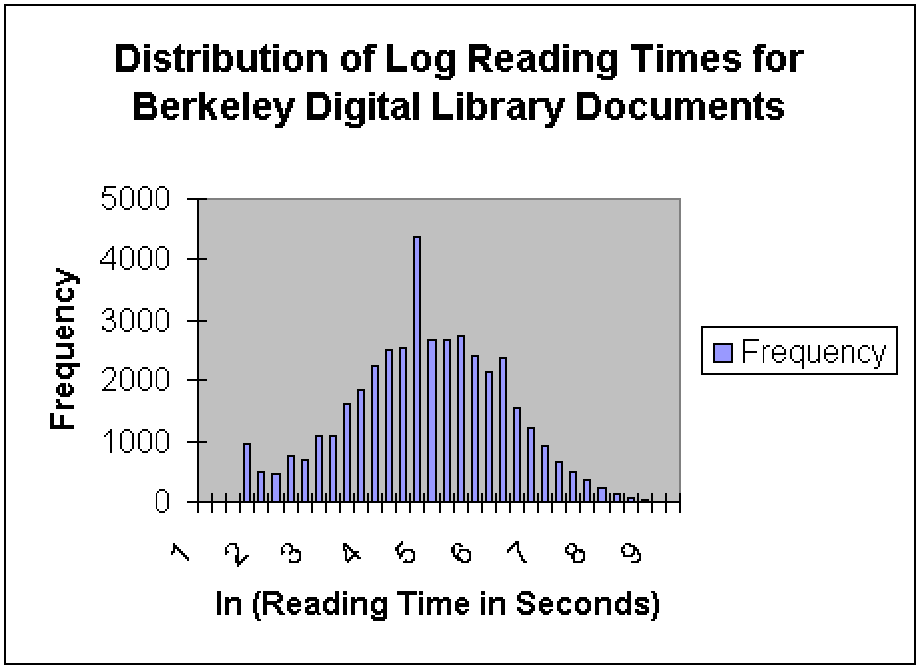

Figure 2. Distribution of log reading times for the Berkeley Digital Library document collection. Data includes Web server logs from June 1998 to December 1999. The peaks at x=4.7 and x=2.2 are artifacts.

3.3. Information Need Inference

Let the random variables *n* and *q* represent the current information need and query string, respectively. Let D_c denote the set of documents $\{ d_1, d_2, \ldots , d_m \}$ recently browsed by the active user. Let T_c denote a set of random variables $t_1, t_2, \ldots , t_m$ representing the amount of time that the active user has spent reading documents $d_1, d_2, \ldots , d_m$, respectively, since the beginning of the current information need. We assume that the current information need *n* is one of the historical information needs in the database, and we model *n* as a probability distribution over *N*. In Knowledgescapes, we model the information seeking process as a Bayesian network (shown in Figure 3) in which *n* causally influences both *q* and T_c. We assume that *q* is conditionally independent of T_c given *n*, and vice versa. Given *q* and T_c as evidence, we can infer *n* as follows:

$$Prob(n \mid q,Tc) = k * Prob(q|n) * Prob(T_c|n), \text{ where } k \text{ is a constant and } \Xi_i \; Prob(n_i) = 1.$$

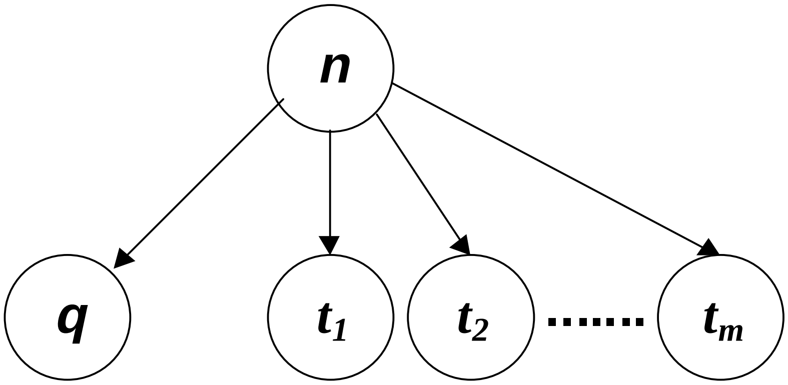

Figure 3. The Knowledgescapes model as a Bayesian network. The user's information need causally influences both the query string submitted and the reading times for different documents.

Note that the set T_c is non-empty only when dynamic reranking is performed after the user has browsed some documents; for computing an initial result set the latter equation reduces to $Prob(n \mid q, Tc) = k * Prob(q|n)$. For dynamic reranking, the reading times $t_1, t_2, \ldots , t_m$ are obtained by logging the user's requests for documents in the collection in real time. Once a query has been initially processed, the user may request dynamically updated search results at any time during the browsing process. Therefore, when dynamic reranking is invoked, we cannot assume that the user has finished browsing all the documents in the set $D_{c,}$ or that $t_1, t_2, \ldots , t_m$ represent final reading times for the current information need. Instead, we are primarily interested in the relative differences among these reading times, as users requesting reranked search results are typically interested in finding documents similar to one or more of the documents they have just visited. Thus, we normalize $t_1, t_2, \ldots , t_m$ according to the following heuristic. Let L_c denote the mean log reading time for the current information need over the set of documents D_c, i.e. $L_c = 1/m * [\sum_{(1<= i <=m)\ ln}\ t_i\ ]$. If $L_c >= L_{mean}$, no adjustment is needed, since long absolute reading times generally indicate significant user interest and effort. If $L_c < L_{mean}$, we multiply each reading time $t_1, t_2, \ldots , t_m$ by $exp(L_{mean}$-$L_c)$ to raise the mean log reading time for the current information need to match the overall historical mean.

The real difficulty lies in the estimation of the parameters $Prob(q|n)$ and $Prob(T_c|n)$. If n is associated with the query q, then we could reasonably assume that $Prob(q|n) = 1$. However, if n is not associated with the query q, we cannot assume that the probability of a user with need n issuing query q is zero. This is because some needs in the database are not explicitly associated with any query, and also because the query space is multi-dimensional such that two queries may be orthogonal (for example, a full-text keyword query and a query on documents by a certain author) and people may select several different queries to pursue the same information need. In cases where n is not associated with q, we hypothesize that $Prob(q|n)$ should reflect the likelihood of a user with any arbitrary information need submitting query q. In general, queries containing rare keywords tend to be more specific and better able to discriminate between relevant and irrelevant documents; therefore, such queries are typically less likely to be entered by users with broad or vague information needs. For our model, we assumed that $Prob(q|n) = k*IDF(q)$, where k=0.2 and $IDF(q)$ is the sum of the IDFs (within the particular collection) of the individual terms contained in q. In effect, we give greater weight to query string matches for more salient query strings.

Similarly, we must make several simplifying assumptions in order to tractably compute estimates for $Prob(T_c|n)$. We assume that the reading time probability distributions for different documents are mutually conditionally independent given information need n, and therefore $Prob(T_c|n) = \prod_{(\forall\ j,\ Dj \in Dc)} Prob(t_j|n)$. This conditional independence assumption is inherently imperfect even when the information need is precisely specified. This is because the utility of a document from the user's perspective is often influenced by the set of documents already browsed during the current information need; for example, if the user has already found the desired information by perusing document d_j, he would be less likely to ascribe a high utility to a nearly identical document d_k. Likewise, if one has previously read d_k in its entirety, one might be quite unlikely to invest significant time in browsing d_k again. Nevertheless, the independence assumption is unavoidable because Web server logs do not contain enough data to estimate the exponential number of joint probability distributions, not to mention that the data storage requirements for the joint distribution would be intractable for any realistic document collection.

