

# A General Evaluation Framework for Topical Crawlers

Padmini Srinivasan\*

School of Library & Information Science

The University of Iowa

Iowa City, IA 52245

`padmini-srinivasan@uiowa.edu`

Filippo Menczer<sup>†</sup> and Gautam Pant

Department of Management Sciences

The University of Iowa

Iowa City, IA 52245

`{gautam-pant,filippo-menczer}@uiowa.edu`

## Abstract

Topical crawlers are becoming important tools to support applications such as specialized Web portals, online searching, and competitive intelligence. As the Web mining field matures, the disparate crawling strategies proposed in the literature will have to be evaluated and compared on common tasks through well-defined performance measures. This paper presents a general framework to evaluate topical crawlers. We identify a class of tasks that model crawling applications of different nature and difficulty. We then introduce a set of performance measures for fair comparative evaluations of crawlers along several dimensions including generalized notions of precision, recall, and efficiency that are appropriate and practical for the Web. The framework relies on independent relevance judgements compiled by human editors and available from public directories. Two sources of evidence are proposed to assess crawled pages, capturing different relevance criteria. Finally we introduce a set of topic characterizations to analyze the variability in crawling effectiveness across topics. The proposed evaluation framework synthesizes a number of methodologies in the topical crawlers literature and many lessons learned from several studies conducted by our group. The general framework is described in detail and then illustrated in practice by a case study that evaluates four public crawling algorithms.

---

\*Current address: National Library of Medicine, Bethesda, MD 20894

<sup>†</sup>Corresponding author. Tel: +1-319-335-0884, Fax: +1-319-335-0297

# 1 Introduction

Topical crawlers, also known as topic driven or focused crawlers, are an important class of crawler programs that complement search engines. Search engines serve the general population of Web users. In contrast, topical crawlers are activated in response to particular information needs. These could be from an individual user (query time or online crawlers) or from a community with shared interests (topical search engines and portals). The crawlers underlying search engines are designed to fetch as comprehensive a snapshot of the Web as is possible; topical crawlers are designed to target portions of the Web that are relevant to the triggering topic. Such crawlers have the advantage that they may in fact be driven by a rich context (topics, queries, user profiles) within which to interpret pages and select the links to visit. Today, topical crawlers have become the basis for many specialized services such as investment portals, competitive intelligence tools, and scientific paper repositories.

Starting with the early breadth first [33] and depth first crawlers [14] defining the beginnings of research on crawlers, we now see a variety of crawler algorithms. There is Shark Search [18], a more aggressive variant of De Bra's Fish Search [14]. There are crawlers whose decisions rely heavily on link based criteria [12, 15, 6]. Others exploit lexical and conceptual knowledge such as those provided by a topic hierarchy [11]. Still others emphasize contextual knowledge [1, 29, 25] for the topic including those received via relevance feedback. In a companion paper we study several machine learning issues related to crawler algorithms, including for example, the role of adaptation in crawling and the scaling of algorithms [27].

One research area that is gathering increasing momentum is the evaluation of topical crawlers. The rich legacy of information retrieval research comparing retrieval algorithms in the non-Web context offers many evaluation methods and measures that may be applied toward this end. However, given that the dimensions of the crawler evaluation problem are dramatically different, the design of appropriate evaluation strategies is a valid challenge.

In a general sense, a crawler may be evaluated on its ability to retrieve "good" pages. However, a major hurdle is the problem of recognizing these good pages. In an operational

environment real users may judge the relevance of pages as these are crawled allowing us to determine if the crawl was successful or not. Unfortunately, meaningful experiments involving real users for assessing Web crawls are extremely problematic. For instance the very scale of the Web suggests that in order to obtain a reasonable notion of crawl effectiveness one must conduct a large number of crawls, i.e., involve a large number of users.

Crawls against the live Web also pose serious time constraints. Therefore crawls other than short-lived ones will seem overly burdensome to the user. We may choose to avoid these time loads by showing the user the results of the full crawl — but this again limits the extent of the crawl. Next we may choose indirect methods such as inferring crawler strengths by assessing the applications that they support. However this assumes that the underlying crawlers are openly specified, and also prohibits the assessment of crawlers that are new.

Thus we argue that although obtaining user based evaluation results remains the ideal, at this juncture it is appropriate and important to seek user independent mechanisms to assess crawl performance. Moreover, in the not so distant future, the majority of the direct consumers of information is more likely to be Web agents working on behalf of humans and other Web agents than humans themselves. Thus it is quite reasonable to explore crawlers in a context where the parameters of crawl time and crawl distance may be beyond the limits of human acceptance imposed by user based experimentation.

Our analysis of the crawler literature [1, 2, 4, 6, 8, 11, 10, 17, 18, 29, 37] and our own experience [21, 24, 25, 26, 22, 31, 32, 27] indicate that in general, when embarking upon an experiment comparing crawling algorithms, several critical decisions are made. These impact not only the immediate outcome and value of the study but also the ability to make comparisons with future crawler evaluations. In this paper we offer a general framework for crawler evaluation research that is founded upon these decisions. Our goal is both to present this framework and demonstrate its application to the evaluation of four off-the-shelf crawlers. Our generic framework has three distinct dimensions. The first dimension is regarding the nature of the crawl task addressed (Section 2). This includes consideration of how topics are

defined and how seeds and target relevant pages are identified. The second dimension deals with evaluation metrics both for effectiveness and for efficiency analysis (Section 3). The last dimension of the framework looks at topics in greater detail, by examining particular characteristics such as popularity and authoritativeness and their effect on crawler behavior (Section 4). We present this framework as a means for systematically increasing our understanding of crawler technologies through experimentation. After these sections, we take four off-the-shelf crawlers and compare them using this framework (Section 5). We conclude in Section 6 with a discussion on the experiment in our case study and on the evaluation framework in general.

