Studying Trailfinding Algorithms for Enhanced Web Search

Adish Singla
Microsoft Bing
Bellevue, WA 98004 USA
adishs@microsoft.com

Ryen W. White
Microsoft Research
Redmond, WA 98052 USA
ryenw@microsoft.com

Jeff Huang
University of Washington
Seattle, WA 98195 USA
sigir@jeffhuang.com

ABSTRACT
Search engines return ranked lists of Web pages in response to queries. These pages are starting points for post-query navigation, but may be insufficient for search tasks involving multiple steps. Search trails mined from toolbar logs start with a query and contain pages visited by one user during post-query navigation. Implicit endorsements from many trails can enhance result ranking. Rather than using trails solely to improve ranking, it may also be worth providing trail information directly to users. In this paper, we quantify the benefit that users currently obtain from trail-following and compare different methods for finding the best trail for a given query and each top-ranked result. We compare the relevance, topic coverage, topic diversity, and utility of trails selected using different methods, and break out findings by factors such as query type and origin relevance. Our findings demonstrate value in trails, highlight interesting differences in the performance of trailfinding algorithms, and show we can find best-trails for a query that outperform the trails most users follow. Findings have implications for enhancing Web information seeking using trails.

Categories and Subject Descriptors
H.3.3 [**Information Storage and Retrieval**]: Information Search and Retrieval – *search process*, *selection process*

General Terms
Algorithms, Experimentation, Human Factors

Keywords
Search trails, trailfinding, best-trail selection

1. INTRODUCTION
Web search engines provide keyword access to Web content. In response to search queries, these engines return lists of Web pages ranked based on estimated relevance. Information retrieval (IR) researchers have worked extensively on algorithms to effectively rank documents (c.f. [20]). However, research in areas such as information foraging [18], berrypicking [2], and orienteering [16], suggests that individual items may be insufficient for vague or complex information needs. In such circumstances, search results represent only the starting points of user exploration [17][21].

Logs containing the search engine interactions of many users have been mined extensively to enhance search-result ranking [1][13]. Richer log data from sources such as browser toolbars offers insight into the behavior of many users beyond search engines. Search trails comprising a query and post-query page views can be mined from these logs [28]. Although trail components—origins (clicked search results) and destinations (trail end points [27]) have been used previously to support search, the typical application of trails is to better rank Web pages [1][3]. In *As We May Think* [4], Vannevar Bush envisioned using trails marked and willingly shared by trailblazing users to guide others. Joachims et al. [14] suggest that in many cases, only a sequence of pages and the knowledge about how they relate can satisfy a user's information need. This suggests that trails should be a unit of retrieval, or at least shown to users on the search engine result page (SERP). Although others have investigated trail generation for site or hypertext navigation [11][25], the challenge of finding the best trails to show to users directly on the SERP is unaddressed.

In this paper we present a log-based study of trailfinding for Web search. We mine trails from logs and investigate the value that full trails bring to users over the trail origins (i.e., the search results). We then represent trails as graphs and create algorithms to find the best trail for each search result—so-called *trailfinding*—using graph properties such as breadth, depth, and strength. Since "best" may be task dependent, we use a variety of metrics to evaluate the trails found. Our study answers the following questions: (i) How much benefit do users gather from following trails versus stopping after the origin page? (ii) Which trailfinding algorithms perform best? (iii) Can we extend our algorithms to handle unseen queries? We conduct this study using a log-based methodology since logs contain evidence of real user behaviors at scale and provide coverage of many types of information needs. Information need coverage is important since differences in algorithm performance may not hold for all search tasks. Our findings demonstrate value in trails, interesting differences in the performance of the algorithms, and performance tradeoffs when moving beyond logs to handle unseen queries using term matching.

2. RELATED WORK
A search trail consists of an origin page, intermediate pages, and a destination page. Origin pages are the search results that start a search trail. Query and origin pages from search engine click logs can be used to improve result set relevance [13]. Agichtein et al. [1] and Bilenko and White [3] found that using trails as endorsements for trail pages helped search engines learn to rank search results more effectively. The goal of their research was to improve ranking rather than show trails to users on the SERP. White et al. [27] added trail destination suggestions to the SERP. User study participants found destination suggestions useful. *Our research extends that work to consider the suggestion of full trails rather than only destinations on the SERP*. Prior to adding trails to result pages, we first study a variety of trailfinding methods to find performant algorithms that are worth further testing in user studies.

Systems such as *WebWatcher* [14], *ScentTrails* [15], and *Volant* [17] highlight candidate pages based on models of information needs or user interests. Studies of these systems show that they can improve search speed and search success. Highlighted pages form a trail over time, but the link-at-a-time approach does not expose the user to much needed initial context [14].

SIGIR'10, July 19–23, 2010, Geneva, Switzerland.

Wexelblat and Maes [25] introduced annotations in Web browsers called *footprints*, which are trails through a Website assembled by the site's designer. Their evaluation found that users required significantly fewer steps to find information using their system. Freyne et al. [10] extend footprints by adding icons to links to offer users visual cues. These cues are gathered from past users and include popularity, recency, and annotations. Wang and Zhai [24] continues the footprint metaphor in a topic map that lets users navigate to related queries, and to queries of varying specificity. Simulation studies revealed potential benefit from topic maps.

Guided tours and trails constructed by domain experts have been proposed, mainly in the hypertext community. Hammond and Allison [12] and Trigg [22] proposed guided tours in hypertext to ease problems of user disorientation. Zellweger [30] introduced scripted documents which are more dynamic than guided tours since they have conditional and programmable paths, automated playback, and active entries. Chalmers et al. [5] propose that human "recommenders" construct and share Web navigation paths.

Rather than requiring human intervention, tours and trails can also be generated automatically. Guinan and Smeaton [11] generate a tour for a given query based on term matching for node selection and inter-node relationships (e.g., "is_a", "precedes") for node ordering. In a user study using a collection of lecture materials, they found that users followed these trails closely; 40% of the time, subjects did not deviate from the proposed trail. Wheeldon and Levene [26] propose an algorithm for generating trails to assist in Web navigation. They define trails as trees and expand them from the root node using the expected information gain as the probability of expansion. This gain is based on the term frequency of the query in the document, with a penalty for duplicate URLs. They presented trails using an interface attached to the browser. User study participants found trails to be useful and noted that seeing the relationship between links helped.