If need n_i is associated with document d_j, we model $Prob(t_j| n_i)$ as a log-normal distribution with mean and standard deviation set to $ln\ (t_{ij})$ and σ_d, respectively. If n_i is not associated with t_j, we assume a bi-modal distribution,

$$Prob(t_j| n_i) = k_1{*}Prob_{mean}(tj) + k_2{*}Prob_{short}(tj),$$

where $Prob_{mean}(tj)$ and $Prob_{short}(tj)$ are unit log-normal distributions with means of L_{mean} and L_{short} respectively and standard deviations of σ_{mean}. In our model, the constants k_1 and k_2 are set to 0.08 and 0.02 respectively; L_{mean} and L_{short} are set to 4.78 and ln(5.0) to reflect the average and minimum recorded log reading times respectively. The reasoning behind this bimodal model is that the lack of association between n_i and d_j in the Knowledgescapes graph typically has one of several explanations: 1) d_j was not browsed because it was not listed near the top of the search results when the user queried the existing full-text search engine; 2) the user saw d_j listed in the search results but decided that the title of the document was irrelevant to his information need; and 3) the user had already browsed d_j while pursuing an earlier information need and thus chose not to revisit the document. Cases 1) and 3) cannot generally be interpreted as either a positive or negative evaluation of the utility of $d_{j,}$ while case 2) implies a negative judgment. For cases 1) and 3), our model assumes that $Prob(t_j|n_i)$ is equivalent to the overall distribution of reading times in the historical server logs ($Prob_{mean}(tj)$). For case 2) it assumes that ln $Prob(t_j| n_i)$ follows a normal distribution of the same width as the overall distribution, but with the mean shifted downwards to reflect the minimum recorded document reading time of 5 seconds ($Prob_{short}(tj)$). In practice, we believe that case 1) is far more common than cases 2) and 3), since the existing full-text search engine often returns far more search results than people are interested in browsing. In other words, the lack of association between a need and a document most often does not imply a discernable judgment of the document, and occasionally is evidence of a negative evaluation, hence the relative values of k_1 and k_2. Note that the sum of k_1 and k_2 is much less than unity; this indicates that a user with need N_i is typically far more likely to browse a document that is historically associated with n_i than one that has no historical association.

3.4. Computation of Document Scores

Knowledgescapes then computes a score s_j for each document d_j based on the posterior probability distribution over the set of information needs, $Prob(n| q,Tc)$. We will denote the posterior probability of need n_i by $Prob(n_i)$. There are a number of different heuristics that can be used for this computation; in selecting an appropriate heuristic, we must remember that documents in the query result set are ranked according to these scores, and that the practical utility of a search engine is primarily a function of the ordinal ranking of the documents and not the absolute numerical scores. One naïve approach would be to score documents based on their expected reading times, i.e. $s_j = \Sigma_{(ni\in N)}\ t_{ij} * Prob(n_i)$. While this heuristic has the intuitive interpretation of scoring documents according to their expected "economic utility" to the end user, it suffers from several apparent shortcomings. First, users often fail to browse documents that are highly relevant to their information needs because they were not highly ranked by the existing content-based search engine. According to this formula, however, a document that has not been frequently browsed in the past will tend to receive low scores, even if everyone who has browsed the document has found it to be very useful. Conversely, a document that is typically rejected quickly upon examination may receive a relatively high score if it is listed near the beginning of the result sets for many Cheshire II queries. In addition, document reading times follow a heavy-tailed distribution with the log of the reading time exhibiting approximately Gaussian noise; thus a weighted average of reading times is very sensitive to noise, such that a single information need with an unusually large reading time might cause a document of questionable relevance to be ranked very highly.

In order to address these issues, we devised a different scoring heuristic:

$$
\begin{aligned}
s_j &= 1/r_j * \Sigma_{(ni \in N)} [\, (\ln t_{ij} + b) * Prob(n_i).\,], \text{ where} \\
r_j &= \Sigma_{(\forall\, i,\, tij>0)}\, Prob(n_i)\ , \text{ and} \\
b &= Min[\ln (f_1/f_2)\ \ ,1]\ ,
\end{aligned}
$$

where f_1 is the average posterior probability of the needs that are linked to document d_j, and f_2 is the average posterior probability of all needs with nonzero posterior probabilities. The motivation for this heuristic is as follows. We use log reading times rather than absolute reading times to compute scores, in order to reflect the log-normal distribution of reading times. To reduce the distortion caused by Cheshire II rank ordering bias, we consider only the set of information needs that are historically linked to d_j when computing s_j; this is done by dividing by the aggregate posterior probability r_j of this set of needs. Thus, a document is scored based on the amount of time invested by people with relevant information needs who have actually read the document, rather than the number of people who have initially browsed it. One disadvantage of this approach is that a document d_j that is associated with a set of needs with very small posterior probabilities may be assigned an artificially high score. To correct for this potential bias, we added an adjustment factor *b* that reflects the posterior probabilities of the information needs that are linked to d_j relative to the average posterior probabilities of all needs considered to be potentially relevant. This effectively causes higher scores to be assigned to documents that are strongly associated with historical information needs which are more likely to match the active user's need. In any case, the practical utility and applicability of scoring heuristics can only be verified through human-centered testing in a realistic setting.

4. Knowledgescapes System Design and Implementation

4.1. Experimental Testbed

We implemented a working prototype of the Knowledgescapes search engine for the Berkeley Digital Library online document collection [1], consisting of 2513 documents of varying lengths on a broad range of topics related to the environment. The collection contained roughly 280,000 pages of text requiring 67 MB of storage; documents consisted of mostly text with some embedded images and maps. The collection contained several distinct types of Web pages: 1) individual pages of documents (scanned images in GIF format, as well as OCR-generated text in ASCII format); 2) OCR texts of entire documents; 3) cover pages describing individual documents; and 4) maps associated with the documents. Users are allowed to choose between scanned image and OCR document formats; typically a user would access a document via its cover page and browse its content by downloading scanned images of the desired pages. Most user queries are processed by Cheshire II, a probabilistic full-text search engine based on logistic regression analysis [2, 3]. Cheshire II accepts conjunctive queries on several fields including full-text keywords, title keywords, author, document type, and project ID; it provides relevance scores both for entire documents and for specific pages within documents. This is an good testbed for our research because of the availability of detailed page-by-page document access records; the prevalence of large documents and relatively long reading times is also conducive to a high signal to noise ratio in the usage data. We obtained the complete web server logs for this collection spanning an 18-month period from June 1998 to December 1999.

4.2. Pre-processing of historical usage data

The duration and structure of information needs are highly variable, and undoubtedly depend on a number of factors including the type of information sought, the quantity and quality of information available on the requested subject, and the user's particular task at the time. As a rough heuristic, an information need boundary is drawn if at least 60 minutes has elapsed between consecutive downloads by a client host. Imposing a short cutoff time would tend to split coherent information needs into several separate nodes, resulting in an artificially sparse graph of document-need and document-document associations. On the other hand, using a long cutoff time increases the risk of treating several distinct information needs as a single unit, causing associations to be inferred between unrelated documents and adding noise to the database. We examine the tradeoff further in Section 5.1. The Knowledgescapes database for our prototype contained 3426 distinct query strings, 2273 documents, and 26,457 information needs, and required about 1.2 MB of storage. While the size of the database is very small relative to the quantity of text in the document collection, it is not a comprehensive index of the collection and is not intended to be used as such. For usage-based models such as our, the distribution of the data in document space and in query space is inevitably highly uneven, containing far more information on frequently accessed documents and common query topics than on rarely requested information objects and topics.

