Navigation-Aided Retrieval

Shashank Pandit
Carnegie Mellon University
shashank@cs.cmu.edu

Christopher Olston
Yahoo! Research
olston@yahoo-inc.com

ABSTRACT

Users searching for information in hypermedia environments often perform *querying* followed by manual *navigation*. Yet, the conventional text/hypertext retrieval paradigm does not explicity take post-query navigation into account. This paper proposes a new retrieval paradigm, called *navigation-aided retrieval* (NAR), which treats both querying and navigation as first-class activities. In the NAR paradigm, querying is seen as a means to identify starting points for navigation, and navigation is guided based on information supplied in the query. NAR is a generalization of the conventional probabilistic information retrieval paradigm, which implicitly assumes no navigation takes place.

This paper presents a formal model for navigation-aided retrieval, and reports empirical results that point to the real-world applicability of the model. The experiments were performed over a large Web corpus provided by TREC, using human judgments on a new rating scale developed for navigation-aided retrieval. In the case of ambiguous queries, the new retrieval model identifies good starting points for post-query navigation. For less ambiguous queries that need not be paired with navigation, the output closely matches that of a conventional retrieval system.

Categories and Subject Descriptors

H.3 [**Information Systems**]: Information Storage and Retrieval

General Terms

Algorithms, Design, Experimentation

Keywords

Web search, Navigation, Browsing, Link analysis, Underspecified search tasks

1. INTRODUCTION

While performing *search tasks*[1] in the hypertext environments such as the World Wide Web, users often supplement querying with extensive manual *navigation* (i.e., traversal of hyperlinks) [30]. There are at least three reasons for using navigation as a search tactic:

1. **Difficulty in formulating appropriate queries**: Users often do not convey their search task to a retrieval system accurately enough to allow them to forgo navigation. A variety of factors may be to blame, including limited query language expressiveness, user inexperience, lack of familiarity with terminology, and cognitive ease of entering short queries.
2. **Open-ended search tasks**: In many cases the scope of the task is broad, and the user has not (yet) formed a concrete notion of what information would lead to its successful completion. (Indeed there may not even be a meaningful notion of completion.) For example, consider a guitar enthusiast who recently moved to a new city and wishes to find out about the city's guitar culture and local guitar-related resources. This task's scope may include performances, lessons, shops, social networking, gigs, as well as other aspects the user discovers in the process of searching (e.g., perhaps he discovers that a local museum features an exhibit of musical instruments). Open-ended search tasks often entail a significant amount of manual navigation, as part of an extended process of exploration, discovery, and task/query refinement known as berrypicking [2] or information foraging [29].
3. **Preference for orienteering**: At times, users prefer to navigate rather than "teleport" to a target document, because doing so enables them to understand the surrounding context. This behavior is called *orienteering* [36].

Despite the prevalence of navigation as a search tactic, the conventional (hyper)text retrieval paradigm focuses uniquely on querying. In this paper we approach the combination of querying and navigation as the unit of interest, in which querying merely identifies starting points for navigation, and navigation is guided based on the user's query.

1.1 Navigation-Aided Retrieval

We propose a new hypertext retrieval paradigm that incorporates post-query user navigation as an explicit component, called *navigation-aided retrieval* (NAR). With NAR, as with the conventional paradigm, users submit freely-chosen keyword queries and are presented with a ranked list of documents. Unlike with the conventional paradigm, the documents in the list represent *starting points* from which the user can commence exploration. Loosely speaking, good

[1] A search task is the quest for information by a person with insufficient knowledge to solve a certain problem at hand [3].

WWW 2007, May 8–12, 2007, Banff, Alberta, Canada.
ACM 978-1-59593-654-7/07/0005.

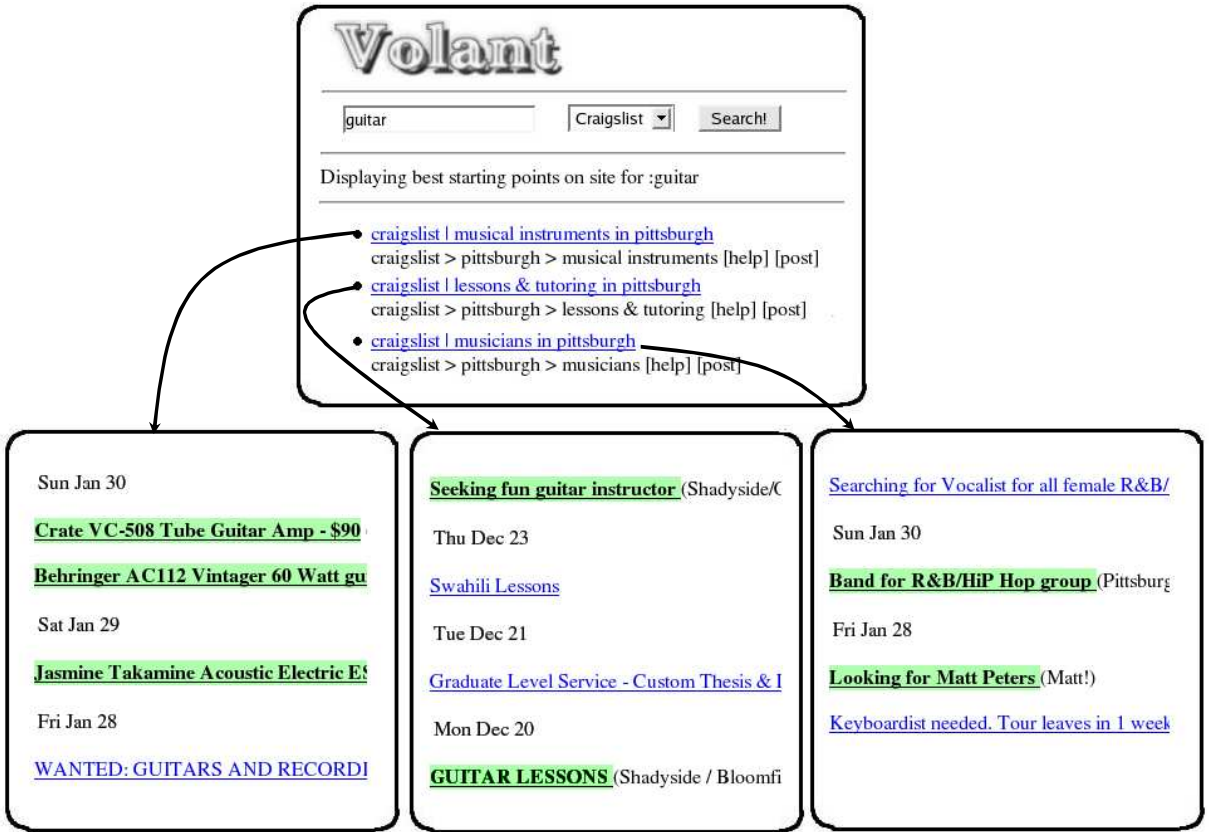

Figure 1: Screenshots of our prototype navigation-aided retrieval system, for query "guitar" over Craigslist Pittsburgh.

starting points are hypertext documents that, while they may not match the user's query directly, permit easy navigation to many documents that do match the query, via one or more outgoing hyperlink paths (our starting points differ from Kleinberg's "hubs"; see Section 2.1).