We extend previous work in a number of ways: (i) we recommend full trails rather than only suggesting next steps; (ii) we focus on general Web search, where the content is less constrained than Websites or small hypertext collections, and information such as inter-node relationships is typically unavailable, and (iii) we find best-trails based on real user behaviors evident in logs, avoiding the scalability challenges associated with human intervention.

3. TRAILS

In this section, we describe the log data from which trails are extracted, outline trail mining, introduce some trailfinding algorithms, and describe unseen query handling using term matching.

3.1 Log Data

The primary source of data for this study was the anonymized logs of URLs visited by users who consented to provide interaction data through a widely-distributed browser plugin. Log entries include a unique user identifier, a timestamp for each page view, an identifier for each browser instance, and the URL of the Web page visited. Intranet and secure (https) URL visits were excluded at the source to maintain user privacy. Revisits to pages made through the browser "back" button are also captured in the log data. To remove variability caused by geographic and linguistic variation in search behavior, we only include entries generated in the English speaking United States locale. The results described in this paper are based on URL visits during a nine-month period from February 2009 through December 2009 inclusive, representing billions of URL visits from millions of unique users.

3.2 Trail Mining

From these logs, we mined around a billion search trails, each trail followed by a single user. Trails start with a search engine query (which also includes the SERP) followed by a click on one of the search engine results (trail *origin*). Search trails are represented as temporally-ordered URL sequences. Trails terminate once they reach 10 steps (to facilitate more controlled analysis later in the study) or a period of user inactivity of 30 or more minutes (also used in [8]), whichever condition is satisfied first. In our logs, there were 1.4 billion search trails followed by 80 million users. This comprised 314 million unique queries (Q), 226 million unique origins (R), 542 million query-origin pairs, and 1.1 billion unique search trails (T). Figure 1 illustrates three search trails expressed as Web behavior graphs. Each trail starts with the same query ($q1$) and the same origin URL (u_a), then proceeds to different pages. The number in brackets on each node represents its sequence order in the trail based on timestamps of user activity.

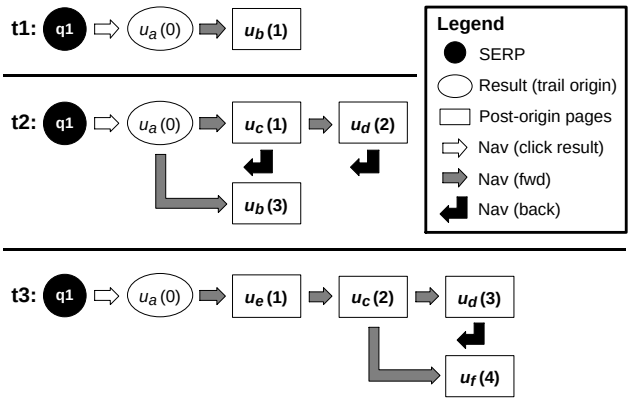

Figure 1. Web behavior graphs illustrating three trails.

Properties of these behavior graphs, among other things, are used to find the best trails. We now describe the trailfinding algorithms.

3.3 Trailfinding Algorithms

The trailfinding task is defined: given a query q and an observed click to trail origin r, find the trail t in T which has the largest $Score(t, q, r)$. The scoring function can be defined in many ways. In this study we experiment with a sample of techniques that include graph properties, relevance, and Web domain information.

Trail Length: The $Score(t, q, r)$ for trail length is defined as the length of t in terms of the total number nodes following r. This algorithm prefers long trails which may be most engaging for some users in terms of browsing activity (or could signify that users are struggling to find useful information). The limitation is that long trails could be obscure, especially if frequency is low. Trails from Figure 1 ordered by length are: $t3$ (four nodes), $t2$ (three nodes), and $t1$ (one node).

Trail Breadth: The $Score(t, q, r)$ for trail breadth is defined as the number of branches in t from the origin r. In Figure 1, $t2$ has the maximum trail breadth of two and would be the best trail in the figure according to this algorithm. Broad trails let users explore various sub-topics while retaining the overall concept, e.g., users might look for specific e-cards within an e-card website.

Trail Depth: The $Score(t, q, r)$ for trail depth is defined as the maximum number of nodes on a single branch from the origin r. Deep trails are usually exploratory and can take users to new concepts or topics. $t3$ is the "deepest" trail in Figure 1 (depth = 3).

Trail Frequency: The $Score(t,q,r)$ is based on the frequency of t for a given query q and origin r. If we assume that in Figure 1, $Freq(t1,q1,u_a)$ = 3; $Freq(t2,q1,u_a)$ = 2 and $Freq(t3,q1,u_a)$ = 1, this algorithm would associate scores of 3, 2 and 1 to $t1$, $t2$ and $t3$ respectively. This algorithm favors short trails.

Trail Strength: Scoring trails based on their strength: (i) the engaging potential of the behavior graph in terms of size, and (ii) the ease of navigation. To estimate the strength of tree starting with query q and origin r, we first compute the total frequency of all navigations of type $< u_x \rightarrow u_y >$ with the user navigating to u_y from u_x in trails starting with query q and origin r. That is:

$$\left(q, r, < u_x \rightarrow u_y >\right) = \sum_{u_x \rightarrow u_y \, in \, t} Freq(t,q,r) \quad (1)$$

where $Freq(t,q,r)$ is the frequency of t for query q and origin r. For $q1$, u_a in Figure 1, post-SERP navigations over all three trails, with frequencies as above are: $< u_a \rightarrow u_b >$: 5; $< u_a \rightarrow u_c >$: 2; $< u_c \rightarrow u_d >$: 3; $< u_a \rightarrow u_e >$: 1; $< u_e \rightarrow u_c >$: 1; $< u_c \rightarrow u_f >$: 1. Given this navigation model, trail strength is defined as:

$$Score(t,q,r) = \sum_{u_x \rightarrow u_y \, in \, t} \left(u_x \rightarrow u_y, q, r\right) \quad (2)$$

This helps find long trails that are easy to navigate. Applying this to trails in Figure 1 results in a trail ranking of $t2$ ($Score$ = 10) followed by $t3$ ($Score$ = 6) and then $t1$ ($Score$ = 5).