We computed the reading time for each page request as the elapsed time between successive HTTP requests from the same DNS or IP client address in the web server logs. To reduce the amount of noise in the data set, we performed outlier filtering using the following heuristics. Reading times exceeding a threshold for the appropriate page type (5 minutes for document covers and individual GIF pages) were considered to be anomalous and discarded. We then computed the mean and standard deviation of the reading times for each page type and truncated reading times to a maximum of two standard deviations above the mean time for the appropriate page type (115 seconds for individual pages). The weights for the need-document matrix **T** were determined by setting $\mathbf{T}_{\mathbf{ij}}$ equal to the log of the sum of the filtered reading times for all accesses to document d_j for information need n_i. Very short document reading times were rounded up to a minimum of 5 seconds to account for differences in the network connection bandwidth available to different users and the download latencies incurred.

4.3. Query Processing Techniques

To increase the density of the query data, all logged queries were converted to a canonical form consisting of three fields – the author, the union of full-text and title query terms, and the document type. Knowledgescapes only stores information on keywords that have appeared in queries submitted by users in the past, generally a far smaller set of terms than that contained in the full text of the document collection. In fact, the query database for the environmental document collection contains only 2375 author and full-text terms, whereas a full-text index contains hundreds of thousands of terms. For comparison, the Cheshire full-text index for this collection is over 100 MB in size, while the Knowledgescapes database occupies 1.2 MB with query information comprising only a small fraction. The Knowledgescapes model is also susceptible to the phenomenon of topical drift that affects most non-content based retrieval paradigms. For practical purposes, therefore, Knowledgescapes is better suited for ranking a set of somewhat relevant documents according to their expected value for a particular information need than for evaluating the topical relevance of the documents. Thus, in our latest implementation, we use the Cheshire II full-text search engine to filter the results for keyword relevance, so that Knowledgescapes is used as an enhancement to the existing system, rather than as a monolithic information retrieval platform.

Users may also enter queries that are not found in the Knowledgescapes database, or which retrieve an insufficient number of documents. For many collections including the WWW, existing content-based search facilities may be queried to retrieve an initial set of weighted or ranked document matches; after the user has browsed some documents from this set, it is often possible to rerank the documents using the Knowledgescapes algorithm. Thus, we can use the content-based search facility as the default information retrieval mechanism to bootstrap the Knowledgescapes system, and allow performance to improve incrementally as more usage data becomes available. This approach is particularly advantageous in domains where the usage data is relatively sparse. In many cases, users may enter query strings that do not exactly match any query strings in the historical database, but which contain one or more query terms in common with some query strings in the database. For such partial matches, the user's query is represented in Knowledgescapes as a weighted vector of partially matching query nodes; any information needs associated with any of these query nodes would then be considered when generating the initial set of search results. The specific algorithm used for computing query match weights is described in section 5.3.

4.4. Server Implementation

The Knowledgescapes database is periodically regenerated offline from the updated server logs and is stored on the server. We implemented a Web-based interface that allows users to navigate the search system and the content of the document collection; the server is notified of all queries and document requests received through this interface. The interface consists mostly of dynamically generated HTML pages. While all navigation links point to URLs on our server's Web site, the actual document content is transferred directly from the Digital Library Web server. Our document browsing interface allows users to request refined query results at any time, regardless of whether or not a query has been previously issued by the client. The server processes all Knowledgescapes queries; query results from Knowledgescapes and from Cheshire II are aggregated at the server before being forwarded to the client. For each distinct client IP address, the server stores a separate log listing in chronological order all of the client's queries and data requests since the beginning of the current information need. The Knowledgescapes server was initially implemented as a set of several Java applications invoked using CGI, and was later re-implemented as a Java Servlet for efficiency reasons.

We observed that the usage data for our testbed is very unevenly distributed with some queries occurring quite frequently while some keyword queries on common environmental terms are not represented at all. The fact that the data is sparse over most of the query space is not surprising; we expect that this is typical of most application domains, as a small fraction of all distinct queries is likely to account for a large proportion of query requests. From our experience, most full-text keyword searches retrieve at most partial matches in the Knowledgescapes query database, and a significant fraction of searches require the retrieval of initial results from Cheshire II. Once the user has browsed one or two documents, however, Knowledgescapes is generally able to contribute to the dynamic reranking process. Thus it is likely that for less common queries, the value of the Knowledgescapes paradigm lies primarily in its dynamic reranking capabilities.

5. Experiments and Results

Our performance evaluation of the Knowledgescapes model involved a combination of statistical computations for internal validation and empirical human-centered evaluation in a realistic user environment. We believe that practical and experimental evaluation would be potentially much more meaningful and informative for our research purposes, and thus we have applied the greater share of our efforts in this direction. Human-centered evaluation turned out to be one of the more difficult aspects of this project, however, since standardized evaluation procedures such as TREC that are commonly used to benchmark traditional information retrieval engines are far less established in the relatively young field of tacit-knowledge-based retrieval systems. Our evaluation procedures and results are discussed in detail below.

5.1. Internal Consistency of the Model

We first attempted to validate the internal consistency of the Knowledgescapes model, assuming for the time being that reading time is a good indicator of the perceived utility of a document, as we had hypothesized. For these experiments, we are not concerned with the correctness of this assumption, but only with the ability of the model to predict reading times for documents given the current query, the browsing history for the current information need, and the contents of the historical usage database. We generated the Knowledgescapes database for the Berkeley Digital Library document collection as described in Section 4, and performed cross-validation on the weights in the information need-document matrix. Each experiment consisted of five iterations through the database; on each iteration, we removed 20% of the information needs from the database and predicted the reading time weights for the corresponding links by applying the Knowledgescapes inference algorithm to the remaining data. For each experiment, we computed the statistical correlation R_{cv} between the logarithms of the predicted and actual need-document reading times; the logarithms of the actual reading times can be obtained directly from the need-document weights in the Knowledgescapes database. The strength of this correlation is considered to be a rough measure of the performance of the model, such that a strong positive correlation is indicative of an internally consistent model. We repeated this experimental procedure for various settings of the internal parameters, for various subsets of the data set, and for both the initial ranking and dynamic reranking cases. For initial ranking, we assume that no browsing history is available for the current information need; for dynamic reranking, we assume that the first d documents accessed during a logged information need are part of the current information need and then predict the reading time weights for the remaining documents accessed during the same logged need. We also varied the parameter d to examine the predictive performance of the model as more documents are browsed before dynamic reranking is invoked. For cross-validation computations, no partial query matches are allowed, i.e. a query q_2 is considered to match another query q_1 only if q_2 contains all of the terms in q_1.

The results of our simulations provide some insight into the capabilities and limitations of the Knowledgescapes approach to information retrieval. For initial processing of ad-hoc queries, we found a significant correlation (R_{cv} =0.30, significance=34.00) between predicted and actual log reading times. While users do tend to spend somewhat more time reading longer documents on the average, this correlation cannot be fully explained by variations in document length. For the entire Knowledgescapes database, we found a small but significant correlation (r=0.15, significance=16.19) between the log of the need-document reading time and the log of the length of the document in pages. This suggests that some useful tacit knowledge is being extracted from the historical Web server logs. We found that predictive performance varied widely when different kinds of queries are considered. Queries that specify only the document type field are generally vague and are typically used for browsing the document collection, while queries that specify particular keywords for full-text search are much more specific and more suited to focused information-seeking tasks. As shown in Table 1, R_{cv} is much larger for document type queries than for text queries. We suspect that since the broad and vague document type queries are submitted much more frequently than the more specific text queries, there is simply far more information available in the database for processing the type queries. In processing a keyword query, Knowledgescapes might typically draw on the browsing records of one or several other people who had entered the query before; whereas for document type queries the collective experience of hundreds of previous users may be available. Denser historical usage data will be needed to evaluate the predictive performance of Knowledgescapes for frequently submitted but specific text queries.