## 2 Nature of Crawl Tasks

A crawl task is characterized by several features. These include how the topic is defined, the mechanism by which seed pages for starting the crawl are selected and the location of the topic’s relevant target pages relative to the seed pages. Obviously, a crawl task where the seeds are themselves relevant to the topic is likely to be less challenging than one in which the seeds and targets are separated by some non trivial link distance. These issues are discussed in this section.

### 2.1 Topics and Descriptions

Unlike questions that are built around inquiries of some kind, a topic such as ‘Sports’ or ‘US Opens’ or ‘anthrax’ delineates a particular domain of discourse. As seen for example in [1, 5, 6, 18, 11, 9], topics offer a handy mechanism for evaluating crawlers, since we may examine their ability to retrieve pages that are on topic. Topics may be obtained from different sources as for instance asking users to specify them. One approach is to derive topics from a hierarchical index of concepts such as Yahoo or the Open Directory [11, 26, 32]. A key point to note is that all topics are not equal. Topics such as ‘2002 US Opens’ and ‘trade

embargo’ are much more specific than ‘Sports’ and ‘Business’ respectively. Moreover, a given topic may be defined in several different ways, as we describe below.

Topic specification has a very critical role in our framework. We start by asking: given a hierarchy of concepts how are topics to be specified? One method is to use the leaf node concepts as topics [26]. The problem with this approach is that the selected topics may be at different levels of specificity. In our framework we control for this by deriving topics from concept nodes that are at a predefined distance (`TOPIC_LEVEL`) from the root of the concept hierarchy, i.e., they are at about the same level of specificity. Once a topic node is identified, the topic *keywords* are formed by concatenating the node labels from the root of the directory tree to the topic node.

Instead of building topics from single nodes we take a more general approach and build them from subtrees of a given maximum depth (`MAX_DEPTH`) whose roots are `TOPIC_LEVEL` links away from the root of the original concept tree. Depth, as used here, refers to the height of a subtree. Figure 1 illustrates these ideas with topic subtrees of `MAX_DEPTH = 2` built from a concept hierarchy at `TOPIC_LEVEL = 2`.

In our framework, subtrees offer a systematic way to delineate topics. Moreover, by varying a parameter `DEPTH` from 0 to `MAX_DEPTH`, it is possible to generate alternative *descriptions* of a given topic. If we use information from the root of the topic subtree alone (`DEPTH = 0`), then we get the most minimal set of topic descriptions. If in addition, we use information from the next level of nodes in the subtree (`DEPTH = 1`), then we get a more detailed view of the topic and so on till the leaf nodes of the subtree (`DEPTH = MAX_DEPTH`) are involved. Both the descriptive text that embeds the external links and the anchor text that labels the external links in the page at the root of the topic subtree may be used to provide the minimal description of the topic. Note that these textual descriptions of external pages are written by expert human editors, independent of the authors who generate the content of the pages described. Similar text extracted from higher depth nodes may be added to provide an augmented description of the topic and so on. Thus a single topic may have

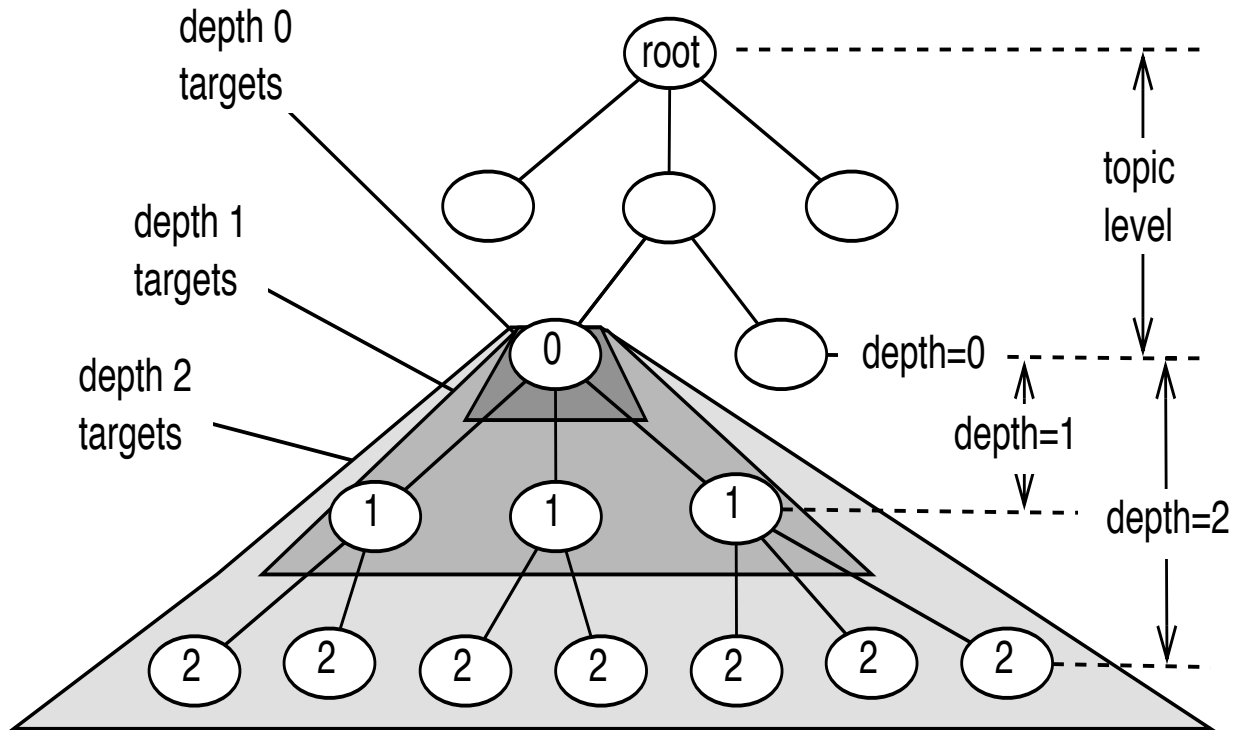


Figure 1: Illustration of a topic subtree from a hierarchical directory. The topic in this example has `TOPIC_LEVEL=2` and `MAX_DEPTH=2`. Topic nodes are labeled with their depth. The external pages linked from nodes at a given depth are the targets for that depth. Shaded areas represent target sets corresponding to subtrees of depth between 0 and `MAX_DEPTH`, i.e., to progressively broader interpretations of the topic. A broader interpretation (lighter shade of gray) includes additional, more specific targets.