Trail Diversity: The $Score(t,q,r)$ is based on the number of pages in t whose Web domain (extracted automatically from the URL string for each page) differs from that of the origin r. In Figure 1, if we assume the domain of URLs u_c, u_d and u_f differs from that of URL u_a, then the trail ordering would be $t3$ (three new domains), $t2$ (two new domains), and $t1$ (zero new domains). Best-trails selected using this algorithm are diverse, offering the user new information relative to the origin page.

Trail Relevance: The $Score(t,q,r)$ for trail relevance for each t page is first calculated using the $max(\%\ query\ terms\ in\ Title,\ \%\ query\ terms\ in\ URL)$ then averaging these scores across all pages in t to obtain a final trail score. If in Figure 1, the URL of u_b contains all query words of $q1$ and title of u_c and u_d contains all query terms of $q1$. The scores assigned to $t1$, $t2$ and $t3$ are 50, 75 and 20 respectively. This algorithm favors trails with query-relevant titles and URLs, suggesting the trail itself is relevant.

3.4 Trailfinding Using Term Matching

The trailfinding algorithms described in this section so far rely on an exact match between the user query and the query starting the trails. The algorithms can be extended to associate trails to *unseen queries* using term matching based on a variant of $tf.idf$. This is important because over half of queries have never been seen by the search engine [27]. Let $\{w_1, w_2, \ldots\}$ be terms in q and for each w_i, get all trails in T occurring from a prior $w_i \times r$. The following equation generates a score for each trail for query q and origin r:

$$Score(t,q,r) = \sum_{w \in q} \frac{(1 + F(w,r)) \times weight(t,w,r)}{Log[1 + D(w)]} \quad (3)$$

where $F(w,r)$ is the frequency with which w appears in a query leading to result click on r, $D(w)$ is the document frequency of w computed as the number of origins to which w is associated in logs, and $weight(t,w,r)$ is based on the trailfinding algorithms above, e.g., breadth algorithm sets $weight(t,w,r)$ as t's breadth.

4. EXPERIMENT

In this section we present the research questions that drive our study, summarize the trail data preparation, present metrics used to compare the algorithms, and describe the experimental variants.

4.1 Research Questions

Our study answers a number of research questions:

- *RQ1*: Of the trails and origins, which source: (i) provides more relevant information? (ii) provides more coverage and diversity of the query topic? (iii) provides more useful information?
- *RQ2*: Among trailfinding algorithms: (i) how does the value of best-trails chosen differ? (ii) what are the effects of query characteristics on best-trail value and selection? (iii) what is the impact of origin relevance on best-trail value and selection?
- *RQ3*: In associating trails to unseen queries: (i) how does the value of trails found through query-term matching compare to trails with exact query matches found in logs? (ii) how robust is term matching for longer queries (which may be noisy)?

4.2 Data Preparation

To help ensure experimental integrity, we did not use all search trails in T. Instead, we filtered the data as discussed below.

4.2.1 Human Judged Query-URL Data

In addition to the trail data, we also obtained human relevance judgments for over eighty thousand queries that were randomly sampled by frequency from the query logs of a large search engine. Trained judges assigned relevance labels on a six-point scale —*Bad*, *Poor*, *Fair*, *Good*, *Excellent*, and *Perfect*—to top-ranked pooled Web search results for each query from the Google, Yahoo!, and Bing search engines during a separate search engine assessment activity. This led to relevance judgments for hundreds of pages for each query. These judgments allowed us to estimate the relevance of information encountered at different parts of the trails. We filtered original trail data so that the origins of the trails (R) have human judgments for at least one query.

4.2.2 ODP Labeling

Two of the four evaluation metrics used in our study—coverage, and diversity—required information about page topicality and query interest. Firstly, we classified trail pages present in T into the topical hierarchy from a popular Web directory, the Open Directory Project (ODP) (dmoz.org). Given the large number of pages involved, we used automatic classification. Our classifier assigned one or more labels to the pages based on the ODP using a similar approach to Shen et al. [19]. Classification begins with URLs present in the ODP and incrementally prunes non-present URLs until a *match* is found or *miss* declared. Similar to [19], we excluded Web pages labeled with the "Regional" and "World" top-level ODP categories, since they are location-based and are typically uninformative for constructing models of user interests. The coverage of our ODP classifier with URL back-off was approximately 65%. A missing or partial labeling of trail was allowed. Next, we constructed a set of *query interest models* for each query having human judged data. These models served as the ground truth for our estimates of coverage and diversity. A query's interest model comprises the ODP category labels assigned to the URLs in the union of the top-200 search results for that query from Google, Yahoo!, and Bing. ODP labels are grouped and their frequency values are normalized such that across all labels they sum to one. For example, the most popular labels in the interest model for the query *[triathlon training]*, and their normalized frequencies (f_l), are shown in Figure 2.

Label	f_l
$Top/Sports/Multi\_Sports/Triathlon/Training$	0.58
$Top/Sports/Multi\_Sports/Triathlon/Events$	0.21
$Top/Shopping/Sports/Triathlon$	0.11

Figure 2. Top ODP categories for *[triathlon training]*.

To improve the reliability of our evaluation metrics, the query interest models had to be based on at least 50 fully-labeled search results (i.e., were not missing a label and did not have a label from an ignored category) and based only on category labels with a frequency of at least three (to reduce label noise).

4.2.3 Data Normalization and Pruning

We applied normalization and pruning to ensure data quality:

- All queries were normalized (involving removing punctuation, lowercasing, etc.) to facilitate comparability among trails and between the trails and other resources.
- Query-origin pairs were required to contain at least five unique trails and at least one trail of length exceeding two to maintain substantial variety for trailfinding.
- Common queries such as *[facebook]*, *[myspace]*, and *[yahoo]* contained thousands of short trails in the data since the ideal result for such queries presents users with a number of ways to branch into social networks or directory structure. To handle this, we first bucketed each query-origin pair in T based on trail length. Then, for all trails of a particular length for each query-origin pair, pruned the trails for which rank based on frequency was greater than 50 and ratio of frequency to maximum frequency for this bucket was less than 25%. This allowed us to maintain high variability in trail data yet remove many spurious trails for some common queries.