Table 1. Simulated performance of initial query processing, by query type.

Queries	Data Points	R_{cv}	Level of Significance
All Queries	12375	0.30	34.00
Text-only Queries	1524	0.14	5.69
Type-only Queries	3842	0.37	24.66

We also examined the effect of modifying the heuristic used to partition the Berkeley Digital Library Web server logs into individual information needs. Recall that a sequence of requests from a particular client IP address is partitioned into separate information needs whenever a new query is submitted or a threshold time interval τ elapses between successive requests. Table 2 below shows the relationship between the cutoff interval τ, the total number of information needs after partitioning, and the predictive performance R_{cv}. Note that R_{cv} varies little as τ increases from 5 minutes to 8 hours, although the number of information needs decreases by 23%. This suggests that the length of the interval τ is not crucial to the performance of Knowledgescapes, perhaps because realistic information needs vary widely in their duration. The optimal value of R_{cv} is achieved by setting the cutoff interval τ to between one and two hours; based on this finding we set τ to 60 minutes in our test prototype.

Table 2. Simulated performance of initial query processing vs. information need cutoff interval.

Information Need Cutoff Interval (τ)	Information Needs	R_{cv}
5 minutes	33208	0.28
10 minutes	29578	0.28
30 minutes	27159	0.29
1 hour	26457	0.30
2 hours	26036	0.30
8 hours	25617	0.29
10 days	24058	0.23

Cross-validation simulations of the dynamic reranking feature generally show that the predictive performance of Knowledgescapes improves when information about the user's recent browsing behavior is considered. The correlation R_{cv} also increases with *d*, the number of documents read between the issuance of the query and the computation of revised search results. In order to obtain a meaningful comparison of predictive performance on the same set of information needs using different values of *d*, we limited the data set under consideration to information needs that are linked to a sufficient number of documents. In addition, only information needs associated with document type queries are considered for this analysis, since only sparse data is available for full-text queries. Table 3 shows the predictive performance R_{cv} for information needs linked to at least 4 documents, as *d* increases from 0 to 3. In this case, R_{cv} is strongly correlated with the number of documents browsed before dynamic reranking is invoked (r=0.99, significance = 11.72). This observation lends circumstantial support to our hypothesis that adding recent browsing information to the user's query string results in a more precise definition of the user's current information need, with which the search engine can better predict the user's level of interest for various documents. While the overall results of our simulations are encouraging, they should be interpreted as partial validation of the internal consistency of our model, and not as proof of its effectiveness for general-purpose information retrieval applications.

Table 3. Simulated performance of dynamic reranking vs number of documents browsed since query was submitted. This simulation only considered information needs which are associated with document type queries and which are linked to at least 4 documents.

Number of Documents Visited (*d*)	Data Points	R_{cv}	Level of Significance
0	1586	0.51	23.64
1	1549	0.53	24.65
2	1302	0.57	25.14
3	1018	0.60	23.77

When evaluating these results, it is important to consider the structure of the Knowledgescapes database and the distribution of useful tacit knowledge therein. In order to support and enhance information retrieval, Knowledgescapes relies on associative knowledge encoded in the information needs stored in the database. An information need with links to multiple documents imply relationships among the documents which may be useful for dynamically updating search results based on recent browsing. On the other hand, an information need linked to a single document does not encode any associations between documents and thus is only relevant for computing initial search results given a query string. An analysis of the Knowledgescapes graph for our test prototype reveals that most information needs are poorly connected, with fewer than 10% of needs linked to three or more documents. Table 4 below shows the number of information needs and queries in the database as low-degree information needs are removed; it also shows the predictive performance R_{cv} for $d=0$ when only information needs with outdegree of at least three are considered. Most importantly, it shows that removing all information needs with outdegree of one or two has no noticeable effect on R_{cv}. This suggests that the tacit knowledge useful for information retrieval is distributed very unevenly across the Knowledgescapes database.

Table 4. The effect of removing poorly connected information need nodes on database size and initial query processing performance. For computing the correlations shown below, only information needs with outdegree of at least three are considered.

Database Subset	Info Needs	Distinct Queries	R_{cv}
All info needs	26457	3426	0.41
Info needs with outdeg>=2	6475	1435	0.40
Info needs with outdeg>=3	2597	714	0.42

5.2 User Evalution of the Initial Prototype

To provide a more realistic evaluation of the Knowledgescapes sytem, we made a test version of the Knowledgescapes search engine available to the regular users of the Berkeley Digital Library document collection. Since our approach is fundamentally based on leveraging the knowledge of previous users with similar information needs, it is especially important for any human-centered evaluation process to involve people whose information needs and work practices are representative of the regular user community for the particular information collection. To obtain a direct performance comparison of Knowledgescapes with existing content-based technology, we designed the user interface to present users with a sequence of search results drawn alternately from Knowledgescapes and Cheshire, with the source of the first entry randomly selected. The interface also asked users to provide subjective ratings (on a scale of 1-5 with 5 being the best rating) of the utility of the documents they browsed, with respect to their information needs at the time. To facilitate validation of the statistical user models underlying the system, the interface also logged all requests to the Knowledgescapes server, including both queries and content accesses.

The results of the initial study were inconclusive. While the Knowledgescapes site logged requests from 45 distinct client IP addresses, only about 10 document ratings were entered, all of which involved documents retrieved by the original Cheshire search engine. The reasons for the lack of data were several, and involved both technical and sociological factors. The Knowledgescapes prototype incurred noticeably longer latencies than the Cheshire site for queries and document requests, largely due to an inefficient implementation platform combining Java and CGI, which spawned a relatively heavyweight process for every request received by the Knowledgescapes web server. More importantly, we found that most of the query strings received by our server were not present in the database of historical usage data; this explains the lack of user ratings for documents recommended by Knowledgescapes. The rarity of query string matches resulted from the uneven and limited coverage of the query space by the Knowledgescapes database, as well as the prototype's insistence on exact string matches. We observed in many cases that Knowledgescapes could provide results for the queries entered if the user browses one or several documents and then requests dynamically updated results. The reranking feature, while an obvious component of our search interface, was almost never invoked; furthermore, while the interface provided buttons to rate documents and return to the search results page, users often bypassed these features and simply hit the "Back" browser button repeatedly until they retrieved the page of interest. We suspect that the mechanistic behaviors of regular users of the document collection are largely habitual, and that the activation energy required to alter these patterns is considerable. People are generally unlikely to make the effort to use a new and unfamiliar feature unless there is an obvious need for it or the performance of the existing technology is sufficiently lacking.

5.3 User Evaluation of the Revised Prototype

We then developed an improved prototype which attempted to remedy several of the key shortcomings of the initial Knowledgescapes site. The revisions ranged from cosmetic enhancements to the user interface to significant alterations of the underlying query processing mechanisms. To improve efficiency, the entire search engine interface was implemented using Java Servlet technology to eliminate the large per-request overhead incurred by CGI. In an attempt to expand the range of user queries supported by Knowledgescapes, the query processor was redesigned to retrieve partially matching query strings from the historical usage database. In the revised prototype, each query is converted into a set of atomic components, either stemmed full-text search keywords or document type categories. Each component is then assigned a numerical weight according to an inverse document frequency metric, to give greater emphasis to relatively rare topic-specific terms; the quality of a match between two queries is computed as the sum of the weights of the components that both queries have in common. We found that Knowledgescapes was able to recommend documents for the majority of user queries using this query matching heuristic, although search results based on partial matches often appeared to include a larger proportion of topically irrelevant documents.