$\text{MAX\_DEPTH} + 1$  sets of descriptions that differ in their level of detail. Descriptions at higher depths include those at lower depths. Figure 2 illustrates the concept of topic description with an example corresponding to a leaf topic, i.e.,  $\text{DEPTH} = \text{MAX\_DEPTH} = 0$ .

## 2.2 Target Pages

Since it is difficult to get users to judge retrieved Web pages for relevance, it is more typical to apply indirect methods to identify target pages. Hierarchical concept based indices offer some options in this regard. Such directories are designed to assist the user by offering entry points to a set of conceptually organized Web pages. Thus the Yahoo directory page on *Newspapers* leads to *USA Today*, *New York Times* and the Web sites of other news media. In effect, one may regard the resources pointed to by the external links as the relevant set for the concept represented by the directory page: *USA Today* and *New York Times* may be viewed as part of the set of target relevant pages for the concept of Newspapers.

In our framework, parallel to topic descriptions, topic target pages are also differentiated by the  $\text{DEPTH}$  of the topic subtree. Thus when the topic is described by a subtree of  $\text{DEPTH} = 0$ , then the relevant target set consists of the external links from the root node of the topic subtree. Such an example is depicted in Figure 2. The target set corresponding to the topic description at  $\text{DEPTH} = 1$  also includes the external links from the topic's nodes at this level and so on. Thus for a single topic there are  $\text{MAX\_DEPTH} + 1$  sets of target pages defined, with the set at a higher depth including the sets at the lower depths.

## 2.3 Seed Pages

The specification of seed pages is one of the most crucial aspects defining the crawl task. The approach used in several papers [11, 4, 26, 22, 31] is to start the crawlers from pages that are assumed to be relevant. In other words some of the target pages are selected to form the seeds. This type of crawl task mimics the query by example search mode where the user provides a sample relevant page as a starting point for the crawl. As an alternate strategy

dmz open directory project

Topic: **Home: [Cooking: Baking and Confections: Cookies: Chocolate Chip](#) (6)**


Description: **The Big Chocolate Chip Cookie Page** - Devoted to the chocolate chip cookie.  
**Chocolate Chip Cookies** - Various recipes for cookies with morsels of chocolate.  
**Chocolate Chip Cookies from Allrecipes** - Include regular, nuts, white chocolate.  
**In the Chips** - Cookies, cakes, candy, muffins, etc. using chocolate chips.

Copyright © 1998-2000 Netscape Terms of Use

---

Recipe and Tips R	Recipe	Rating
A chocolate chip cookie recipe that uses Karo syrup in it. A recipe using metric measurements? Any recipes that don't use eggs? A chocolate chip cookie recipe that you bake in mini muffin pans w How does one avoid dry, 'cakey' cookies? Any recipes for Chocolate Chip Coolie Pies? The recipe for chocolate chip cookies in a jar. All the dry ingredien	<b>Absolutely Excellent Oatmeal Cookies</b> Submitted by: <b>Marylou</b>  These are chewy, healthy oatmeal cookies which can be prepared in a number of variations, just add nuts, raisins, chocolate chips, coconut, candied fruit or any other additions.  <b>Absolutely Sinful Chocolate Chocolate Chip Cookies</b> Submitted by: <b>Marsha</b>	★★★★* 46 Ratings 25 Reviews  ★★★★* 63 Ratings 49 Reviews

***Cookies that are out of this world...***



In the kitchen of a Whitman Massachusetts country inn, the first chocolate chip cookie emerged in 1937. Simple experiments led to a recipe combining bits of chocolate candy with a kind of butter cookie cookie dough resulting in a delicious mixture that offered the crunchiness of a cookie with a taste of chocolate candy in every bite. Obviously the cookies were a hit at the inn and wherever else the recipe spread. Chocolate chip cookies have remained an American homemade treat.

**CHOCOLATE CHIP COOKIES**

**RECIPE INDEX**

[BLACK AND WHITE CHOCOLATE CHIPPERS](#)  
[CLASSIC CHOCOLATE CHIP COOKIES](#)  
[COW CHIP COOKIES](#)  
[DEVIL'S FOOD CHOCOLATE CHIP COOKIES](#)  
[GOTTA HAVE EM' NOW! COOKIES](#)  
[MINT CHOCOLATE SANDWICH COOKIES](#)  
[NEIMAN MARCUS CHOCOLATE CHIP COOKIES](#)  
[OLD FASHIONED CHOCOLATE CHIPPERS](#)

Figure 2: Illustration of a topic node from from the Open Directory (dmoz.org), with its associated topic keywords, description, and target set. In this abridged example the topic has TOPIC\_LEVEL=5. Since this is a leaf node (no subtopics), the only possible target set corresponds to DEPTH=0.

these relevant seeds may also be obtained from a search engine [31, 39]. The idea is to see if the crawlers are able to find other target pages for the topic. An assumption implicit in this crawl task is that pages that are relevant tend to be neighbors of each other [21, 23]. Thus the objective of the crawler is to stay focused, i.e., remain within the neighborhood in which relevant documents have been found.