4.2.4 Query and Term-based Trail Data

Filtering and pruning reduced T to 209 million trails, roughly 20% of its original size. We created two data sets from T: (i) T_q filtered by queries with query interest models and human judgments, and (ii) T_w created by splitting T into terms and filtering by term-origin pairs in T_q. T_q comprises 20 thousand unique queries, 109 thousand unique origins, 139 thousand query-origin pairs and 20 million unique trails. T_w comprises 15 thousand unique query terms, 109 thousand unique origins, 265 thousand term-origin pairs and 78 million unique trails. This filtering created high-quality data sets for our log-based investigation.

4.3 Metrics

We used four metrics to compare the best-trails selected using our trailfinding algorithms to compare *sources* (origins and trails). The metrics were coverage, diversity, relevance, and utility. These metrics were chosen to capture many important elements of information seeking, as highlighted by relevant research (e.g., [6] [7]). The use of multiple metrics allows us to compare the value of the sources in different ways and also understand how the trailfinding algorithms affect different aspects of information gain.

4.3.1 Coverage

Topic coverage is meant to reflect the value of each source in providing access to the central themes of the query topic. To estimate the coverage of each of source, we created a *source interest model* (i_s) comprising ODP labels for each source assigned as described in Section 4.2.2. We then computed the fraction of the query interest model (i_q) covered by i_s. That is:

$$Coverage(i_s, i_q) = \sum_{l \in (i_s \cap i_q)} f_l \quad (4)$$

where l is an ODP label and f_l represents the normalized frequency weight of that label in the query interest model i_q.

4.3.2 Diversity

Topic diversity estimates the fraction of unique query-relevant concepts surfaced by a given source. Exposure to different perspectives and ideas may help users for complex or exploratory search tasks. To estimate the diversity of the information provided by each source we use an approach similar to our coverage estimation, but we only require the fraction of distinct category labels from i_q that appear in i_s (i.e., frequency is ignored). That is:

$$Diversity(i_s, i_q) = \sum_{l \in (i_s \cap i_q)} \frac{1}{|i_q|} \quad (5)$$

where l is an ODP label and $|i_q|$ is number of distinct i_q labels.

4.3.3 Relevance

The next metric used to compare the trail sources was relevance to the query that initiated the trail. For each trail, we computed the average relevance judgment score with respect to the query. In this analysis, the missing judgments for a page were labeled *Bad* since the judged label data was quite exhaustive for each query and hence missing pages may signify irrelevance to the query.

4.3.4 Utility

We also studied the utility of each source, estimated using dwell time (i.e., the amount of time spent on a particular page by a user). Prior research has demonstrated that during search activity, a dwell time of 30 seconds or more on a Web page can be indicative of page utility [9]. We apply this threshold in our analysis to determine if a source contains at least one page of utility

In all metrics used, a higher value is more positive. The metrics are computed for each source, micro-averaged within each query, and macro-averaged across all queries to obtain a single value for each source-metric pair. This ensures that all queries are treated equally and popular queries do not dominate aggregate metrics. More detail on the metrics is provided by White and Huang [29].

4.4 Methodology

In this section so far we have described the research questions, the trail data preparation procedures, and the metrics used to evaluate the sources. Our methodology comprised the following steps:

1. For each search trail t in T_q, assign ODP labels to all pages in t. Build source interest models for origin page and full trail. Compute metrics using methods described in Section 4.3.
2. For each query-origin pair, select the best trail using each trailfinding algorithm (T_{q_best}). For each trail t in T_{q_best}, compute metrics. Split findings on query length, query type (informational versus navigational), and origin relevance.
3. For each query-origin pair, find the best trail by applying the term-matching approach to T_w, generate a trail set T_{w_best}, and compare trails in T_{w_best} to those in T_{q_best} using our metrics.

5. FINDINGS

We report findings separately for each of our three research questions. We use parametric statistical testing where appropriate. Given the large sample sizes, all observed differences are significant at $p < 0.01$ unless otherwise stated.

Table 1. Comparison of full trails relative to origins. Segments based on trail length to study effect of length on the metrics.
Numbers are averages. Underlined numbers represent statistically-significant differences w.r.t the origin ($p < 0.01$) based on paired t-tests.

Segment		Metrics (Δ% (or just Δ for relevance) are computed relative to origin as baseline)					Trail Statistics		
		Trail Source	Coverage (Δ%)	Diversity (Δ%)	Relevance (Δ)	Utility (Δ%)	#Queries	#Query-Origins	#Trails
All		Origin	8.5	5.7	2.9	43.7	20,521	139,592	20,429,904
		Full Trail	9.7 (+14)	6.6 (+15)	1.0 (-1.9)	82.8 (+89)			
Trail Length	2	Origin	8.5	5.7	2.9	43.9	20,143	134,495	1,687,304
		Full Trail	9.2 (+8)	6.2 (+9)	1.6 (-1.4)	72.4 (+65)			
	3-5	Origin	8.5	5.7	2.9	43.7	20,416	137,172	6,801,382
		Full Trail	9.7 (+14)	6.6 (+15)	1.0 (-2.0)	83.7 (+92)			
	6-10	Origin	8.7	5.8	2.9	43.6	19,615	122,490	11,941,218
		Full Trail	10.3 (+19)	7.1 (+21)	0.5 (-2.5)	92.0 (+111)			

5.1 *RQ1*: Effectiveness of Trails vs. Origins

Table 1 shows summary statistics and reports on the average performance of trails and origins over all trails in T_q. Significance testing involved paired t-tests with Bonferroni corrections.

5.1.1 Different Metrics

Coverage was computed using Equation 4. The average coverage scores of trails and origins are reported in the "All" row of Table 1. Full trails show a 14% increase in topic coverage over origins.

Diversity was computed using Equation 5. The average diversity scores of trails were 15% higher than origins.

Coverage and diversity increases for trails over origins reflect the extra information that users find during post-origin navigation. Although it seems that most of the value comes from origin pages, users can still derive value from trails, including benefits not captured by our metrics (e.g., topic novelty).