Figure 1 shows the output of our prototype NAR system called *Volant* for the query "guitar" over a community bulletin-board Web site called Craigslist Pittsburgh[2]. The upper screenshot shows the initial response page (list of starting points); the other three show sample content from each of the top three starting points. Notice how the list of starting points neatly categorizes the relevant information available on the site. This categorization is a consequence of the fact that the Web site is appropriately organized, combined with the navigation-aware design of the scoring function used by Volant to identify and rank starting points. The model underlying the scoring function assumes the user has a certain propensity to navigate outward from the initial query results, and that navigation is directed based on the user's search task. Importantly, our navigation-aided retrieval model strictly generalizes the conventional probabilistic information retrieval model, which implicitly assumes no propensity to navigate (formal details are provided in Section 3).

Coming back to Figure 1, notice that certain hyperlinks are highlighted (i.e., they have a shaded background). Volant assists users as they navigate outward from the starting points by highlighting hyperlinks that lead to one or more documents matching the query ("guitar," in this case), following [20, 28]. In general, we refer to post-query hyperlink annotation as *navigation guidance*, which is the second key aspect of the NAR paradigm.

In NAR systems, the user may revise his query while navigating. Doing so causes the navigation guidance annotations (e.g., link highlighting) to update in place, without changing the content on display.[3] The user can therefore refine/reformulate his query incrementally, without losing his bearings.

To summarize, in light of the fact that users tend to supplement querying with navigation, navigation-aided retrieval ensures that querying and navigation work together, rather than interfering with each other: The initial query takes the user to document neighborhoods that are both relevant and navigable. Once there, the user can navigate while retaining relevance indicators created by querying. Conversely, he can re-query while retaining the context arrived at by navigation.

1.2 Organic versus Synthetic Structure

The importance of supplying context with query results is well recognized [15]. Most prior work along these lines aims to synthesize some sort of structure automatically when a query arrives, to serve as a contextual backdrop for query results and provide semantically meaningful avenues for exploration. Examples include query result clustering [40, 41] and

[2]`http://pittsburgh.craigslist.org`

[3]The user can of course request a fresh list of starting points instead, if desired.

"faceted" query/navigation interfaces [16, 33]. These approaches can be characterized as navigation-aided retrieval with *synthetic* structure (Synthetic NAR).

In this paper we are primarily interested in navigation-aided retrieval using *organic* structure, i.e., structure that is naturally present in pre-existing hypermedia documents (Organic NAR). Synthetic and Organic NAR each offer certain advantages. Clearly, Synthetic NAR is more flexible, and does not rely on the a priori existence of suitable starting point documents. The principal advantages we perceive for Organic NAR are:

- **Human oversight.** Working with pre-existing structure ensures that a human oversees the way information is organized. Therefore it is more likely that categories make sense, have proper labels, and that each category has information organized in a useful way (e.g., Craigslist postings are sorted by date).
- **Familiar user interface.** In our paradigm, the user interface departs only slightly from the one users are already familiar with (i.e., queries return a simple, flat list of links to documents). Interface complexity can be dangerous: a user study evaluating a kind of faceted search interface [33] found that the interface, which displays several types of information views and navigation paths simultaneously, required significant learning time, which hampered effectiveness.
- **A single view of the document collection.** Unlike with faceted search interfaces, with Organic NAR users only have to contend with one "view" of the document collection, and content is presented in the same way each time they visit (modulo navigation guidance). Hence users are more easily able to retrace their steps, a property that is crucial for orienteering.
- **Robust implementation by a third party.** Our paradigm based on exploiting pre-existing structure requires no semantic knowledge. Consequently, it is relatively easy to achieve a robust implementation. Moreover, the retrieval service can be provided by a third party that does not own or understand the corpus.

(A preliminary empirical comparison of Organic and Synthetic NAR is presented in Section 5.5.)

1.3 Contributions

The contributions of this paper are:

- Formal model of navigation-aided retrieval (Section 3).
- Implementation techniques for a NAR-based retrieval system (Section 4).
- Empirical evaluation via a user study (Section 5).

Related work is discussed next.

2. RELATED WORK

We have already discussed work on presenting query results in the context of dynamically-synthesized structure, in Section 1.2. There is a relatively small body of work that shares our approach of leveraging pre-existing structure, which we discuss here. We consider two distinct subbodies in turn: (1) finding starting points for exploration, and (2) providing guidance during navigation.

2.1 Selecting Starting Points

Best Trails [39] is a retrieval system which selects starting points in response to queries, however, it restricts them to be documents matching the query. The scoring function proposed is ad-hoc in nature and does not take into account navigability factors, which are central to our work. Further, its user interface departs substantially from the traditional query/browse interface and is difficult to use (as reported by the user study in [39].) In contrast, our prototype Organic NAR system closely adheres to familiar interfaces offered by popular query and browse tools.

The hypertext retrieval paradigm known as *topic distillation* [5, 7, 8] bears some similarity to NAR. Topic distillation aims to identify a small number of high-quality documents that are representative of a broad topic area (e.g., bicycling). Much of the work to date on topic distillation uses as a primitive the HITS algorithm [22], which identifies bipartite subgraphs consisting of *hubs* (related to our notion of starting points) and *authorities*. Under the HITS model, good hubs are those that have many links to good authorities. Symmetrically, good authorities are those that are linked to by many good hubs.

HITS-based approaches are inherently effective only for broad topic areas for which there are many hubs and authorities. NAR does not share this limitation, and in principle can accommodate arbitrarily narrow search tasks (see Section 5.3). Furthermore, a NAR starting point is more general than a HITS hub in the following ways:

- A starting point may be multiple links away from documents matching the query.
- It is easy to navigate outward from a starting point (i.e., the link anchor texts are informative and indicative of the documents reachable via the link).

There exist specific approaches [25, 6] to address each of these concerns, but none of them is as general as NAR.