A harder crawl problem is when the seeds are different from the target pages. In this case there is less prior information available about the target pages when the crawl begins. Links become important not only in terms of the particular pages pointed to, but also in terms of the likelihood of reaching relevant documents further down the path [15]. This problem is also very realistic since quite commonly users are unable to specify a known relevant page and also the search engines may not return relevant pages. With a few exceptions, this second task is rarely considered in the literature. The effort in [1] is somewhat related to this task in that the authors start the crawl from general points such as Amazon.com. Although Cho *et al.* [12] start their crawls at a general point, i.e., the Stanford Web site, topics have rather primitive roles in their research.

Our framework takes a general approach and provides a mechanism to control the level of difficulty of the crawl task. One may specify a distance  $\text{DIST} = 0, 1, 2, \dots$  links between seeds and targets. Thus when  $\text{DIST} = 0$ , we have the simple crawl task where some targets form seeds. As  $\text{DIST}$  increases, so does the challenge faced by the crawlers. The following procedure implements the selection of up to  $\text{N\_SEEDS}$  seed pages for a given topic:

```
select_seeds (DIST, N_SEEDS, N_TOPICS, N_QUERIES) {
  n_sample = MAX_QUERIES / (N_TOPICS * DIST);
  seed_set = targets(DEPTH = 0);
  repeat DIST times {
    sample = random_subset(seed_set, n_sample);
    seed_set = get_backlinks(sample);
  }
  return random_subset(seed_set, N_SEEDS);
}
```

We start from the  $\text{DEPTH} = 0$  target pages and select a set of seed pages such that there

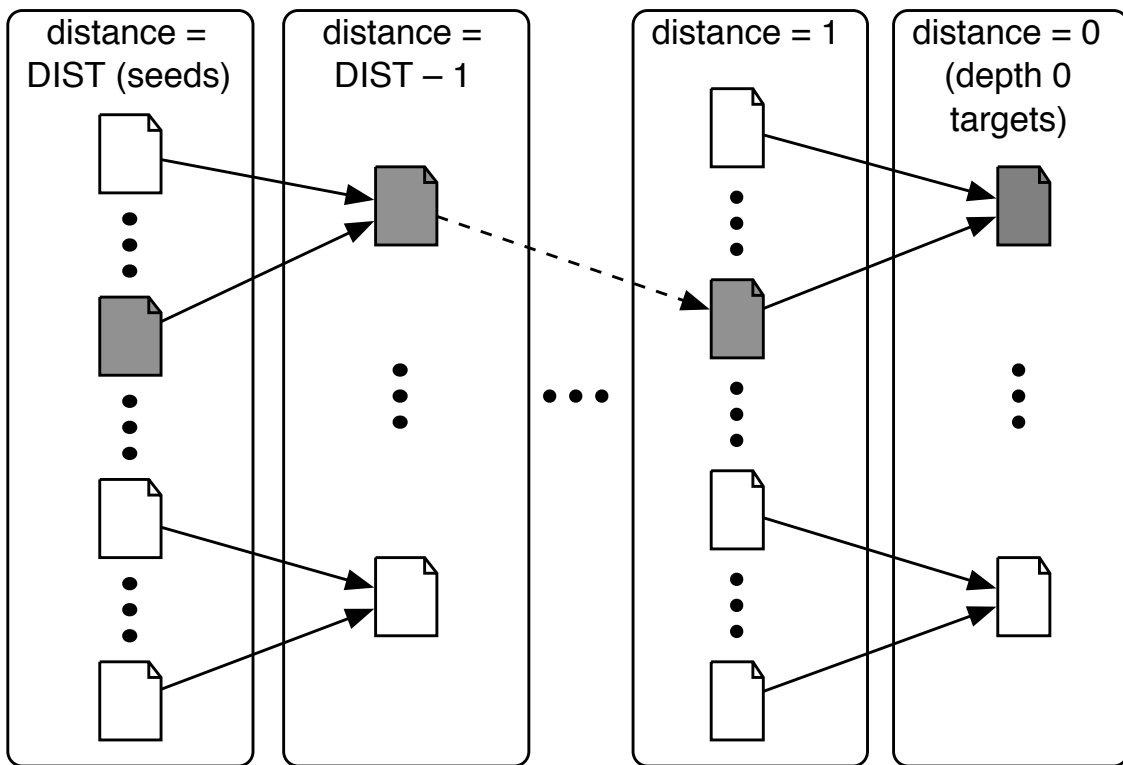


Figure 3: Illustration of the procedure to select seed pages starting from DEPTH = 0 targets and moving DIST links backwards. For DIST = 0, the seeds are a subset of the DEPTH = 0 targets. Increasing DIST makes the crawl task more and more difficult.

is a forward path of DIST from each seed to some target. The `get_backlinks()` function submits to a search engine a `link:` query for each URL in the set identified by its first argument. The search engine returns a set of backlinks, i.e. URLs of pages that link to the URL in the query. These backlinks form a new set of pages to be sampled in the next iteration of the loop. The `random_subset()` function guarantees that at most one URL is selected among the backlinks of each page from the previous iteration. The procedure may return fewer than N\_SEEDS pages because the set in the final iteration may not contain sufficient URLs. Note that the `n_sample` variable is set to ensure that over all topics, the search engine is queried exactly N\_QUERIES times. If there is no practical limitation on the number of search engine queries, N\_QUERIES can be set high enough to make `n_sample` equal to the size of the DEPTH = 0 target set. That way all DEPTH = 0 targets will be reachable from the seed set. Otherwise at most `n_sample` targets will be reachable from the seeds. More precisely, we guarantee that in the absence of broken links there exists a path with a minimum length of at most DIST links from each seed page and `n_sample` target pages at DEPTH = 0. The procedure for selecting seed pages is illustrated in Figure 3.

Observe that a breadth first crawler would have a chance in  $\ell^{\text{DIST}}$  to visit a target from a seed, assuming an average fanout of  $\ell$  links. Therefore a crawler starting from N\_SEEDS pages will find a target at most DIST links away by crawling  $\text{N\_SEEDS} * \ell^{\text{DIST}}$  pages. For  $\ell \leq 20$ , N\_SEEDS=10 and DIST=2, it would be reasonable to expect a non-zero recall of target pages after crawling N\_PAGES=4000 pages. For larger DIST values, targets become harder to find making the crawl task more difficult.