Relevance was computed using human relevance judgments on a six-point scale ranging from *Bad* (rating=0) to *Perfect* (rating=5). While the relevance of origins is on average *Good*, the average relevance of trails is *Poor*. We attribute this to mapping missing judgments for deep links in trails to the label *Bad*, perhaps related to dynamism in users' information needs as they search [29].

Utility was estimated using dwell times. Findings show that just under half of origins are useful (43.7%) and over three-quarters of trails have useful pages (82.8%). This shows that the likelihood of finding a useful page via navigation is high, a finding supported by previous work on post-query search behavior [23]. This may also be because origins are search results, typically the starting points for a task, and hence have rapid click-though [17][21].

5.1.2 Effect of Trail Length

To determine the effect of trail length on trail performance, we segmented all search trails into three segments based on length=2, length=3-5, and length=6-10. We did this because: (i) there were insufficient trails for a segment for each length, and (ii) so that we could maintain usable levels of trail variety in each segment. The findings are reported in Table 1 adjacent to "Trail Length." First, even small trails of length 2 added value over origins in terms of coverage, diversity and utility (coverage:+8%, diversity:+9%, utility:+65%). Second, trail length appears to affect trail value. For example, coverage increased from 9.2 to 10.3 (the gain over origin increased from 8% to 19%) in moving between length=2 to length=6-10. This suggests that the longer the trail, the more different topic-related information users are exposed to.

The above findings show that trails can deliver value to users over origins. Even small trails of size 2 can add significant value. Although further study is needed, this analysis suggests that trails may be a useful addition to results on the SERP. Once we know that showing trails may help, the next step is deciding which trails to show. We now report on trailfinding algorithm performance.

5.2 *RQ2*: Trailfinding Algorithms

We compare the best trails from T_{q_best} selected by each of the seven algorithms for each query-origin pair. We used origins-only as a baseline for the algorithms. Results are shown in Table 2. Independent-measures analysis of variance (ANOVA) were used among eight sources (seven best full trails + origin) for each metric to measure statistical significance. Also, we carried out post-hoc Tukey tests to show if best-trails were significantly better than origins. To select the best algorithm(s), each algorithm is first given votes equal to the number of algorithms it performs significantly better than, using post-hoc Tukey tests ($p < 0.01$). Those algorithms with the most votes performed best for each metric.

5.2.1 Different Metrics

Coverage: Frequency-based trails performed worst among seven algorithms with gain of only 11% over origin (9.4 vs. 8.5). This may be because frequent trails are typically short and may cover less of the topic space. Best-trails based on tree-size and tree-strength had average gain of 20% over origins. The trail diversity algorithm performed best with an average gain of 27% (10.7 vs. 8.5), perhaps because different domains discuss different aspects. Even though trails found by the diversity and strength algorithms were shorter than those found by the trail length algorithm, they covered more of the query topic. These and the findings for other metrics show that there are often better criteria than just length.

Diversity: These findings are somewhat similar to the coverage metric. Length-based trails and strength-based trails have an average gain of 22% over the origin. As expected, the diversity algorithm performed best with on average a 30% gain (7.5 vs. 5.7).

Relevance: Trails selected based on relevance scoring have the highest relevance of 1.4 (*Poor-Fair*). Length and depth based trails performed worst, each having average relevance of 0.5 (*Bad-Poor*). In long or deep trails, users may get sidetracked or information needs evolve during searching [2].

Utility: Best-trails based on trail length have highest utility with an average increase of 109% over origins (91.2 vs. 43.7). It seems that the longer the trail, the more likely a user finds a useful page.

Table 2. Average performance of trail selection algorithms for query and term matching approaches. Underlined numbers represent statistically-significant difference relative to origin ($p < 0.01$) based on post-hoc Tukey tests. **Bold** numbers within each segment represent the trailfinding algorithm(s) that is/are significantly better than the other algorithms most frequently ($p < 0.01$ using post-hoc Tukey tests). The "Origin" rows have the average metric scores across all origins. The "All Trails" rows have the average metric scores across all trails.