Knowledgescapes, like most information retrieval models that rely entirely on mining implicit and associative knowledge, is highly susceptible to the phenomenon of topical drift. This occurs because a particular document may be relevant to many different information needs that are unrelated to each other; we found this effect to be especially noticeable for the document collection we used, as many of the documents were quite lengthy and spanned a wide range of topics. Furthermore, since Knowledgescapes contains no explicit information about the content of documents, the similarities between the user's query and the historical queries in the database provide the only topical information available for computing initial search results. For most user queries, only partial matches can be found, with these matches typically limited to one or two query terms that do not necessarily convey the intended semantics of the actual queries. For all these reasons, we realized that it would be inappropriate to use Knowledgescapes in its pure form as a stand-alone search engine, and unreasonable to compare our prototype directly against well-optimized production versions of existing information retrieval engines. However, Knowledgescapes would be potentially more useful as an enhancement to existing content-based search engines, and as a supplementary source of human knowledge to be used mainly to infer the relative utilities of documents already determined to be potentially relevant based on their content. Accordingly, we modified our prototype to use a two-stage retrieval process. First, an initial set of documents is retrieved from Knowledgescapes; this set of candidate documents is then filtered for topical relevance by removing all documents that fail to achieve a threshold relevance score from the Cheshire text retrieval engine. Our evaluation process is thus intended to determine the degree to which Knowledgescapes adds value to a traditional content-based system by producing an ordering of the result set that better reflects the user's specific information needs.

Finally, to encourage user participation, we offered several prizes to be awarded to Digital Library users according to a lottery in which participants received additional tickets for each document rating entered. The revised test prototype of Knowledgescapes was deployed for a 43-day period during May and June 2000. We also added a hyperlink from the main Berkeley Digital Library web site to the Knowledgescapes search site to make it more easily accessible to digital library users.

5.4 Results of the Final User Evaluation

During the test period, the Knowledgescapes server logged requests from 52 client IP addresses, which we believe to be roughly indicative of the number of users who used the search site. The combination of system improvements and usage incentives appeared to generate only a modest increase in user interest and participation. While Knowledgescapes processed 93 distinct query strings during the test period, only eight people registered for the prizes, and only one person rated more than three documents. Overall, 28 document ratings were entered from 16 different client addresses; five of these ratings involved documents recommended by Knowledgescapes. The average over all ratings was 2.57 on a scale of 1 to 5; if we consider separately the documents recommended by Knowledgescapes and Cheshire, the average ratings were 2.40 and 2.61, respectively. We found no significant systematic difference in subjective ratings between these two sets (significance = -0.25). In only one instance did any user rate a document after requesting dynamically updated search results. The preponderance of Cheshire recommendations among the set of documents rated can be explained by the relatively small size and uneven topical coverage of the Knowledgescapes database, as discussed in the previous section. While the overall results suggested that our users were not very satisfied with the recommendations of either search engine, we did observe a bipolar distribution of subjective ratings, with ratings of 1 (11 entries) or 5 (7 entries) the most common. This may be reflective of a professional user community with high expectations and very specific information needs.

Users of the Knowledgescapes search site spent an average of 200 seconds browsing each document they visited and requested an average of 9 pages. We found a correlation of 0.359 (significance = 1.962) between the time spent browsing a document and the rating entered; we observed a similar correlation of 0.367 (significance = 2.013) between the number of pages read and the document rating. When we compared the log of the reading time and the log of the number of pages read, respectively, to the rating entered, the correlation decreased to 0.165 (significance = 0.855) and 0.244 (significance = 1.281) respectively. These results lend some support to the basic assumption in our model that the amount of time people spend reading a document is directly related to its perceived utility. Given the limited number of data points available, however, the mathematical nature of the relationship between reading times and subjective utility ratings for digital library document collections remains unclear.

Undoubtedly, a substantial body of usage and rating data would contribute towards a better understanding of the potential capabilities of usage-based inference in information retrieval and of the nature of the information-seeking and document-browsing processes. Unfortunately, the quantity of usage and subjective rating data obtained was insufficient to support a conclusive evaluation of the effectiveness of the Knowledgescapes paradigm or of the predictive accuracy of our statistical models. User participation (as measured by the frequency of requests to the

Knowledgescapes server) decreased noticeably over the course of the trial period, and it became apparent that further extension of the user testing period would not generate a data set of the magnitude needed for detailed evaluation and performance analysis. Considering the similar levels of participation in our initial and final user trials despite our extensive efforts to enhance system performance and usability, we believe that additional iterations of the development cycle are unlikely to greatly increase participation and data collection. Sociological factors undoubtedly contributed to the limited usage of our prototype. We suspect that many regular users of the environmental document collection are professionals who use the collection quite extensively in their work and who have become accustomed to following well-established patterns of behavior when searching for information. Previous research has suggested that web-surfing navigation patterns are often habitual, containing many repeated sequences of navigation steps [16]. Some users may also have bookmarked the standard Cheshire search site or specific documents of interest to them. In general, people will modify their behavior to use a new feature only when the perceived additional utility of the feature, and their actual need for this additional performance, is sufficient to provide the activation energy to overcome their inertia. We also suspect that users are typically not willing to spend time browsing many pages of additional search results or trying out advanced features such as dynamic reranking when they are dissatisfied with the first page of results; instead, they are far more likely to attempt to refine or reformulate their queries. The common practice among web surfers of successively revising, specializing, or generalizing an ad-hoc query is well documented in the literature [12]. In hindsight, we believe that users were more likely to be concerned about the smaller set of available search fields in Knowledgescapes than to be enticed by the additional features we included.

During the development process, we experimented with using the Knowledgescapes database to find additional documents similar to a particular document. Each document was represented as a unit vector in information need space, with each information need node receiving weight proportional to the reading time for the document by the respective information need. The similarity of two documents was then defined as the inner product of the respective unit vectors. This computation is essentially a form of collaborative filtering in which information need nodes, rather than people, serve as the participants. This simple feature appeared to perform quite well at first glance; while we did not pursue a quantitative comparison, Knowledgescapes often exhibited noticeably higher precision in retrieving documents that are topically related to a given document than in finding documents relevant to a query string. We also experimented briefly with applying naïve agglomerative clustering algorithms to the historical usage data to generate a hierarchical clustering of the documents in the collection. While we made no serious attempts to tune or optimize the clustering process, the resulting hierarchy appeared to contain numerous topically-coherent document clusters, as well as some plausibly organized subtrees. We suspect that our techniques may be much better suited for finding useful semantic relationships among information objects than for processing ad-hoc text queries. After serious consideration, we decided not to deploy these features in our test prototypes, as we believed that doing so would result in an excessively complex user interface and would interfere without our efforts to obtain a direct performance comparison between Knowledgescapes and Cheshire. Both of these directions remain promising avenues for further research.

6. Discussion

The task of designing an effective, practical information retrieval system based on mining implicit knowledge is fraught with many challenges. These include fundamental limitations of these methods which will require further research; technical difficulties in obtaining richer data sets and/or extracting more usable information from existing ones; and practical and sociological constraints that might be addressed either through revisions in our methodology or through the passage of time. Our experience in developing such a system has provided us with a better understanding of the key issues in this field and the ramifications for designers of future systems. We discuss each of these issues below.