In summary, as regards crawl task definition, our framework capitalizes on the structure of hierarchical Web directories. Topics are defined as subtrees of such a hierarchy with a TOPIC\_LEVEL parameter to control for specificity of topics. Note that the approach of using leaf nodes as topics is a special case of the above with maximal TOPIC\_LEVEL and MAX\_DEPTH = 0. Alternative topic descriptions, varying in extent of detail, may be derived by considering different regions of the topic subtree via the DEPTH parameter. Target sets of relevant pages

are also identifiable for each DEPTH as the external resources linked by the directory pages. When it comes to seed pages, since the subtree really represents a single topic, a single set of seed pages is identified for each topic. This is done by starting a backlink iterative procedure from the target pages at the root of the topic subtree (DEPTH = 0). Seed pages are chosen such that, barring broken links, from each seed there is at least one target page at most DIST links away. By varying DIST we can evaluate crawlers on problems of varying difficulty. When DIST = 0, we get the common strategy of using some target pages as seeds.

### 3 Crawler Evaluation Metrics

The second major dimension of our general evaluation framework is regarding the evaluation measures required to assess crawlers. In the previous section we discussed how relevant target sets of pages may be identified. These relevant sets provide a convenient basis for computing crawler specific recall and precision scores. But the question still remains: how does one gauge the relevance of new pages, i.e., pages that are retrieved but not in the target set? Although target sets are very useful for evaluation, they have been defined using a local strategy, i.e., by exploiting the direct links from the directory pages. Thus we need measures that gauge the relevance of new pages that are retrieved.

A second aspect that needs to be addressed is the following. Assuming that we have a mechanism for assessing page relevance, we need to be able to summarize in some reasonable way the performance of a crawler. In an ideal world one would appreciate having a single number or score such that differences in scores indicate differences in value of the crawlers. However, generating a single number such as recall, precision or an F-score [40] is complicated by the fact that crawlers have a temporal dimension. Depending upon the situation, performance may need to be determined at different points of the crawl. A person interested in quickly obtaining a few relevant pages wants crawlers that return speedy dividends. For a crawler operating to establish a portal on behalf of a community of users, both high recall

and high precision are critical after a reasonably large crawl span. These are some of the issues that must be considered when deciding on methods to summarize crawler performance. Accordingly, this section discusses strategies for gauging the importance of new pages not in the target set as well as methods for summarizing crawler performance.

### 3.1 Background

Page relevance measures that have been considered are generally of two types: those that use lexical criteria and those that use link based criteria. Lexical measures show a range of sophistication. Cho *et al.* [12] explore a rather simple measure: the presence of a single word such as ‘computer’ in the title or above a frequency threshold in the body of the page is enough to indicate a relevant page. Amento *et al.* [2] compute similarity between a page’s vector and the centroid of the seed documents as one of their measures of page quality. Chakrabarti *et al.* apply classifiers built using positive and negative example pages to determine page importance [11]. Aggarwal *et al.* adopt a more generic framework that allows user designed plug in modules that specify relevance criteria for relevance [1]. The modules that they use in their tests require the presence of pre-specified words in particular parts of the page, such as the URL. Similarity to the topic measured using page text [6] or the words surrounding a link [11] may also be used to augment what are primarily link based relevance measures.

In-degree, out-degree, PageRank [7], hubs and authorities are commonly used link based page importance measures [2, 5, 6, 11, 12]. Cho *et al.* consider pages with PageRank above a specified threshold as being relevant to the query [12]. Kleinberg’s recursive notion of hubs and authorities [20] has been extended by several others. For example, edge weights are considered important [11] and so are edges that connect different sites [2, 6, 11]. Link based quality metrics rely on the generally reasonable notion of links reflecting the credibility of the target pages. Amento *et al.* show that in-degree, authority and PageRank are effective at identifying high quality pages as judged by human experts [2].

The literature shows a wide variety of summarization methods. The following are just a sample. Given a particular measure of page importance Cho *et al.* [12] examine the percentage of important pages retrieved over the progress of the crawl. Menczer *et al.* [25] measure search length, i.e., the number of pages crawled until some predetermined fraction of important pages have been visited. Chakrabarti *et al.* [11] compute the average “harvest ratio,” which is the average proportion of relevant pages retrieved over different time slices of the crawl. Harvest ratio has also been used by Aggarwal *et al.* [1]. Najork and Weiner plot the average day on which the top  $N$  pages are retrieved, where  $N$  is a variable [29] while Diligenti *et al.* examine the average relevance of pages computed using a sliding window of 200 downloads [15]. Renni and McCallum compute the percentage of relevant pages found [35]. Finally in our own research we have examined the average rank of the retrieved pages over the progress of the crawl [26].

### 3.2 Effectiveness Measures

The above variety in summarization methods for trend analysis is typical of a field that is still in its most creative phase. It is expected that as the combined evidence accumulates, some methods will begin to dominate over others. A second observation is that some summarizing methods are analogs of precision while others correspond to recall. For instance, the percentage of relevant pages retrieved over time [12] and the percentage of papers found as the percent of hyperlinks followed increases [35] are both estimates of recall. Similarly, the average rank of retrieved pages [26] and the average relevance of pages [15] are estimates of precision (although the latter is within a sliding window).

Based upon our previous experience [25, 26, 22, 32, 27] and our study of the related literature we have selected for our framework a minimal set of measures that provides for a well rounded assessment of crawler effectiveness. In addition we propose performance/cost analysis as a way to gauge the effectiveness of the crawlers against their efficiency.