Segment (#Queries, #Query-Origins)		Algorithm	Metrics							
			Coverage (Δ%)		Diversity (Δ%)		Relevance (Δ)		Utility (Δ%)	
			Full query	Term match	Full query	Term match	Full query	Term match	Full query	Term match
All (20521, 139592)		Origin	8.5		5.7		2.9		43.7	
		All Trails	9.7 (+14)		6.6 (+15)		1.0 (-1.9)		82.8 (+89)	
		Length	10.2 (+20)	10.2 (+21)	7.0 (+22)	7.1 (+23)	0.5 (-2.4)	0.4 (-2.5)	**91.2** (+109)	**91.6** (+110)
		Diversity	**10.7** (+27)	**11.1** (+30)	**7.5** (+30)	**7.7** (+35)	0.9 (-2.1)	0.7 (-2.2)	85.4 (+95)	88.6 (+103)
		Breadth	9.9 (+17)	10.1 (+19)	6.8 (+19)	6.9 (+21)	0.7 (-2.2)	0.6 (-2.3)	87.6 (+100)	89.3 (+104)
		Depth	10.1 (+18)	10.1 (+18)	6.9 (+21)	6.9 (+20)	0.5 (-2.4)	0.4 (-2.5)	89.9 (+106)	90.3 (+107)
		Relevance	9.5 (+12)	9.4 (+11)	6.4 (+13)	6.4 (+11)	**1.4** (-1.5)	**1.5** (-1.4)	76.4 (+75)	74.5 (+70)
		Frequency	9.4 (+11)	9.3 (+9)	6.4 (+12)	6.3 (+10)	1.3 (-1.6)	1.5 (-1.4)	77.2 (+77)	73.8 (+69)
		Strength	10.2 (+20)	10.2 (+20)	7.0 (+22)	7.0 (+22)	0.6 (-2.3)	0.6 (-2.4)	90.1 (+106)	90.6 (+107)
Origin Relevance	Best Origin (13614, 25890)	Origin	10.0		6.5		4.3		43.0	
		All Trails	11.1 (+11)		7.3 (+12)		1.3 (-2.9)		83.9 (+95)	
		Length	11.4 (+14)	11.5 (+15)	7.7 (+17)	7.7 (+18)	0.6 (-3.6)	0.6 (-3.7)	**90.6** (+111)	**90.9** (+111)
		Diversity	**12.1** (+21)	**12.3** (+23)	**8.2** (+26)	**8.4** (+29)	1.0 (-3.2)	0.8 (-3.4)	86.7 (+102)	89.3 (+108)
		Breadth	11.4 (+14)	11.5 (+15)	7.6 (+17)	7.7 (+18)	0.9 (-3.3)	0.8 (-3.5)	88.8 (+107)	90.2 (+110)
		Depth	11.3 (+13)	11.4 (+14)	7.6 (+16)	7.6 (+16)	0.7 (-3.6)	0.6 (-3.7)	89.0 (+107)	89.2 (+107)
		Relevance	10.8 (+8)	10.8 (+8)	7.1 (+9)	7.1 (+9)	**2.1** (-2.2)	**2.2** (-2.0)	75.0 (+74)	73.8 (+72)
		Frequency	10.8 (+8)	10.7 (+7)	7.1 (+9)	7.1 (+8)	2.1 (-2.2)	**2.2** (-2.0)	74.9 (+74)	72.6 (+69)
		Strength	11.5 (+15)	11.5 (+15)	7.7 (+18)	7.7 (+18)	0.8 (-3.5)	0.8 (-3.5)	90.1 (+110)	**90.6** (+111)
	Worst Origin (6324, 11754)	Origin	7.5		5.1		0.0		39.7	
		All Trails	9.0 (+20)		6.1 (+20)		0.1 (+0.1)		80.2 (+102)	
		Length	9.7 (+29)	9.7 (+29)	6.6 (+29)	6.6 (+30)	0.1 (+0.1)	0.1 (+0.1)	**90.4** (+128)	**90.6** (+128)
		Diversity	**10.2** (+36)	**10.6** (+41)	**7.0** (+37)	**7.3** (+43)	0.1 (+0.1)	0.1 (+0.1)	84.0 (+112)	88.3 (+123)
		Breadth	9.3 (+24)	9.6 (+27)	6.3 (+24)	6.6 (+29)	0.1 (+0.1)	0.1 (+0.1)	85.3 (+115)	87.6 (+121)
		Depth	9.5 (+27)	9.6 (+28)	6.5 (+27)	6.5 (+28)	0.1 (+0.1)	0.1 (+0.1)	**89.1** (+125)	89.2 (+125)
		Relevance	9.0 (+20)	8.9 (+18)	6.1 (+19)	6.0 (+17)	**0.2** (+0.2)	**0.2** (+0.2)	73.8 (+86)	71.5 (+80)
		Frequency	8.9 (+18)	8.6 (+15)	6.0 (+18)	5.8 (+14)	0.1 (+0.1)	0.1 (+0.1)	75.3 (+90)	70.2 (+77)
		Strength	9.7 (+29)	9.7 (+29)	6.6 (+29)	6.7 (+30)	0.1 (+0.1)	0.1 (+0.1)	**88.8** (+124)	89.2 (+125)
Query Length	Words =1 (4514, 38830)	Origin	8.4		5.9		2.7		42.7	
		All Trails	9.6 (+14)		6.8 (+15)		0.9 (-1.8)		82.8 (+94)	
		Length	10.0 (+19)	10.1 (+20)	7.1 (+21)	7.2 (+22)	0.4 (-2.3)	0.4 (-2.3)	**90.7** (+112)	**91.4** (+114)
		Diversity	**10.8** (+28)	**10.9** (+30)	**7.8** (+32)	**7.9** (+33)	0.8 (-2.0)	0.7 (-2.1)	86.2 (+102)	88.6 (+107)
		Breadth	9.8 (+16)	9.9 (+18)	6.9 (+18)	7.0 (+19)	0.7 (-2.1)	0.6 (-2.1)	87.4 (+105)	88.9 (+108)
		Depth	9.9 (+18)	9.9 (+18)	7.1 (+19)	7.1 (+20)	0.5 (-2.3)	0.4 (-2.3)	**89.5** (+109)	90.0 (+110)
		Relevance	9.5 (+13)	9.5 (+13)	6.7 (+13)	6.6 (+13)	**1.4** (-1.3)	**1.4** (-1.3)	74.6 (+74)	74.5 (+74)
		Frequency	9.3 (+11)	9.2 (+10)	6.6 (+11)	6.5 (+10)	1.3 (-1.5)	1.4 (-1.4)	76.7 (+79)	75.4 (+76)
		Strength	10.1 (+20)	10.1 (+20)	7.2 (+21)	7.2 (+22)	0.5 (-2.2)	0.5 (-2.2)	**89.5** (+109)	90.2 (+111)
	Words > 3 (3662, 14320)	Origin	8.4		5.6		3.1		46.1	
		All Trails	9.5 (+14)		6.4 (+16)		1.1 (-2.0)		84.2 (+83)	
		Length	10.0 (+20)	10.1 (+21)	6.9 (+23)	6.9 (+25)	0.5 (-2.6)	0.4 (-2.7)	**92.2** (+100)	**92.9** (+101)
		Diversity	**10.5** (+26)	**11.0** (+31)	**7.2** (+30)	**7.7** (+38)	0.9 (-2.2)	0.7 (-2.4)	86.2 (+87)	90.0 (+95)
		Breadth	9.8 (+17)	10.0 (+20)	6.6 (+20)	6.9 (+23)	0.8 (-2.3)	0.6 (-2.5)	88.4 (+92)	90.4 (+96)
		Depth	9.9 (+18)	9.9 (+19)	6.7 (+21)	6.8 (+22)	0.6 (-2.5)	0.4 (-2.6)	**91.1** (+97)	91.4 (+98)
		Relevance	9.3 (+11)	9.1 (+9)	6.2 (+12)	6.1 (+10)	**1.4** (-1.7)	**1.6** (-1.5)	79.8 (+73)	76.0 (+65)
		Frequency	9.3 (+11)	9.1 (+9)	6.2 (+12)	6.1 (+9)	1.4 (-1.7)	**1.6** (-1.5)	78.9 (+71)	74.2 (+61)
		Strength	10.0 (+20)	10.1 (+21)	6.8 (+23)	6.9 (+24)	0.6 (-2.5)	0.6 (-2.5)	**91.4** (+98)	91.9 (+99)

5.2.2 Breakdown Based on Origin Relevance

We also studied the effect of origin relevance on algorithm performance to determine whether best-trails add value to all results or only those with high or low origin quality. We divided the data into two buckets: one with origins having the highest human-judged label for the query (note that this need not be *Excellent*) and another with origins judged *Poor* or *Bad*.