6.1 Fundamental Limitations

As we had found during our experimentation with Knowledgescapes, usage-based information retrieval systems require dense access data to be effective. The historical database must be sufficiently dense in the query, document, and information need spaces. In our experiments with the Knowledgescapes prototype, the sparse and uneven distribution of data in the query space presented the greatest obstacle for most searches. Our database contained exact matches for only 30 of the 93 distinct query strings submitted during the testing period; and while partial matches were found for the majority of the remaining queries, these tended to be semantically quite different from the user's queries, often omitting some of the more salient query terms. It is plausible that heavily used commercial Web search engines might achieve much higher data densities for popular queries, although the query space is also undoubtedly much larger for general-purpose applications than for our prototype. The distribution of actual user queries over the query space is also very uneven for most application domains; in most cases, a relatively small subset of the query strings submitted account for the large majority of requests to a search engine. Similarly, the frequencies with which Web documents are requested has been shown to follow a long-tailed Zipf distribution [15]. In general, usage-based retrieval techniques are not well suited to processing rare queries or evaluating infrequently requested documents, and will not aid in the discovery of new information objects that have not been browsed by previous users. The dependence of these methods on the density and coverage of historical data limits their utility for providing initial results in response to ad-hoc queries. For these reasons, as well as the problem of topical drift discussed in Section 5, we believe that any successful and practical applications of usage-based methods will use them in combination with content-based retrieval techniques.

The distribution of access data in the information need space also constrains the range of information-seeking tasks that Knowledgescapes can support. The contents of a usage database necessarily reflect the needs and work practices of the people who used the information collection during the period for which server logs have been maintained. While our inference model represents the user's need as a probability distribution over the set of historical needs, the user's information need may in fact be fundamentally different from all of the recorded needs. Furthermore, the information need space is high-dimensional and needs may in many cases be orthogonal to each other. For example, a user seeking information on earthquakes in California and another user researching local fire regulations in Berkeley might both consider a zoning document for the city of Berkeley to be highly relevant. In the former case, the information need is topically specified; while in the latter case the need is geographically specified. Knowledgescapes is likely to return irrelevant results if the user's need is orthogonal to the information needs of most of the people who have entered similar queries in the past. Unfortunately, there is no easy way to "retune" Knowledgescapes to support a different user community, or one whose needs have changed since the web server logs were compiled.

The nature of short-term information needs is also problematic. People typically do not browse very many documents from each search result set; more often, they either find the answers they are looking for after visiting a small number of documents, or reformulate their queries in the hope of retrieving more useful results. Consequently, the Knowledgescapes graph contains only a small amount of data for most historical needs. On the average, a need was linked to only 1.55 documents, and 76% of all needs in the database were linked to a single document. This data is generally insufficient to describe or specify the detailed semantics of an actual user's information need, and in practice the database contains only a vague description of most historical needs. It is very possible that the user is in fact looking for a very specific piece of information, while Knowledgescapes represents the information need as a brief interval of time during which he visited one document (which might have been requested for a wide range of reasons). These characteristics of short-term information needs also limit the effectiveness of the dynamic reranking feature, which infers semantic relationships between the user's current information need and historical needs based on the intersections of the weighted sets of documents linked to the respective needs. In the extreme case, no relationship can be inferred between two semantically identical needs for which the required information is retrieved from different documents.

As we had discussed in Section 4.2, the exact structure of the Knowledgescapes graph depends largely on the heuristics used to partition the access log into information needs. While changing the maximum duration of information needs is unlikely to improve results, we suspect that the partitioning algorithm can be improved by formulating a more sophisticated model of the search process that accounts for the successive refinement of query strings within the context of a single need. If successful, this would result in a graph that contains fewer information needs, with more detailed information on individual needs and more associative relationships among needs. In addition, we could incorporate more detailed information about each historical need by representing the document collection at a finer level of granularity, i.e. treating individual pages rather than documents as atomic information objects in the Knowledgescapes graph. This would mean enumerating the individual page reading times logged for each need, instead of aggregating reading times over entire documents as was done in our prototype. However, we suspect that the density of the test data set over the space of individual document pages would be much too sparse to support information retrieval for the Berkeley Digital Library testbed. In general, the distribution of data in the information need space, the structure of the need-document graph, and the optimal heuristics for drawing need boundaries are likely to be highly specific to the application domain.

In order to evaluate the long-term prospects for tacit knowledge mining paradigms in information retrieval, we must consider several very fundamental issues. First and foremost, the raw data must contain sufficient information in the form of implicit human judgment to provide useful support for information-intensive tasks. If we are mining web server logs, for example, the stochastic pattern of HTTP requests must be punctuated by a signal that correlates strongly with users' subjective judgments of the utility of various documents for various information needs. Furthermore, access log data is intrinsically very noisy due to the stochastic nature of the underlying processes, and the raw data must contain patterns with sufficiently large signal-to-noise ratios to support practical applications. For example, it is not sufficient simply to build a search engine on the assumption that a particular pattern of logged requests indicates the relevance of a certain document to a particular information need; we must also be able to say with a reasonable degree of confidence that this pattern most likely did not occur by chance. For people searching large semistructured or unstructured information collections such as digital libraries or the WWW, high precision is usually more important than high recall, and is far more difficult to achieve in practice. In addition, since traditional full-text search engines have a ubiquitous tendency to retrieve large numbers of documents for common queries at the expense of quality and selectivity, the marginal utility of including a usage-based component is largely a function of its ability to increase precision by filtering out useless or irrelevant items. For these reasons, it will be very difficult to develop a useful and practical search engine that is based on analyzing short-term information needs for which only limited data is typically available. For example, the dynamic reranking feature in Knowledgescapes attempts to infer the user's current information need based on records of his recent browsing behavior; in most cases, this data consists of reading times for 1-3 documents, which is generally insufficient to accurately define the semantics of the need. As we discussed above, most existing collaborative filtering systems are based on models that make recommendations based on long-term user profiles. While these models lack the temporal context and specificity of the short-term models, they have the important advantage of higher signal-to-noise ratios and potentially higher precision. Therefore, we expect that long-term profiling will remain a dominant paradigm in collaborative information systems for the foreseeable future. Further research will be needed to test the feasibility of the short-term models for practical applications.

The feasibility of a usage-based paradigm for general-purpose information retrieval also depends on the existence of domain-independent patterns in historical access data that generalize across a wide spectrum of subject areas and information-intensive tasks. Such patterns cannot be specific to the structure and semantics of a particular information collection or application, but must reflect basic properties of the behavioral and cognitive processes by which people build and use collections. Link-based search engines are based on one such property, namely the fact that people prefer to create hyperlinks to sites that they consider to be useful, authoritative, or relevant to their own site. This pattern is highly domain-independent; a site that is pointed to by many other sites is usually an influential site, at least in some particular Web community. It also has a high signal-to-noise ratio, since creating a hyperlink requires conscious effort and judgment; Web sites with the connectivity of a Yahoo do not happen by chance. Our Knowledgescapes model is also based on a pattern that we hypothesized to be very general – the tendency for people to spend more time browsing documents that better satisfy their needs. However, this pattern is less salient and consistent, because browsing is often an exploratory process and because people do not know what content a document contains until they have spent some time reading it. Even though there is far more Web server log data in existence than hyperlink data, the link-based model is in many ways more precise and reliable. It is possible that there are other patterns embedded in server logs that might be mined to support information retrieval. However, domain-specific patterns are likely to be of limited utility, as the research and knowledge acquisition overhead required to formulate and validate a different usage model for each application is probably too high. It remains to be seen whether a reliable, domain-independent usage-based model exists, and we believe that this is a fertile subject for long-term research.