The work of [21] aims to identify nodes of maximum influence in a social network, where the definition of influential nodes is related to our notion of good starting points. As with all prior work we are aware of, a key difference is that our work takes navigability factors into account when evaluating potential starting points.

2.2 Guiding Navigation

WebWatcher [20] highlights hyperlinks along paths taken by previous users who had posed similar queries. This approach may not be suitable for open-ended search tasks where the query is not a good representation of the underlying search task. Letizia [24] and Personal WebWatcher [26] do not rely on queries but instead passively observe the user's browsing behavior in order to learn a model of his search task, and highlight links that match the inferred task. This approach can in principle be incorporated into a NAR system, to augment guidance based on explicit queries.

Highlighting of hyperlinks based on an explicit query, as a method of navigation guidance, was evaluated via a user study in our previous work [28]. Guided navigation was found to result in significantly faster completion of certain search tasks compared to traditional query and browsing interfaces, assuming the user already knows of a suitable starting point. Automatic identification of good starting points is the focus of the present paper.

Symbol	Meaning
D	set of documents in the corpus
T	user's search task
$\mathcal{S}_T$	answer set for search task T, i.e., the set of documents any of which is sufficient to complete the search task T
$\mathcal{Q}_T$	set of valid queries for search task T, i.e., the (possibly infinite) set of queries that might be posed by a user performing T

Table 1: Table of symbols used in the NAR model

3. NAVIGATION-AIDED RETRIEVAL MODEL

A navigation-aided retrieval *system* takes as input a user query, and returns one of: (1) a synthetic starting point document (in the case of a Synthetic NAR system), or (2) an ordered list of links to organic starting points (in the case of an Organic NAR system). Subsequent navigation on the part of the user is performed under guidance from the retrieval system based on the user's query.

In this paper we focus on Organic NAR. Our aim is to develop a numeric *scoring function* for ranking of organic starting point documents. We first supply an abstract formulation (Section 3.1), and then explore concrete instantiations (Section 3.2).

3.1 Generic Model

We now describe the basic framework and terminology underlying the NAR model. A quick reference for the symbols used is provided in Table 3.1.

Let $\mathcal{D}$ be the set of documents in the searchable corpus. Let T denote the user's underlying search task, which is unknown to the retrieval system (and perhaps not fully known to the user himself at the outset). We assume the user is able to complete a search task by viewing exactly one document[4] in $\mathcal{D}$. The *answer set* ($\mathcal{S}_T$) for a search task T is defined as the set of all documents, any one of which suffices to complete the search task T. For a given search task T, a *valid query* is one which might be posed by a user performing T. The (potentially infinite) set of all valid queries is denoted by $\mathcal{Q}_T = \{q_1, q_2, \ldots\}$.

The NAR scoring function is based on two distinct "submodels" that account for query relevance and user navigation, respectively:

Query submodel. A query submodel provides a belief distribution for the answer set $\mathcal{S}_T$, given that the user reveals a query q to be a member of $\mathcal{Q}_T$. It is used to estimate the likelihood that a particular document d results in completion of the task, i.e., $Pr\{d \in \mathcal{S}_T \mid q \in \mathcal{Q}_T\}$.

Navigation submodel. A navigation submodel assesses the likelihood with which a user starting from document d is able to navigate to a document $d' \in \mathcal{S}_T$, thereby successfully completing the search task. To be precise, let $d \rightsquigarrow d'$ denote the event that a user performing the search task T navigates successfully from document d to document d', under guidance $G(q)$ provided for query q. A navigation submodel is used to estimate $Pr\{d \rightsquigarrow d' \mid d' \in \mathcal{S}_T, q \in \mathcal{Q}_T\}$.

[4]In general, a search task might require visiting multiple pages for completion; we believe our model can be extended to handle such situations.

Our goal is to formulate a scoring function, such that document d is given a score equal to the probability that the user completes his search task if he starts navigation at document d. We denote the scoring function as $\sigma(d, q)$. Thus,

$$\begin{aligned}
&\sigma(d,q)\\
&= Pr\{\text{ user completes T } \mid \text{ user starts navigating at } d\}\\
&= \sum_{d' \in \mathcal{D}} Pr\{d \rightsquigarrow d', d' \in \mathcal{S}_T \mid q \in \mathcal{Q}_T\}\\
&= \sum_{d' \in \mathcal{D}} \left[\begin{array}{l} Pr\{d' \in \mathcal{S}_T \mid q \in \mathcal{Q}_T\} \times \\ Pr\{d \rightsquigarrow d' \mid d' \in \mathcal{S}_T, q \in \mathcal{Q}_T\} \end{array} \right]
\end{aligned}$$

3.2 Instantiations of Generic Model

We supply two instantiations of the generic NAR model given in Section 3.1: the conventional probabilistic retrieval model, which does not anticipate navigation outward from query results (Section 3.2.1), and a new approach that incorporates a model of possible post-query user navigation actions (Section 3.2.2).

3.2.1 Conventional Probabilistic IR Model

Our generic NAR model reduces to the classical probabilistic IR model if we apply the following assumption:

Conventional IR assumption. The user has no propensity to navigate outward from any of the retrieved documents, i.e.,

$$\begin{aligned}
Pr\{d \rightsquigarrow d' \mid d' \in \mathcal{S}_T, q \in \mathcal{Q}_T\} &= 1 \quad \text{if } d' = d\\
&= 0 \quad \text{otherwise}
\end{aligned}$$

Under this assumption, the NAR scoring metric reduces to:

$$\sigma_c(d, q) = Pr\{d \in \mathcal{S}_T \mid q \in \mathcal{Q}_T\}$$

Recall that $Pr\{d \in \mathcal{S}_T \mid q \in \mathcal{Q}_T\}$ represents the probability that document d is relevant to the search task underlying query q. Several methods have been proposed in the probabilistic IR literature for estimating this term, e.g., [23, 35].

3.2.2 Navigation-Conscious Model

Now, we present an instantiation of the NAR model which takes into account the possibility of the user navigating further from the results retrieved for a query. We first instantiate the two submodels and then derive a formula for the scoring function, under these submodels.

Query submodel. Any probabilistic IR relevance ranking function can be used to estimate the term $Pr\{d' \in \mathcal{S}_T \mid q \in \mathcal{Q}_T\}$. We refer to such an estimate as $R(d', q)$.

Navigation submodel. As mentioned before, the navigation submodel estimates the probability of a user browsing from a given document to another document while performing a search task. To estimate this quantity, we adopt the stochastic model of user navigation behavior called *WUFIS* proposed by Chi et al. [9, 10]. WUFIS is based on the theory of *information scent* [29], and has been validated against real user traces [10].