Table 1 depicts our evaluation scheme. It consists of two sets of crawler effectiveness

measures differentiated mainly by the source of evidence to assess relevance. The first set focuses only on the target pages that have been identified for the topic (row 1). For example the recall measure assesses the proportion of the target set retrieved at a given point of time during the crawl. The second set of measures (row 2) employs relevance assessments based on the lexical similarity between crawled pages (whether or not they are in the target set) and topic descriptions. Further details are given below. All four measures are dynamic in that they provide a temporal characterization of the crawl strategy. Dynamic plots offer a trajectory over time that displays the temporal behavior of the crawl. We suggest that these four measures of our framework are sufficient to provide a reasonably complete picture of crawler effectiveness.

To assess page relevance using topic descriptions, the topic and retrieved pages must be represented using any reasonable, mutually compatible vector representation scheme. In our experiments topics and pages are represented by vectors of terms weighted by tf\*idf (term frequency times inverse document frequency). Further details are provided in Section 5. Given topic and page vectors,  $D$  and  $p$  respectively, their similarity may be computed as their cosine, designated by  $\sigma()$  in Table 1:

$$\sigma(p, D) = \frac{\sum_{i \in p \cap D} p_i D_i}{\sqrt{(\sum_{i \in p} p_i^2)(\sum_{i \in D} D_i^2)}} \quad (1)$$

where  $p_i$  and  $D_i$  are the tf\*idf weights of term  $i$  in page  $p$  and topic description  $D$ , respectively.

The key difficulty in calculating recall for the full crawl set is that the number of relevant documents in the collection is unknown. The best we can do is to require the crawler to retrieve *only* relevant pages. That is, if a crawler is given a lifespan  $S$ , i.e., it is stopped after retrieving  $S$  pages, then we expect all  $S$  pages to be relevant. The problem is that this criteria is unsatisfiable in the many cases in which topics have fewer than  $S$  relevant targets. However, this problem is considerably mitigated since our goal is to compare crawlers and not assess them in isolation. Thus, by requiring crawlers to only retrieve relevant pages and

estimating page relevance through lexical similarity we may estimate recall for the full crawl set by accumulating similarity over the crawled pages. The ideal crawler will achieve at each point of time the maximum possible similarity. True recall calculations would require division by the number of relevant pages, in this case by  $S$ . However, since this is now a constant across crawlers and topics we may drop it from the calculations. For precision, the proportion of retrieved pages that is relevant is estimated as the average similarity score of retrieved pages. In this way, our framework allows for a well rounded analysis with analogs of recall and precision performance measures using both a known target set of relevant pages as well as topic descriptions to assess the relevance of any crawled page. Finally, by plotting these measures over time, we get a dynamic characterization of performance.

### **3.3 Efficiency**

Crawlers consume resources: network bandwidth to download pages, memory to maintain private data structures in support of their algorithms, CPU to evaluate and select URLs, and disk storage to store the processed text and links of fetched pages. Obviously the more complex the link selection algorithm, the greater the use of such resources. In order to allow for a fair comparison of crawling algorithms, our framework prescribes tracking the CPU time taken by each crawler for each page and each topic while ignoring the time taken by fetching, parsing and storing routines common to all the crawlers. We do this since it is impossible to control for network traffic and congestion, and we want to benchmark only the crawler-specific operations. The monitored CPU time will be used to compare the complexity of the crawling algorithms and gauge their performance against their efficiency.

## **4 Characterization of Topics**

The third dimension of our evaluation framework pertains to topic characteristics. In information retrieval research it is understood that query characteristics affect performance

[30, 36, 3, 28]. In the classic 1988 study by Saracevic and Kantor [36], query characteristics were explored within a larger context that included the study of users and search methods. Their questions were classified by expert judges regarding: domain (subject), clarity, specificity, complexity and presupposition. They found for example that the number of relevant documents retrieved was higher in questions of low clarity, low specificity, high complexity and many presuppositions. Beaulieu *et al.* correlated search outcomes with query characteristics examining aspects such as topic type [3]. Mitra and colleagues explore the effect of query expansion strategies by differentiating queries based on their initial retrieval performance [28].

There is also active research on the types of queries users input to search engines [38, 19]. For example in [38] the authors study over a million queries posed to the Excite search engine and find that the language of Web queries is distinctive in that a great many terms are unique. A key difference in our general framework is that topics form the basis of our focused crawls and not user provided questions. Moreover, given our underlying retrieval context, we seek to exploit topic features that derive from the Web.

Topic features are seldom explored in crawler research. An exception is when topic features are examined in order to elaborate on observed performance and to provide an explanation of results. For example, Chakrabarti *et al.* [11] discuss a few of their twenty topics from Yahoo in detail in order to elaborate on their crawler mechanisms and to explore their notions of cooperative and competitive domains. Although Bharat and Henzinger [6] do not differentiate between their 28 topics they do present results for the full topic set and for two special subsets: rare and popular topics as determined by the retrieval set size from AltaVista. Amento *et al.* [2] experiment on a set of five topics that are somewhat homogeneous in that they are all representative of popular entertainment topics. Menczer and Belew [25] test two crawlers (InfoSpiders and best-first) on topics from a limited Encyclopaedia Britannica (EB) corpus and analyze the dependence of performance on the depth of the topics within the EB subject hierarchy, where deeper topics are more specific.

In our general framework for crawler evaluation research, we seek to include consideration of topic characteristics that hold some potential for increasing our understanding of crawler performance. Thus our framework should allow one to look for significant correlations, positive or negative, between topic characteristics and performance. We begin our exploration by discussing the following four distinct characteristics that have proved significant in preliminary experiments with our framework:

**Topic Popularity:** Number of pages containing topic keywords;

**Target Cohesiveness:** Cliquishness of target pages in link space;

**Target Authoritativeness:** Average authority score of target pages among neighbor pages;

**Seed-Target Similarity:** Average similarity between seed pages and target descriptions.

## 4.1 Popularity

Popularity indicates the level of interest in the topic. More popular topics will have larger numbers of interested individuals, related Web pages and discourse units. For instance “IBM computers” could be a more popular topic than “Web crawlers.” We are interested in this property because it may be the case that crawler differences are accentuated when we pay attention to the popularity of topics. For example, some crawlers may perform more poorly on more popular topics if they are too reliant on lexical clues.