Best Origins: The coverage of origins increased from 8.5 to 10.0. More relevant origins appear to cover more of the topic space. Also, the coverage values of all best-trail algorithms have increased; for example, trails selected based on diversity increase coverage from 10.7 to 12.1 However, the percentage gain of full trails over origins has decreased to a maximum value of +21% for diverse trails (12.1 vs. 10.0) which is lower than the 27% coverage gain for all origins. This can be explained by the fact that when origins are high quality, the value added by trails drops. Similar results are observed for diversity. Second, trails found using relevance-based scoring performed fairly well: average relevance of *Fair* as compared to 1.4 (*Poor-Fair*) for all origins. This suggests that relevant origins may also link to relevant pages.

Worst Origins: While the absolute coverage values of origin and full trails decreased, the percentage gain from trails increased across all trail selection algorithms. Diverse trails again performed best with an average increase of 36% compared to origin (10.2 vs. 7.5). This almost doubles the 21% increase we observed for best origins. Similar trends can be seen for diversity. Second, there was a decrease in utility for origins whereas some trail selection algorithms showed an increase.

5.2.3 Breakdown Based on Query

We studied the effect of query length and query intent on trail performance to determine whether trails were equally useful for all queries. We segmented query length in three ways: length=1, length=2-3, and length > 3 (long queries). For query intent, we segmented the queries into navigational and informational intent based on click frequencies in search engine logs separate from those used in this study. Per our definition, navigational queries led to a click on the same search result 95% of the time; informational queries led to on average two or more different result clicks per query. The results from query intent were somewhat aligned with that of breakdown based on origin quality. Clear intent queries had trends similar to experiments with best origins and informational queries had trends similar to those of worst origins. Due to space constraints, we only discuss results on query length since those are also important for *RQ3*. The experiments on query length showed no major difference among trailfinding algorithms. We observe similar behavior in terms of relative differences of full trails versus origins. On coverage and diversity metrics, the relevance-based trailfinding algorithm failed to obtain significant differences relative to the origin on long queries. Recall that the relevance-based scoring finds trails based on the match between the query terms and trail titles/URLs. For longer queries, there may be more noise in the queries and the trails found may not cover as much of the query topic space. Another interesting finding was in the absolute values of utility of trails and origins. On long queries, utility increased, suggesting users spent more time on Web pages following those queries.

Interestingly, across all trails and the various segmentations there is at least one trailfinding algorithm (and often many) that outperforms the average over all trails followed by users (shown in the "All Trails" rows). This suggests that trailfinding algorithms may be helpful to users, at least in cases where the benefit brought by the algorithm (e.g., a boost in diversity) matches the user intent.

5.3 *RQ3*: Trailfinding Using Term Matching

Next we report the quality of trails found using the term-matching based approach described in Section 3.4. We use this approach to find trails from T_w for all query-origin pairs for which we have best-trails selected from T_{q_best}. This leads to a new set called T_{w_best}. Note that: (i) for comparability the same queries appeared in both sets, and (ii) creating T_{w_best} could result in associating new trails to query-origin pairs which were not logged.

The average performance numbers for the trails in T_{w_best} are reported in Table 2 alongside those obtained from best-trails of T_q. First, the results from all best trail selections from term-based approaches have similar trends as that of best trails selections from the query based approach. This strongly suggests that our trail selection criteria can be effectively applied to unseen query-origin pairs. Second, the segment based on query length suggests robustness of this technique for longer queries, which posed a challenge because of possible noise. Thirdly, term-based trails have occasionally higher coverage and diversity. For example: for diversity-based best-trails starting with long queries, we have a coverage of 11.0 (i.e., 31% gain over origin) for *term*-based and 10.5 (i.e., 26% gain over origin) for *query*-based trails. Fourth, relevance dropped from 0.9 to 0.7. This suggests that despite the coverage and diversity gains, term-based trails are slightly less relevant than query-based trails, perhaps because the term-based technique finds trails that may only be partially query relevant.

6. DISCUSSION AND IMPLICATIONS

We have described a log-based study of various trailfinding algorithms to support post-query search interaction. Trails are selected from the search and browsing logs of many users. Our findings show that users' trails bring them value, best-trails can be chosen that outperform users' own trails, different trailfinding algorithms perform well under different metrics, and a term-matching variant lets algorithms effectively handle unseen queries.

Our first research question compared the value of trails with origin pages. The findings showed a significant increase in value for trails over origins across almost all metrics except relevance when we normalized for trail length. As more information is viewed by users, there is more opportunity for them to gain. Relevance degraded because un-judged pages were labeled *Bad*. If we ignore un-judged pages, trails have the same relevance as origins.

Since search trails appeared to demonstrate value over origins, the next research question addressed the issue of whether we could find the trail from the available options that maximized coverage, diversity, relevance, and/or utility. Although there was no clear winner, the findings were roughly in line with our intuitions. The diversity algorithm that preferred trails with multiple domains performed best in terms of coverage and diversity and the relevance-based algorithm preferring trails with a high query-to-title/URL match performed best in terms of relevance. Trail length algorithms had the best utility, perhaps because the longer the trail, the more likely that users would encounter a useful page. On average, trailfinding outperforms the trails that users follow themselves. This suggests that there is typically a trail with higher return than that followed by a user and, may improve a user's search effectiveness if shown. It also allows us to exclude underperforming algorithms from further study (e.g., frequency, strength).

As part of this analysis we studied the effects of origin quality, query type, and query length on trailfinding algorithm performance. The findings showed differences in the effectiveness of the algorithms depending on origin quality and query characteristics. Trails may not be appropriate for all search results and more work is needed to determine which results or queries deserve trails, to investigate trailfinding algorithms, and to explore ways to effectively select between these algorithms given different user needs. For example, if the user cares about topic coverage, then we should select trails based on the trail diversity algorithm.