At the heart of WUFIS lies a function $W(N, d_1, d_2)$, which gives the probability that a user whose search task is characterized by *information need* N navigates successfully along some path from document d_1 to document d_2 (N is encoded as a term vector). Every hyperlink, via anchor and surrounding context, provides some information scent regard-

ing the content reachable by clicking that hyperlink. WUFIS assumes that the probability of a user following a hyperlink depends on how well his information need matches the information scent provided by the hyperlink. The information scent is approximated by a weighted vector of terms occurring in the anchor text and optionally in a small window around the hyperlink. The cosine similarity between the information need vector and the information scent vector is used to estimate the probability that the user follows the given hyperlink, along with an assumed *attrition probability* $0 < \alpha \leq 1$ (set to 0.85 in our experiements). The navigation probability between two documents d_1 and d_2 is given by the product of link navigation probabilities along the sequence of links connecting d_1 to d_2.

We propose to use WUFIS as the basis for our navigation submodel.[5] Constructing an estimate of N from a document in the answer set $\mathcal{S}_T$ is a challenge in itself, but rather orthogonal to the focus of our work. Hence, we take the naive approach of approximating N by extracting the first k terms from the document. Better ways of extracting the information need will likely lead to an improvement in performance.

Let $N(d)$ denote the information need extracted from some d assumed to be a member of $\mathcal{S}_T$. We instantiate our navigation submodel by setting $Pr\{d \rightsquigarrow d' \mid d' \in \mathcal{S}_T, q \in \mathcal{Q}_T\} = W(N(d'), d, d')$.

Scoring function. When we combine our query submodel with our navigation submodel, we arrive at the following starting point scoring function:

$$\sigma_n(d, q) = \sum_{d' \in D} R(d', q) \times W(N(d'), d, d') \qquad (1)$$

The above formula makes intuitive sense — the two terms embody the two key factors in assessing the likelihood of successful task completion, loosely speaking:

1. the number of documents reachable from d that are relevant to the search task, and
2. the ease and accuracy with which the user is able to navigate to those documents.

4. VOLANT: A PROTOTYPE ORGANIC NAR SYSTEM

We built a prototype Organic NAR system called *Volant*, which implements the navigation-conscious model instantiation described in Section 3.2.2. In this section, we describe its design and implementation.

4.1 Overview

Figure 2 provides an overview of Volant's architecture. Volant consists of three components: *content engine*, *connectivity engine*, and *intermediary*. The content engine supports retrieval and ranking of Web documents, via a text index. The connectivity engine maintains a specialized index of linkage between pairs of documents. The intermediary intercepts HTTP requests from the user and communicates with the connectivity and content engines as well as the WWW in order to produce the desired response — either starting points or Web pages enhanced with navigation guidance.

Implementation of a NAR system is not the focus of this paper, and so we heavily relied on third party softwares to build Volant. The Lucene text indexing and retrieval systemserved as Volant's content engine. The connectivity server was built on top of a MySQL database [27], while the intermediary was coded using Java servlets running on an Apache Tomcat server [37]. Exploring more efficient implementation options (e.g., the connectivity server in [4]) would be valuable future work.

Next, in Section 4.2, we describe the index structures Volant creates offline via preprocessing the corpus. Then, we describe how Volant selects starting points in response to keyword queries (Section 4.3) and adds navigation guidance to subsequent Web pages that the user visits (Section 4.4). We discuss performance and scalability issues briefly in Section 4.5.

4.2 Preprocessing

4.2.1 Content Engine

As mentioned before, Lucene serves as Volant's content engine, and is used to estimate the value of $R(d, q)$, the probability that document d is relevant to the query q. We are not aware of any ranking function which can compute the exact probability of a document being relevant to a query. However, certain probabilistic models, which produce a score approximately proportional to logarithm of the relevance probability, have proved to be effective for retrieval. We follow state-of-the-art and use the Okapi BM25 scoring function [32] to estimate $R(d, q)$. This approximation can lead to suboptimal results, however formulating a scoring function which produces exact relevance probabilities is not the focus of our work.

Lucene only supports scoring functions based on the Vector Space Model [1]. We incorporated the Okapi BM25 function into Lucene's source code for document ranking. The precise term weighting formula we used is:

$$w_{td} = tf_d \times \frac{\log\left(\frac{N-n+0.5}{n+0.5}\right)}{k_1 \times ((1-b) + b \times \frac{dl}{avdl}) + tf_d}$$

w_{td}	=	weight of term t in document d
tf_d	=	frequency of term t in document d
N	=	total number of documents in the corpus
n	=	number of documents matching term t
dl	=	length of document d
$avdl$	=	average length of a document in the corpus
b, k_1	=	tuning parameters

We set $b = 0.75, k_1 = 2$, since these values have been effectively used by other retrieval systems [13, 18, 31].

4.2.2 Connectivity Engine

The connectivity engine is used to estimate the value of $W(N(d_2), d_1, d_2)$, i.e., the probability of a user with information need $N(d_2)$ successfully navigating from d_1 to d_2 under the WUFIS model.

Let $W_\Pi(N(d_2), d_1, d_2)$ denote the probability of a user with information need $N(d_2)$ successfully navigating from d_1 to d_2 along path Π. Let Π^* denote the path for which this value is maximized. In order to keep the computation feasible, we assume that Π^* is the only path connecting d_1

[5]Here we ignore the influence of navigation *guidance*, because we have no reliable basis for assuming any particular effect. As future work we plan to extend WUFIS to account for the additional "scent" provided by navigation guidance.

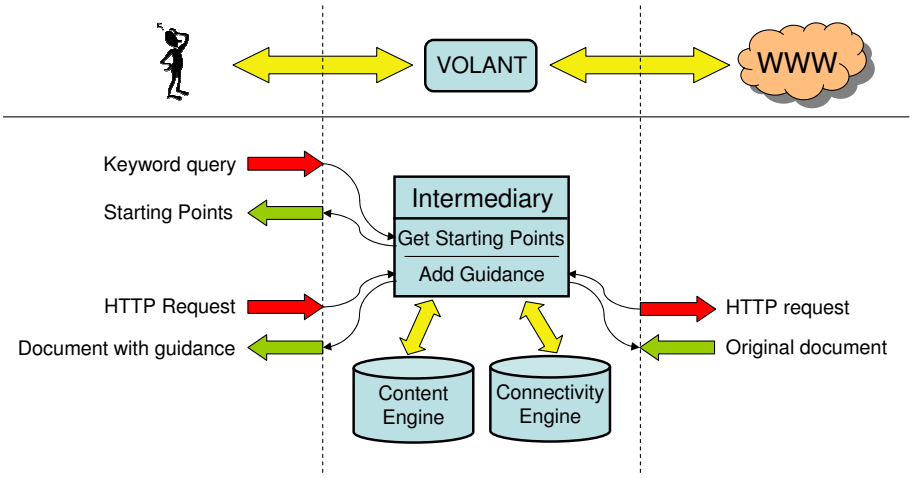

Figure 2: Volant architecture (overview appears above the horizontal line; details appear below).

to d_2, ignoring other "low-scent" paths between d_1 and d_2. Let d_W denote the document immediately following d_1 along Π^*.