Topic popularity may be estimated by the size of its discourse set. One way to do this is to search for the topic keywords directly against a search engine and use the number of hits returned as this estimate. If multiple search engines are employed then the average number of hits returned may be used as this estimate. Note that for this a suitable query representation of each topic for conducting the search is needed. If the topic’s query representation is depth sensitive, i.e., dependent upon the value of DEPTH, then popularity estimates corresponding to different query representations of the topic can be made. Thus we define *popularity* for a

topic  $t$  at DEPTH  $d$  as:

$$P_d(t) = \frac{1}{|E|} \sum_{e \in E} |H_e(K(t))| - \sum_{t' \in G_{d+1}(t)} |H_e(K(t'))| \quad (2)$$

where  $K(t)$  is the keyword representation of topic  $t$  (i.e., the concatenation of node labels from the root of the directory to node  $t$ ),  $H_e(q)$  is the hit set returned by search engine  $e$  in response to query  $q$ ,  $E$  is a set of trusted search engines, and  $G_{d+1}(t)$  is the set of subnodes (subtopics) of  $t$  at depth  $d + 1$ . Thus for  $d = 0$  we look at the popularity of the topic in the most restrictive sense, excluding keywords of any subtopic. For  $d = \text{MAX\_DEPTH}$ , we interpret the topic in a more inclusive sense, corresponding to the whole topic subtree. Note that the keywords in  $K$  are joined by AND syntax (all required) and thus  $P_d$  is a non-decreasing function of  $d$  for any topic.

## 4.2 Cohesiveness

Topic cohesiveness estimates how closely knit the relevant pages are for a topic. The more cohesive a topic, the more interlinked its set of relevant pages. For this we start with the target pages of the topic as these are the ones assumed to be relevant. Cohesiveness is obtained by examining the link features of the target pages. More specifically the focus is on the neighborhood of the target pages. We use the forward links from all target pages and count the fraction of these that point back to the target set:

$$C_d(t) = \frac{\sum_{u \in T_d(t)} |O_u \cap T_d(t)|}{\sum_{u \in T_d(t)} |O_u|} \quad (3)$$

where  $O_u$  is the set of outlinks from page  $u$ . Note that since target sets are DEPTH sensitive, so is our topic cohesiveness metric.

Cohesiveness is a measure of the “cliquishness” of the target pages, and has been used in many contexts, for example to characterize the performance of a random-walk crawler

[21, 23] and to identify Web communities [16]. We speculate that topics with high link cohesiveness could potentially make it easier for a crawler to stay within the vicinity of the relevant pages. This would be especially true for crawlers with localized search strategies.

### 4.3 Authoritativeness

The next topic characteristic metric in our framework pertains to authoritativeness. As proposed by Kleinberg [20] a good authority is a page that has several good hubs pointing to it while a good hub is one that points to several good authorities. Kleinberg provides us with an algorithm that uses this recursive definition on a directed graph of Web pages to get authority and hub scores. We treat the target pages of a topic as a *root set* which is then expanded to get a *base set*. The expansion is done by including the pages corresponding to all the outlinks from the root set pages and the top  $I$  inlinks to the root set. Kleinberg’s algorithm is then applied to the graph representation of the base set. Once the algorithm converges, we calculate the average authority score for the target URLs:

$$A_d(t) = \frac{1}{|T_d(t)|} \sum_{u \in T_d(t)} \Lambda(u, B(T_d(t))) \quad (4)$$

where  $B(T)$  is the base set obtained from root set  $T$  and  $\Lambda(u, B)$  is the convergence authority score for page  $u$  computed from base set  $B$ .

Since the authority scores are normalized, the average authority score  $A_d(t)$ , which we call *authoritativeness*, represents the concentration of authority in the target pages of topic  $t$  as inferred from their link based neighborhood. By taking target sets at different values of DEPTH  $d$ , we obtain depth sensitive estimates of topic authoritativeness.

### 4.4 Seed-Target Similarity

The last topic characteristic included in our framework is seed to target similarity. Here the point explored is that if the targets are lexically very similar to the seeds then it may be

easier to reach the target pages. Thus we differentiate between topics on the basis of the average lexical similarity between the seed pages and the target descriptions:

$$L_d(t) = \frac{1}{|S(t)|} \sum_{u \in S(t)} \sigma(u, D_d(t)) \quad (5)$$

where  $S(t)$  is the seed set for topic  $t$ .

Once again, seed page  $u$  and target description  $D_d$  may be any reasonable vector representation. Similarity  $\sigma$  is then defined as the cosine of the two vectors (see Equation 1). Typically, tf\*idf weighted term vectors are used. Our specific implementation of weight representation is detailed in Section 5. As for the other topic characteristics, seed-target lexically is DEPTH sensitive.

## 5 Case study

Our next goal is to demonstrate the application of the general evaluation framework presented in the previous sections, in an experiment comparing four off-the-shelf crawlers. These consist of two crawlers built using a best-first algorithm, one implementation of the InfoSpiders crawler based on adaptive agents, and a breadth first crawling algorithm. In this case study we describe the specific implementation of the framework, i.e., the choice of parameter values and the decisions related to the three dimensions of our evaluation: crawl task, performance measures, and topic characteristics.

### 5.1 Crawl Task

We use the Open Directory ([dmoz.org](http://dmoz.org)) hierarchy as our source for topics. Two key advantages of this choice are that (i) the Open Directory is maintained by a large number of volunteer editors and thus is not strongly biased toward commercial content, and (ii) it makes all of its data publicly and freely available through periodic RDF dumps.