The final research question addresses whether our trailfinding approach could be adapted to handle unseen queries. Findings showed that performance was roughly equivalent between the best trails selected from the logs and those generated based on our algorithm. The term-based approach saw an increase in the coverage and diversity and a decrease in relevance. This could be part of a backoff strategy where we search within trails in logs and use those chosen through term matching if no trails are found.

One limitation of this research is the assumption that there is a best trail for each query-result pair. It is conceivable that there will be multiple equivalent or complementary trails for any pairing. Ways to tiebreak between trails (e.g., showing trails that the user has not yet traveled) need to be explored. More work is needed to validate metrics used, in particular measures of coverage, diversity, and utility currently inferred from interactions (e.g., [29]).

The next step in our research is to show trails on SERPs. Trails can be presented as an alternative to result lists, as instant answers above result lists, in pop-ups shown after hovering over a result, below each result in addition to the snippet and URL, or even on the click trail the user is following. Although we are limited by what can be inferred from log data, our approach has provided insight on what algorithms perform best and when. Follow-up user studies and large-scale flights are planned to compare trail presentation methods and further analyze trailfinding techniques.

7. CONCLUSIONS AND FUTURE WORK

In this paper we have presented a study of trailfinding techniques to support Web search. We employed a log-based methodology to afford us control over experimental variables and rapidly test multiple trailfinding algorithms. We showed that trails provided additional value over trail origins, especially for longer trails that may contain more varied information. We experimented with different trailfinding algorithms and showed that they can outperform trails followed by most users; their performance was affected by the relevance of the origins and query characteristics, meaning that trails may need to be tailored to query and result properties. We also tested a term matching variant that alleviated the need for an exact term match between queries and trails, which led to coverage and diversity gains at the cost of a slight decrease in relevance. In future work we will integrate best-trails into search engine result pages and conduct user studies on their effectiveness.

REFERENCES

[1] Agichtein, E., Brill, E. & Dumais, S. (2006). Improving web search ranking by incorporating user behavior information. *Proc. SIGIR*, 19-26.

[2] Bates, M.J. (1989). The design of browsing and berrypicking techniques for the online search interface. *Online Review*, 13(5): 407-424.

[3] Bilenko, M. & White, R.W. (2008). Mining the search trails of surfing crowds: identifying relevant websites from user activity. *Proc. WWW*, 51-60.

[4] Bush, V. (1945). As we may think. *Atl. Monthly*, 3(2): 37-46.

[5] Chalmers, M., Rodden, K. & Brodbeck, D. (1998). The order of things: activity-centered information access. *Proc. WWW*.

[6] Clarke, C.L.A. et al. (2008). Novelty and diversity in information retrieval evaluation. *Proc. SIGIR*, 659-666.

[7] Cole, M. et al. (2009). Usefulness as the criterion for evaluation of interactive information retrieval. *Proc. HCIR*, 1-4.

[8] Downey, D., Dumais, S. & Horvitz, E. (2007). Models of searching and browsing: languages, studies, and application. *Proc. IJCAI*, 2740-2747.

[9] Fox, S. et al. (2005). Evaluating implicit measures to improve the search experience. *TOIS*, 23(2): 147-168.

[10] Freyne, J. et al. (2007). Collecting community wisdom: integrating social search and social navigation. *IUI*, 52-61.

[11] Guinan, C. & Smeaton, A.F. (1993). Information retrieval from hypertext using dynamically planned guided tours. *Proc. ECHT*, 122-130.

[12] Hammond, N. & Allison, L. (1988). Travels around a learning support environment: rambling, orienteering, or touring? *Proc. SIGCHI*, 269-273.

[13] Joachims, T. (2002). Optimizing search engines using clickthrough data. *Proc. SIGKDD*, 133-142.

[14] Joachims, T., Freitag, D. & Mitchell, T. (1997). WebWatcher: a tour guide for the world wide web. *Proc. IJCAI*, 770-775.

[15] Olston, C. & Chi, E.H. (2003). ScentTrails: integrating browsing and searching on the web. *TOCHI*, 10(3): 1-21.

[16] O'Day, V. & Jeffries, R. (1993). Orienteering in an information landscape: how information seekers get from here to there. *Proc. INTERCHI*, 438-445.

[17] Pandit, S. & Olston, C. (2007). Navigation-aided retrieval. *Proc. WWW*, 391-400.

[18] Pirolli, P. & Card, S.K. (1999). Information foraging. *Psychological Review*, 106(4): 643-675.

[19] Shen, X., Dumais, S. & Horvitz, E. (2005). Analysis of topic dynamics in web search. *Proc. WWW*, 1102-1103.

[20] Singhal, A. (2001). Modern information retrieval: a brief overview. *Bulletin of the IEEE Computer Society Technical Committee on Data Engineering*, 24(4): 35-43.

[21] Teevan, J. et al. (2004). The perfect search engine is not enough: a study of orienteering behavior in directed search. *Proc. SIGCHI*, 415-422.

[22] Trigg, R.H. (1988). Guided tours and tabletops: tools for communicating in a hypertext environment. *TOIS*, 6(4).

[23] Vakkari, P. & Taneli, M. (2009). Comparing google to ask-a-librarian service for answering factual and topic questions. *Proc. EDCL*, 352-363.

[24] Wang, X. & Zhai, C. (2009). Beyond hyperlinks: organizing information footprints in search logs to support effective browsing. *Proc. CIKM*, 1237-1246.

[25] Wexelblat, A. & Maes, P. (1999). Footprints: history-rich tools for information foraging. *Proc. SIGCHI*, 270-277.

[26] Wheeldon, R. & Levene, M. (2003). The best trail algorithm for assisted navigation of web sites. *Proc. LA-WEB*, 166.

[27] White, R.W., Bilenko, M., & Cucerzan, S. (2007). Studying the use of popular destinations to enhance web search interaction. *Proc. SIGIR*, 159-166.

[28] White, R.W. & Drucker, S.M. (2007). Investigating behavioral variability in web search. *Proc. WWW*, 21-30.

[29] White, R.W. & Huang, J. (2010). Assessing the scenic route: measuring the value of search trails in web logs. *Proc. SIGIR*

[30] Zellweger, P.T. (1989). Scripted documents: a hypermedia path mechanism. *Proc. Hypertext*, 1-14.