In the preprocessing stage, for each ordered pair $\langle d_1, d_2 \rangle$ of documents in the corpus, a variant of Dijkstra's algorithm [11] is used to generate tuples of the form $\langle d_1, d_2, d_W, W(N(d_2), d_1, d_2) \rangle$. These tuples are then indexed so as to optimize lookups based on d_2 (in our case, we build database indexes on appropriate columns of our MySQL tables).

4.3 Selecting Starting Points

At runtime, when a query q is received, Volant constructs a ranked list of starting points as follows:

1. Retrieve from the content engine all documents d' for which $R(d', q) > 0$.
2. For each document d' retrieved in Step 1, retrieve from the connectivity engine all documents d for which $W(N(d'), d, d') > 0$.
3. For each unique document d identified in Step 2, compute the starting point score $\sigma_n(d, q)$.
4. Sort the documents in decreasing order of $\sigma_n(d, q)$; truncate after the top k documents.

4.4 Adding Navigation Guidance

Given query q, and document d requested by the user, Volant intercepts and modifies d by highlighting all links on d that lead to documents relevant to q. The procedure for determining what links on d to highlight is as follows:

1. Retrieve from the content engine all documents d' for which $R(d', q) \geq T$, where $T > 0$ is a fixed relevance threshold (we used $T = 0.1$ in our experiments).
2. For each document d' retrieved in Step 1, retrieve from the connectivity engine all tuples where $W(N(d'), d, d') > 0$.
3. For each $\langle d, d', d_W, W(N(d'), d, d') \rangle$ tuple retrieved in Step 2, highlight links on d that point to d_W.

4.5 Efficiency and Scalability

The primary aim of this paper is to introduce the NAR model and evaluate its effectiveness (our next topic, in Section 5). That said, we discuss efficiency and scalability issues here briefly.

We have used Volant with a corpus of over 1.5 million documents with over eight million hyperlinks (the .GOV collection; see Section 5). Once the content and connectivity indexes have been created, the online components (starting point selection and link highlighting) execute at interactive speeds. Not surprisingly, however, the preprocessing stage (Section 4.2) is problematic in terms of scalability. In particular, creation of the connectivity index requires listing all pairs of documents. For a very large corpus it becomes infeasible to consider all such pairs. Devising principled techniques to limit the number of document pairs considered based on the likelihood of finding paths with strong scent is an important topic for future work.

5. EVALUATION

Having presented our formal navigation-aided retrieval model and an overview of our implementation techniques, we turn now to evaluation.

5.1 Experimental Hypotheses

In addition to evaluating Volant's effectiveness in combined query/navigation settings, we seek to study Volant's behavior in scenarios that do *not* entail navigation. Also of interest is the effectiveness of Organic NAR approaches (such as Volant) relative to that of Synthetic NAR approaches. Ideal experiments would test the following three hypotheses:

- **Hypothesis 1:** In query-only scenarios, Volant does not perform significantly worse than conventional retrieval approaches.
- **Hypothesis 2:** In combined query/navigation scenarios, Volant selects high-quality starting points. Quality is further enhanced by query-driven navigation guidance.
- **Hypothesis 3:** In a significant fraction of query/navigation scenarios, the best organic starting point is of higher

BM25 MAP score	Volant MAP score	p-value
0.23	0.20	0.22

Table 2: Performance on unambiguous queries.

quality than one that can be synthesized using existing techniques.

The experiments we present in this paper represent preliminary efforts toward validating the above hypotheses. Before we describe our experiments, we introduce the search tasks we used as test sets.

5.2 Search Task Test Sets

It is not immediately clear how one would identify navigation-prone scenarios a priori, given a search task description or query. As a proxy we rely on the notion of *query clarity* [14], which measures query ambiguity (a high clarity score indicates low ambiguity). Query clarity has been used to predict retrieval effectiveness. Queries for which retrieval is ineffective are likely to be followed by query refinement and/or navigation (which may be thought of as implicit refinement). Hence we posit that low query clarity serves as a reasonable first-order indicator of navigation-prone scenarios. We used the *Simplified Clarity Score* (SCS) metric, which has been shown to predict retrieval effectiveness well in the case of short queries [19], to guide our selection of two sets of search tasks and corresponding queries:

- **Unambiguous:** The 20 search tasks of highest clarity from the TREC 2000 Web track[6] [17], using the "title" field as the query. (Average clarity of top-20 = 9.6.) These tasks are over the TREC WT10G corpus.
- **Ambiguous:** 48 randomly-selected tasks from the TREC 2003 Web topic distillation track [38], which is geared toward open-ended exploration of broad topics, again using the "title" field as the query. (Average clarity = 4.5.) These search tasks are over the TREC .GOV corpus.

The WT10G and .GOV test collections have been shown to be structurally representative of the real Web [34].

Note that the search tasks in the topic distillation track are almost certain to be broad, open-ended and ambiguous in nature – we just use the clarity score as an added support for our belief. To prevent misclassifications in the Web track search task set, we manually screened the search tasks, and retained only those which were clearly unambiguous in nature.

In our experiments, we use the **unambiguous** test set to represent "query-only" scenarios, and the **ambiguous** test set for "combined query/navigation" scenarios.

5.3 Performance on Unambiguous Queries

As a test of Hypothesis 1 (Section 5.1), we measured mean average precision (MAP) scores on the **unambiguous** test set (Section 5.2). Results are reported in Table 2. Performance did not differ significantly. Indeed, for these queries the ranked query result lists produced by Volant were nearly identical to those produced by BM25. The reason is that relevant documents tended not to be siblings or close cousins in the linkage structure. Consequently, Volant deemed that the best starting points were the relevant documents themselves or, more precisely, documents estimated to have the highest chance of being relevant according to BM25.

[6]Ad-hoc retrieval tasks with varying degrees of ambiguity.