6.2 Technical Challenges

The performance of a usage-based search engine is a function of both the underlying mathematical usage model and the historical access database for the collection of interest. The quality of the database depends not only on the density of the data and the coverage of the relevant domain space, but also on the degree to which the data reflects actual human judgments of the utility of the documents for various information needs. Unfortunately, the web server logs for most realistic document collections, including the document collection used for the Knowledgescapes prototype, fall far short of this ideal. Historical usage patterns for the Berkeley Digital Library document collection are strongly influenced by the rank ordering bias of the Cheshire II search engine, which has served as the primary access route to the actual documents. A comparison of the documents requested by historical users after entering a Cheshire II query with the current results returned by Cheshire II for the same queries revealed that the majority of the logged requests were for the top 10 documents returned by Cheshire II, even though Cheshire II returned well over 100 results for many common query strings. Undoubtedly, the historical usage level for particular documents is a function of Cheshire II's recommendations for frequently entered queries as well as their actual utility with respect to user needs. The standard interface for browsing the collection also did not allow users to enter ratings for documents. In the absence of subjective utility data, it is impossible to distinguish the desired signal reflecting human judgments of document utility from the noise associated with historical rank ordering bias. Such subjective data would also provide empirical evidence for testing and validating the basic assumptions underlying our statistical models. Unfortunately, most existing digital library testbeds are likely to suffer from these difficulties, as it is very difficult to find substantial, publicly available data sources that meet our stringent requirements. For many domains, it may be the case that an effective usage-based search engine can only be developed by incorporating the collection of the required information into the original design of the collection.

6.3 Practical and Sociological Factors

Even if it were possible to solve all of the technical problems discussed above, the large-scale deployment of usage-based information retrieval engines would still face significant social, cultural, and economic barriers. Unlike hyperlink data, which can be obtained simply by crawling the Web, most historical usage data resides in server logs which are distributed over many thousands of institutions and which usually are not available to the public. Corporations increasingly view access data as valuable intellectual capital, as they can often generate substantial revenues through the sale or proprietary use of detailed marketing data. Many institutions may also be reluctant to release log data due to concerns of privacy and confidentiality; furthermore, these concerns cannot be entirely allayed by anonymizing client identities, as the mere perception of loss of privacy can result in the loss of public trust. The tracking of real-time user behavior for supporting dynamic reranking of search results may also raise privacy concerns, especially with web sites intended for general public use.

For all of these reasons, it appears highly unlikely that the mining of historical access logs will prove to be a feasible paradigm for large-scale Web search engines in the near future. We believe that usage-based approaches are far more likely to be applied to information retrieval systems for centrally-administered collections such as corporate intranets and digital libraries where the requisite access data is readily available and privacy concerns are less crucial. These approaches might be particularly helpful for searching large, flat collections of information objects, especially in application domains where the specific information needs of users are not easily satisfied by content-based retrieval techniques.

The potential impact of advanced search features such as dynamic reranking also remains unclear. While it is possible that people are simply reluctant to use the reranking feature because it is new and unfamiliar, attempts to personalize search results based on brief observations of browsing behavior face several basic obstacles. As we discussed above, the amount of useful information available for refining results is an increasing function of the user's interaction with the collection after the active query was issued. In many cases, the user may have to spend a significant amount of time browsing at least several documents before there is a strong signal reflecting the user's information need. We suspect that most people may not have the patience to extensively explore a result set that they find less than satisfactory; moreover, incremental query refinement is known to be an integral part of the information-seeking process for most Web users. The purpose and the behavior of the refinement feature also may not be intuitive to most users, who may not readily comprehend the concept of a personalized search that is not based on a persistent user profile. Furthermore, people may not appreciate automated attempts to "second guess" their intentions, especially if the software sometimes generates incorrect "guesses" that complicate their information-seeking tasks.

7. Extensions and Future Work

We believe that data mining of usage data continues to be a potentially highly rewarding area of research despite the limitations described in the previous section. Considering the accelerating accumulation of historical access data and the logistical difficulty and expense of compiling large knowledge bases by hand, even modestly effective techniques for leveraging the vast and largely under-exploited repositories of access logs could provide substantial dividends for some applications. In the previous section, we discussed the challenges involved in using this data to support information retrieval and identified a number of problems requiring long-term research. In this section, we outline several specific directions for extending and applying our research. We explore the following problems: improving statistical usage models, scaling to large diverse collections, and applying usage-based techniques to support applications that classify and organize information collections and user communities to enhance information access.

7.1. Empirical Models of Information Usage Behavior

The feasibility of usage-based information retrieval systems depends largely on the existence of statistical models of the underlying search and browsing behaviors that meet the requirements discussed in Section 6.1. Further research will be needed to formulate and test these models. This is a very open-ended research problem, and further inquiry is needed to better define and constrain the problem space. Empirical validation will undoubtedly be the most logistically frustrating and time-consuming aspect of this research, as it is likely to require human-centered testing of each candidate model, preferably in the context of several different application domains. This is also an interdisciplinary problem at the intersection of the computational and behavioral sciences; thus, models and empirical results from the social science literature may provide some helpful insight.

The Knowledgescapes model is based on the hypothesis that the amount of time a person spends reading a document is related to the document's perceived utility. Extensive empirical testing of this hypothesis will be needed to determine the strength of this relationship, its generality and applicability across a wide range of domains, and the mathematical form that might best model it. This testing would involve having human subjects browse and rate documents, and then correlating the reading times with the subjective ratings; this should be done for several different information collections and user communities. It would also be worthwhile to experiment with different models for partitioning chronological Web server logs into information needs. Finally, it may be possible to extract from historical access logs other properties besides reading time that reflect human judgments of the relevance and utility of documents. It remains to be seen if any such properties exist which might prove to be of practical value.

7.2. Large Heterogeneous Information Collections and the WWW

In our experimental work, we had developed a prototyped of the Knowledgescapes search engine for a medium-sized collection of text documents. While we chose this collection for key logistical reasons including the availability and density of access log data, our choice of testbed did not allow us to fully exploit the generality of our model nor strenuously test its scalability. Many real-life information collections, including digital libraries, organizational intranets, and especially the WWW, are both large in size and diverse in the range of content types contained. In addition to static HTML and ASCII documents, Web sites can contain embedded images, streaming audio and video objects, and online services which dynamically generate content by executing Java applets or CGI scripts. General-purpose content-based retrieval techniques for unlabeled images and video streams are currently quite primitive, and it is often impossible to evaluate the quality of a dynamic Web page by indexing the HTML and source code text. Since Knowledgescapes works by modeling the generic properties of the information search process and not by analyzing the actual content of the information objects, we would expect that its capabilities will extend naturally to all content types as long as the required historical access data is available. In fact, we believe that the potential value of usage-based information retrieval is even greater for non-text media, because of the limited capabilities of existing content-based retrieval systems for these domains. Undoubtedly, it would be very worthwhile to test these techniques empirically on several different types of non-text information collections.

Further research will be needed to deploy practical usage-based search engines to collections as large as the WWW. The computational complexity of processing a query is highly sensitive to the connectivity of entities in the Knowledgescapes graph; in general, common queries that return many relevant documents are much more expensive to process than rare queries with few matching documents. For a graph in which each query string is linked to s information needs, each need is linked to d documents, and each document is linked to n needs, initial query processing requires $O(s*d)$ time and dynamic reranking results after browsing m documents requires $O(m*n*d)$ time in the worst case. The size of the Knowledgescapes database grows linearly with the total number of requests recorded in the server logs. Both time and space complexity will pose serious challenges for searching a collection having the size and uneven traffic distribution of the WWW. It remains to be seen if it is possible to improve scalability without excessively sacrificing precision by using heuristics to prune the data set, or by applying dimensionality-reduction techniques such as latent semantic indexing.