5.4 Performance on Ambiguous Queries

Our main experiment tests Hypothesis 2 (Section 5.1), i.e., how well Volant performs in scenarios for which it was conceived. This experiment uses the **ambiguous** test set (Section 5.2) to measure the usefulness of starting points provided by Volant in performing a search task via navigation. Unfortunately, the relevance judgments provided by TREC are not suitable for this experiment, because they merely identify documents that constitute the root of a relevant subsite (or some other form of structural hub)—they provide no indication of how useful each such hub would be in performing the search task. Furthermore, we cannot test the efficacy of navigation guidance (which annotates documents in a query-specific manner) using the TREC judgments. Therefore, we ourselves designed and conducted a user study in order to validate Hypothesis 2.

5.4.1 User Study Design

Participants. The study comprised of 48 human judges, who were students at CMU familiar with the Web and Web searching, but with no knowledge of our project. Each participant was assigned a search task (selected at random without replacement), and asked to judge the suitability of various documents as starting points for this search task (a within-subjects design). A given starting point was judged by exactly one participant.

Evaluation criteria. The participants were encouraged to explore the neighborhood of the starting point while forming their assessment. Ratings were taken for four criteria, each on a scale from 1 to 5:

- **Breadth:** Would a broad spectrum of people, who are interested in different aspects of the search task, be satisfied with the information obtainable via the starting point?
- **Accessibility:** Given a user with a particular specialized interest, would he be able to navigate *easily* from the starting point to material that suits his interest?
- **Appeal:** Do you like the way the material is organized and presented? [7]
- **Usefulness:** Would most people be able to complete their task successfully using this starting point?

The breadth, accessibility and appeal criteria focus on specific factors, whereas the usefulness criterion is intended to elicit an overall assessment.

In this experiment, to maximize the information gathered from the human subjects we configured Volant to prune redundant starting points in the following way: documents reachable (with high navigation probability) from the top-ranked starting point are factored out when selecting the second-ranked starting point, and so on. In a real-world deployment it may be desirable to prune query results in such a manner, and indeed to enforce diversity in a more general sense[8]. An in-depth study of methods for ensuring diversity

[7]Based on a posteriori discussions with some of the subjects, we determined that appeal may have been influenced by the fact that documents lacked the embedded images that would normally have been present (images were not included in the corpus used for the TREC 2003 Web topic distillation track).

[8]Most popular commercial search engines already provide such features.

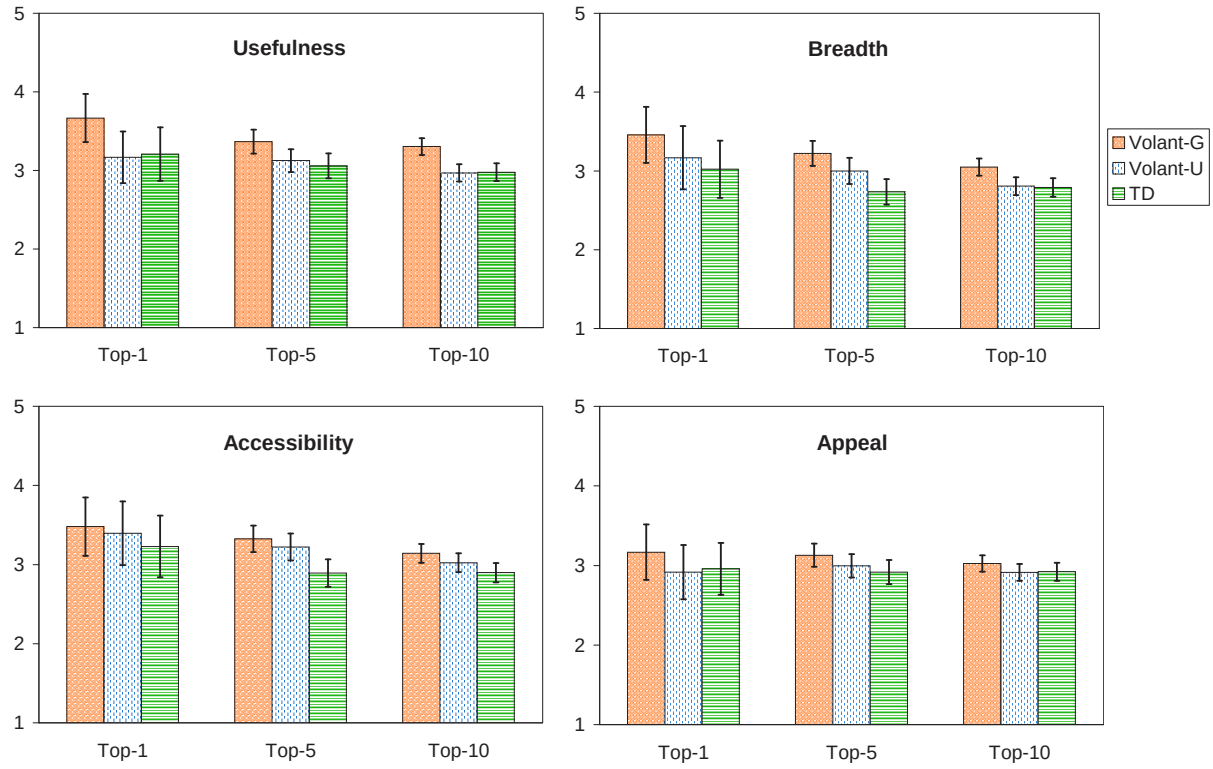

Figure 3: Performance on ambiguous queries: starting point ratings averaged across all tasks/participants (with 95% confidence intervals).

in NAR query results is beyond the scope of this paper, and is left as future work.

Starting points evaluated. For each search task, the following sets of starting points were included in the evaluation: First, we included the top ten starting points produced by Volant (these have hyperlink highlighting). We also included a copy of this top-10 set with link highlighting removed, to allow us to isolate the impact of navigation guidance in our measurements. These 20 starting points constitute Organic NAR results produced by our methods.

Since there is no prior work on Organic NAR, we added the top ten results of the the best performing algorithm in the TREC 2003 Web topic distillation track [12]. This algorithm, developed by CSIRO, is a fine-tuned implementation of a sophisticated topic distillation technique, making use of a variety of document features such as url length, anchor text, off-site/on-site in-degrees and out-degrees, etc.

The total of 30 starting points thus obtained were randomly ordered before presentation, and participants were not told how the starting points were selected.

Effort. Each participant was asked to judge a total of 30 starting points. Since a participant was expected to explore the neighborhood of a starting point before making an assessment, we allocated ten minutes per starting point, which meant that each participant devoted 300 minutes (or five hours) for this study. To prevent fatigue, we distributed the study over five days, wherein each participant was required to spend only an hour each day judging starting points. Physical constraints (rooms, study administrators, etc.) limited the number of participants operating simultaneously in a single session to ten. Therefore, we conducted five sessions of ten users each per day, over a period of five days.

This study involved a substantial investment of time, effort and money. The number of participants (and hence search tasks), the number of results considered for each algorithm and the number of algorithms compared were severely restricted by our resources. The decision to use only the top ten results of each algorithm allowed us to test several algorithms/variants, without excessive burden. We feel this decision is reasonable, given that users of commercial search engines typically only view the top ten results.

5.4.2 *Results*

Figure 3 shows the starting point ratings, averaged across all search tasks (and hence across all participants). Each metric is plotted on a separate graph. In each graph, the leftmost group shows the results for the top-ranked starting points; the center group shows the average across the top five starting points; the rightmost group shows the average across the top ten. The label "Volant-G" corresponds to Volant with navigation guidance; "Volant-U" corresponds to Volant with no guidance; "TD" corresponds to the topic distillation algorithm[9].

We used the paired t-test to identify differences that carry statistical significance (using $p = 0.05$ as the cutoff). In no case does TD perform better than either of Volant-U or Volant-G with statistical significance. There are cases

[9]Contrary to what one would expect, Volant-G and Volant-U did not receive identical breadth ratings (the difference was statistically significant for Top-5 and Top-10). We attribute this difference to a discrepancy in *perceived* breadth.

in which the converse is true, but not across all three rank groups. The only statistically significant difference across all three rank groups is that Volant-G consistently outperforms each of Volant-U and TD in terms of usefulness. [10]

5.4.3 Conclusions

First, this experiment confirms the usefulness of navigation guidance. (Determining the particular form of navigation guidance that works best is left as future work.)

Second, in the absence of navigation guidance Volant performs on par with the best known topic distillation algorithm, even though we performed almost no tuning of Volant prior to the experiment. While the two algorithms have similar performance, Volant offers three important advantages:

- Unlike the topic distillation algorithm, which uses a combination of heuristics and has been tuned extensively, Volant is based on a clean theoretical model.
- Volant invokes a conventional retrieval function as a subroutine, so advances in conventional retrieval technology are trivial to incorporate into Volant to achieve a corresponding improvement.
- As indicated by our first experiment (Section 5.3), Volant is suitable for unambiguous queries as well as ambiguous ones. This property may obviate the need to choose between two retrieval algorithms (e.g., conventional IR and topic distillation) for each query.

5.5 Organic versus Synthetic NAR

To obtain a preliminary understanding of the relationship between Organic and Synthetic NAR, we injected a synthetic starting point document into the set of starting points evaluated by each of our human subjects. We used the well-known search result clustering algorithm of Zamir and Etzioni [40] to generate synthetic starting points. Each cluster produced by the algorithm is treated as a starting point, with links to the documents belonging to that cluster. Our results so far are inconclusive.

On the one hand, if we compare the top-ranked starting point suggested by Volant against the synthetic starting point generated by the clustering algorithm, we find that for breadth, accessibility and appeal, clustering outperforms Volant by a statistically significant margin. A significant difference for usefulness was not found. Examining usefulness on a task-by-task basis, we find that the clustering starting point was rated more useful in 40% of the search tasks, while Volant's top starting point was rated more useful in 17% of the search tasks (in 43% of the search tasks there was a tie).[11]

On the other hand, if we broaden our focus to include all ratings available, we find that for at least 44% of the search tasks the corpus contains a document that, when navigation guidance is added, is considered more useful than the clustering starting point. If we also include documents that were rated as equally useful, that figure rises to 96%.

[10] The p-values for Volant-G versus Volant-U are all below 0.005 (highly significant); for Volant-G versus TD, all are below 0.005 (again, highly significant) except for Top-1, where $p = 0.04$ (borderline significant).

[11] It is unknown whether these cases arise due to characteristics of the search task, the subject's personal tastes, Volant's manner of ranking starting points or, more likely, some combination of factors.

6. SUMMARY AND FUTURE WORK

This paper introduces a new document retrieval paradigm called navigation-aided retrieval (NAR), designed to handle poorly-specified and open-ended search tasks in hypertext environments. These situations usually require a mix of querying and navigation on the part of the user. As such, NAR aims to integrate these two complementary search tactics, which to date have primarily been treated separately.

Effectiveness. The salient properties of the NAR paradigm are: (1) responding to queries by positioning users at suitable starting points for exploratory navigation; (2) helping to guide navigation in a query-driven fashion. Our experiments showed that the scoring function implied by our NAR model performs as well as the best-known topic distillation algorithm at selecting suitable starting points, and that our navigation guidance mechanism is of significant benefit compared with no guidance.

Relationship to conventional IR. Our formal model of navigation-aided retrieval constitutes a strict generalization of the conventional probabilistic IR model. Furthermore, our empirical work suggests that in the case of unambiguous queries for which conventional IR techniques are sufficient, NAR reduces to standard IR *automatically*. This property, if confirmed through further experiments, would obviate the need to choose from two alternative retrieval methods based on the nature of the search task.

Relationship to synthetic approaches. Search result clustering can be thought of as form of navigation-aided retrieval, as the user is expected to navigate within the dynamically constructed topic hierarchy before converging on the desired document(s). We characterize methods of this form as *Synthetic* NAR, whereas our method can be thought of as *Organic* NAR. Although Synthetic NAR has the obvious advantage of being unconstrained by pre-existing structure, in our (albeit preliminary) experiments we found that Organic NAR has significant potential. Specifically, in nearly all cases (96%) the best organic starting point was rated to be at least as useful as the topic hierarchy created by a well regarded clustering algorithm. Furthermore, in a significant fraction of cases (44%) the best organic starting point received a higher rating than the clustering result.

Future work. While these initial results are encouraging, there is more work to be done. First, to perform well over corpora that contain substantial redundancy, it may be appropriate to generate only *topically diverse* starting points. Second, we would like to refine our scoring function so that it can be applied to synthetic starting points as well as organic ones, while still yielding good predictions. Ultimately, we envision a unified method that presents the user with directly relevant documents (in the case of unambiguous queries), organic starting points for exploration (in the case of ambiguous queries), or synthetic starting points (in the case that the corpus does not contain suitable organic ones).

Acknowledgements

We gratefully acknowledge the assistance of Gary Hsieh, Dhananjay Khaitan, Paul Ogilvie, Sandeep Pandey, Samuel Wang and Andrew Yang with the extensive implementation and evaluation effort that went into this project. We also

thank Jamie Callan, Rosie Jones and Malcolm Slaney for helpful discussions and feedback.

7. REFERENCES

[1] R. A. Baeza-Yates and B. A. Ribeiro-Neto. *Modern Information Retrieval*.