7.3. Clustering and Semantic Classification

Web server logs also contain a wealth of tacit knowledge about the relationships among the documents that comprise a collection and among the people who use it. This associative knowledge can be extracted using a number of statistical data clustering techniques. These techniques can often be used to cluster data sets from a wide range of sources, as long as they contain the required similarity information; for geometric clustering, the entities to be clustered are typically represented as vectors in some high-dimensional feature space, while for hierarchical clustering the input usually consists of a matrix specifying the pairwise similarity among entities. Agglomerative hierarchical clustering can be used to organize documents into a topical hierarchy of semantic categories; such approaches have been applied to content-based representations of Web documents [18]. An interesting research project would be to apply these clustering algorithms to both content-based and usage-based data sets from the same document collection and compare the structure and topical coherence of the resulting subject hierarchies. Ultimately, document clustering schemes must be evaluated based on the added utility that they provide to people searching for information. We believe that usage-based clustering may have greater potential than content-based models, since the underlying data more directly reflects associations amongs documents as perceived by human users of the collection. Another worthwhile experiment would be to incorporate document clustering into the Knowledgescapes search engine, to consider previous users' judgments of related documents when evaluating each document's utility for the current user's needs. For large, diverse collections with uneven access patterns, clustering also offers the potential advantage of increasing the density of historical data and reducing the dimensionality of the document space. Finally, hierarchical clustering can also be used to categorize people who have browsed a document collection based on their browsing history. This has several potential applications, such as mapping the structure of the user community and searching for people with similar interests or expertise. The ideal choice of algorithms for clustering historical usage data remains an open research question.

7.4. Authority Mining and Expertise Location

For many information collections including the WWW, the user community is highly heterogeneous and includes people with widely varying degrees of knowledge and experience in different subject areas. Therefore, it may be unreasonable to assume that every person's judgment of a document is equally qualified and useful, especially for specialized topical domains. A more discriminating approach would be to compute authority scores for specific individuals based on their past behavior and use these scores to weight their respective judgments when computing recommendations for a query. The concept of authority has been used very effectively in link-based Web search engines [10]. In these models, the authority of a Web site is a function of both the number of other sites that link to it and the respective authorities of those sites; in other words, authority is conferred by those having high authority. For a wide range of hyperlinked topologies, this recursive equation can be solved by computing the principle eigenvector of the adjacency matrix. Likewise, we believe that it may be possible to compute authority weights for people and/or documents in the Knowledgescapes graph, and to use these weights to emphasize the recommendations of those who are likely to be more knowledgeable and experienced. The heuristics by which these qualifications might be inferred from Web server log data and the mathematical algorithms for computing authority values for nodes in the Knowledgescapes graph are both subjects for future research. One potentially promising approach, modeled after the Kleinberg hubs-authorities algorithm [10], might be to compute the eigenvectors of the matrix corresponding to the bipartite graph of people and documents. Since expertise is intrinsically specialized, we could also extend the concept of authority by assigning separate authority levels describing an individual's qualifications with respect to different topical categories. Each person's range of knowledge and interests would be described by an expertise profile, which could be computed by running the authority algorithm once for each category, excluding from the Knowledgescapes graphs any links to documents not belonging to the respective category. Users could then be clustered according to their profiles to support expertise location services. The incorporation of authority and expertise models into information systems continues to be an important research problem for human-centered computing.

8. Conclusions

We examined the problem of augmenting content-based information retrieval techniques by mining the collective knowledge of the user community supported by an information collection. We devised and formulated a Bayesian probabilistic model for predicting the utility of an information object with respect to a specific short-term information need, based on historical usage data rather than document content. Using this model, we developed a novel search engine for the Berkeley Digital Library document collection. While it is necessarily difficult to measure the performance of such a system, our prototype suggests how the vast repository of implicit human knowledge might be put to use in a practical setting.

Our work on algorithms for mining tacit knowledge and new paradigms for leveraging this resource to support information-intensive tasks is merely one contribution to a broad and fast-growing area of research. Through our experimental efforts we have gained an understanding of the major issues and challenges involved and identified potentially promising avenues for future research. Our experiences have also imbued an appreciation for the inherent limitations of collaborative and usage-based paradigms for information retrieval. It remains an open research question to determine how much useful information can be exploited from tacit behavioral data, and to what degree this can be done using general-purpose, domain-independent models, lest we revisit the well-known problems of knowledge acquisition and artificial intelligence.

9. References

[1] Berkeley Digital Library document collection web site. http://elib.cs.berkeley.edu/docs

[2] Cheshire II full text search engine description. http://elib.cs.berkeley.edu/Cheshire/about.html

[3] William Cooper and Fredric Gey. Probabilistic Retrieval Based on Staged Logistic Regression. *ACM SIGIR*, June 1992, Copenhagen, Denmark.

[4] Xiaobin Fu, Jay Buzdik, and Kristian J. Hammond. *Mining Navigation History for Recommendation. Proceedings on Intelligent User Interfaces 2000.* ACM Press, 2000.

[5] Rodney Fuller and Johannes J. de Graafe. Measuring User Motivation from Server Log Files. Conference presentation for *Designing for the Web: Empirical Studies*, Microsoft Corporation, October 1996.

[6] Google search engine. http://www.google.com/

[7] David Heckerman and Eric Horvitz. Inferring Informational Goals from Free-Text Queries: A Bayesian Approach. *Proceedings of the Fourteenth Conference on Uncertainty in Artificial Intelligence*, Madison, WI, July 1998, pp 230-237.

[8] Eric Horvitz, Jack Breese, David Heckerman, David Hovel, and Koos Rommelse. The Lumiere Project: Bayesian User Modeling for Inferring the Goals and Needs of Software Users. *Proceedings of the Fourteenth Conference on Uncertainty in Artificial Intelligence*, July 1998.

[9] Bernardo Huberman, Peter Pirolli, James Pitkow, and Rajan Lukose. Strong Regularities in World Wide Web Surfing. *SCIENCE* 280: 95-97, 1998.

[10] Jon M. Kleinberg. Authoritative Sources in a Hyperlinked Environment. *Proceedings of the Ninth ACM-SIAM Symposium on Discrete Algorithms*, 1998.

[11] J. A. Konstan, B.N. Miller, D. Maltz, J.L. Herlocker, L.R. Gordon, and J Riedl. Applying collaborative filtering to usenet news. *Communications of the ACM*, 40(3), 77-87, 1997.

[12] Tessa Lau and Eric Horvitz. Patterns of Search: Analyzing and Modeling Web Query Refinement. *Proceedings of the Seventh International Conference on User Modeling*, Banff, Canada, June 1999. New York: Springer Wien, 119-128.

[13] David M. Nichols. Implicit Rating and Filtering. *Proceedings of the 5th DELOS Workshop on Filtering and Collaborative Filtering*, Budapest, Hungary, November 10-12, 1997, ERCIM, 31-36.

[14] David M. Pennock and Eric Horvitz. Collaborative Filtering by Personality Diagnosis: A Hybrid Memory-and Model-Based Approach. *IJCAI Workshop on Machine Learning for Information Filtering, International Joint Conference on Aritificial Intelligence (IJCAI-99)*, August 1999, Stockholm, Sweden.

[15] James Pitkow. Summary of WWW characterizations, *Seventh International World Wide Web Conference,* April 1998, Brisbane, Australia.

[16] Peter Pirolli and James Pitkow. Distribution of surfers' paths through the World Wide Web: Empirical Characterization. *World Wide Web (2)*: 29-45, 1999.

[17] Cyrus Shahabi, Amir. M. Zarkesh, Jafar Adibi, and Vishal Shah. Knowledge Discovery from User Web-Page Navigation. *Proceedings of the IEEE RIDE97 Workshop*, April 1997.

[18] Oren Zamir and Oren Etzioni. Web Document Clustering: A Feasibility Demonstration. *ACM SIGIR*, August 1998, Melbourne, Australia.