[2] M. J. Bates. The Design of Browsing and Berrypicking Techniques for the On-line Search Interface. *Online Review*, 13:407–431, 1989.

[3] N. J. Belkin. Anomalous States of Knowledge as the Basis of Information Retrieval. *Canadian Journal of Information Science*, 5:133–143, 1980.

[4] K. Bharat, A. Broder, M. Henzinger, P. Kumar, , and S. Venkatasubramanian. The Connectivity Server: fast access to linkage information on the Web. In *Proc. WWW*, 1998.

[5] K. Bharat and M. Henzinger. Improved algorithms for topic distillation in a hyperlinked environment. In *Proc. SIGIR*, 1998.

[6] S. Chakrabarti, B. Dom, D. Gibson, J. Kleinberg, P. Raghavan, and S. Rajagopalan. Automatic resource list compilation by analyzing hyperlink structure and associated text. In *Proc. WWW*, 1998.

[7] S. Chakrabarti, B. E. Dom, D. Gibson, R. Kumar, P. Raghavan, S. Rajagopalan, and A. Tomkins. Experiments in topic distillation. In *Proc. SIGIR Workshop on Hypertext Information Retrieval on the Web*, 1998.

[8] S. Chakrabarti, M. Joshi, and V. Tawde. Enhanced topic distillation using text, markup tags, and hyperlinks. In *Proc. SIGIR*, 2001.

[9] E. H. Chi, P. Pirolli, K. Chen, and J. Pitkow. Using information scent to model user information needs and actions and the Web. In *Proc. SIGCHI*, 2001.

[10] E. H. Chi, A. Rosien, G. Supattanasiri, A. Williams, C. Royer, C. Chow, E. Robles, B. Dalal, J. Chen, and S. Cousins. The Bloodhound project: Automating discovery of web usability issues using the InfoScent simulator. In *Proc. SIGCHI*, 2003.

[11] T. H. Cormen, C. E. Leiserson, R. L. Rivest, and C. Stein. *Introduction to Algorithms. MIT Press/McGraw-Hill*. second edition, 2001.

[12] N. Craswell, D. Hawking, A. McLean, T. Upstill, R. Wilkinson, and M. Wu. TREC12 web and interactive track at CSIRO, 2003.

[13] N. Craswell, D. Hawking, and S. Robertson. Effective Site Finding Using Link Anchor Information. In *Research and Development in Information Retrieval*, pages 250–257, 2001.

[14] S. Cronen-Townsend, Y. Zhou, and W. B. Croft. Predicting query performance. In *Proc. SIGIR*, 2002.

[15] S. Dumais, E. Cutrell, and H. Chen. Optimizing search by showing results in context. In *Proc. SIGCHI*, 2001.

[16] J. English, M. Hearst, R. Sinha, K. Swearingen, and K. Yee. Flexible Search and Browsing using Faceted Metadata. `http://bailando.sims.berkeley.edu/papers/flamenco02.pdf`, Jan. 2002.

[17] D. Hawking. Overview of the TREC-9 web track.

[18] D. Hawking, T. Upstill, and N. Craswell. Toward better weighting of anchors. In *Proc. SIGIR*, 2004.

[19] B. He and I. Ounis. Inferring query performance using pre-retrieval predictors. In *Proc. Symposium on String Processing and Information Retrieval*, 2004.

[20] T. Joachims, D. Freitag, and T. Mitchell. WebWatcher: A tour guide for the World Wide Web. In *Proc. International Joint Conference on Artificial Intelligence (IJCAI)*, 1997.

[21] D. Kempe, J. Kleinberg, and E. Tardos. Maximizing the spread of influence through a social network. In *Proc. SIGKDD*, 2003.

[22] J. M. Kleinberg. Authoritative sources in a hyperlinked environment. *J. ACM*, 46(5), 1999.

[23] V. Lavrenko and W. B. Croft. Relevance-based language models. In *Proc. SIGIR*, 2001.

[24] H. Lieberman. Letizia: An agent That assists Web browsing. In *Proc. IJCAI*, 1995.

[25] J. Miller, G. Rae, F. Schaefer, L. Ward, T. LoFaro, and A. Farahat. Modifications of kleinberg's hits algorithm using matrix exponentiation and web log records. In *Proc. SIGIR*, 2001.

[26] D. Mladenic. Using text learning to help Web browsing. In *Proc. SIGCHI*, 2001.

[27] MySQL DBMS. `http://www.mysql.com/`.

[28] C. Olston and E. H. Chi. ScentTrails: Integrating browsing and searching on the Web. *ACM Trans. on Computer-Human Interaction*, 10(3), 2003.

[29] P. Pirolli and S. Card. Information Foraging. *Psychological Review*, 1999.

[30] F. Qiu, Z. Liu, and J. Cho. Analysis of user web traffic with a focus on search activities. In *Proc. International Workshop on the Web and Databases (WebDB)*, June 2005.

[31] S. Robertson, S. Walker, M. Hancock-Beaulieu, A. Gull, and M. Lau. Okapi at TREC. In *Proc. TREC*, 1992.

[32] S. E. Robertson, S. Walker, M. Hancock-Beaulieu, A. Gull, and M. Lau. Okapi at TREC. In *Text REtrieval Conference*, 1992.

[33] V. Sinha and D. R. Karger. Magnet: Supporting navigation in semistructured data environments. In *Proc. SIGMOD*, 2005.

[34] I. Soboroff. Do TREC Web Collections Look Like the Web? *SIGIR Forum*, pages 23–31, 2002.

[35] K. Sparck Jones, S. Walker, and S. Robertson. A probabilistic model of information retrieval: Development and comparative experiments. *Information Processing and Management*, 36, 2000.

[36] J. Teevan, C. Alvarado, M. S. Ackerman, and D. R. Karger. The perfect search engine is not enough: A study of orienteering behavior in directed search. In *Proc. SIGCHI*, 2004.

[37] Apache Tomcat. `http://tomcat.apache.org/`.

[38] TREC-2003 Web Track: Guidelines. `http://es.csiro.au/TRECWeb/guidelines_2003.html`.

[39] R. Wheeldon and M. Levene. The Best Trail algorithm for assisted navigation of Web sites. In *Proc. LA-WEB Conference on Latin American Web Congress*, 2003.

[40] O. Zamir and O. Etzioni. Web document clustering: A feasibility demonstration. In *Proc. SIGIR*, 1998.

[41] H.-J. Zeng, Q.-C. He, Z. Chen, W.-Y. Ma, and J. Ma. Learning to cluster web search results. In *Proc. SIGIR*, 2004.