We identified  $N\_TOPICS = 100$  topics from this hierarchy at  $TOPIC\_LEVEL = 3$  and  $MAX\_DEPTH = 2$ . By varying  $DEPTH$  from 0 to 2 we generated topic descriptions and target sets. Each topic node contributes to the topic description a concatenation of the text descriptions and anchor text for the target URLs, written by the DMOZ human editors (cf. Figure 2). Thus we have 3 sets of descriptions and 3 sets of target pages for each topic. The experiments described are differentiated by topic  $DEPTH$ .

In addition, a set of keywords is defined for each topic. The keywords associated with a particular node are the words in the DMOZ hierarchy down to that node. The keywords are used to guide crawlers in their search for topical pages. For example the best links in a best-first algorithms are selected based on source page similarity to a topic representation build out of the these keywords. Keywords corresponding to different depths than the topic root node ( $DEPTH > 0$ ) may also be used to compute topic popularity as described in Section 4.1. Table 2 provides as an example one of the 100 topics in our case study.

For seed selection we use the procedure described in section 2.3. Backlinks are obtained via the Google Web API. Since the API has a limit of 1000 queries per day, we set  $N\_QUERIES = 1000$ . The other parameters are  $DIST = 2$  and  $N\_SEEDS = 10$ . Thus at each iteration in the procedure we select  $n\_sample = 5$  backlinks. Barring any broken links, each of the 10 seed pages can lead to at least one target page at  $DEPTH = 0$  within at most 2 links.

## 5.2 Evaluation Metrics

To evaluate the crawlers in our case study we follow closely the performance measures defined in Table 1 (Section 3.2).

When assessing relevance of the full crawl set against topic descriptions, both the target descriptions and the retrieved pages are pre-processed by removing common “stop words” and by a standard stemming algorithm [34]. They are then represented by  $tf*idf$  vectors. Moreover,  $DEPTH$  dependent topic vectors are generated by concatenating the topic keywords and the topic descriptions down to the corresponding  $DEPTH d$ . Our  $idf$  calculations are also

Relevance Assessments	Recall	Precision
Target Pages	$ S_c^t \cap T_d / T_d $	$ S_c^t \cap T_d / S_c^t $
Target Descriptions	$\sum_{p \in S_c^t} \sigma(p, D_d)$	$\sum_{p \in S_c^t} \sigma(p, D_d)/ S_c^t $

Table 1: Evaluation scheme.  $S_c^t$  is the set of pages crawled by crawler  $c$  at time  $t$ .  $T_d$  is the target set and  $D_d$  is the vector representing the topic description, both at depth  $d$ . Finally  $\sigma$  is the cosine similarity function.

DEPTH	Keywords	Descriptions	Targets
0	Sports Disabled Wheelchair	ChairSports.com - Information and links on... dizABLED: Wheelchair Stuntman Cartoons... National Wheelchair Poolplayer Association... Wheelchair Adventures - Discusses various... Wheelchair Sports - A celebration of active... World Wheelchair Sports - Homepage of this... Xtreme Medical Sports - Organization in...	<a href="http://www.chairsports.com/">http://www.chairsports.com/</a> <a href="http://www.dizabled.com/">http://www.dizabled.com/</a> <a href="http://www.nwpainc.com/">http://www.nwpainc.com/</a> <a href="http://www.afdl00104.pwp.blueyonder...">http://www.afdl00104.pwp.blueyonder...</a> <a href="http://lenmac.tripod.com/sports.html">http://lenmac.tripod.com/sports.html</a> <a href="http://www.efn.org/~wvcoach/">http://www.efn.org/~wvcoach/</a> <a href="http://www.xtrememedical.com/">http://www.xtrememedical.com/</a>
1	Events Regional	British Commonwealth Paraplegic Games- Brief... Pan-American Wheelchair Games- Brief history... Stoke Mandeville Wheelchair Games - Brief... The Wheelchair Bodybuilding Page - Lists... Hamilton Wheelchair Relay Challenge...	<a href="http://www.internationalgames.net/br...">http://www.internationalgames.net/br...</a> <a href="http://www.internationalgames.net/pa...">http://www.internationalgames.net/pa...</a> <a href="http://www.internationalgames.net/st...">http://www.internationalgames.net/st...</a> <a href="http://www.angelfire.com/ky/thawes/">http://www.angelfire.com/ky/thawes/</a> <a href="http://www.hamilton-wheelchair-relay...">http://www.hamilton-wheelchair-relay...</a>
2	Australia Canada Hong Kong UK US	ElazeSports.com - A disabled sports program... Far West Wheelchair Sports- Events, results... Long Island Wheelchair Athletic Club (LIWAC)... Paralyzed Veterans Association of Florida... Sun Wheelers Sports- Non-profit organization... BC Wheelchair Sports Association- Non-profit... Canadian Wheelchair Sports Association... Manitoba Wheelchair Sport Association- Sport... Ontario Wheelchair Sports Association Canada... Wheelchair Sports Association Newfoundland... i-Wheel Sports: NSW Wheelchair Sport... New South Wales Wheelchair Sport - General... British Wheelchair Sports Foundation (BWSF)...	<a href="http://www.blazesports.com/">http://www.blazesports.com/</a> <a href="http://home.earthlink.net/~fwvaa/">http://home.earthlink.net/~fwvaa/</a> <a href="http://www.liwac.org/">http://www.liwac.org/</a> <a href="http://www.pvaf.org/">http://www.pvaf.org/</a> <a href="http://www.geocities.com/sun_wheelers/">http://www.geocities.com/sun_wheelers/</a> <a href="http://www.bcwheelchairsports.com/">http://www.bcwheelchairsports.com/</a> <a href="http://www.cwsa.ca/">http://www.cwsa.ca/</a> <a href="http://www.sport.mb.ca/wheelchair/">http://www.sport.mb.ca/wheelchair/</a> <a href="http://www.disabledsports.org/owsa.htm">http://www.disabledsports.org/owsa.htm</a> <a href="http://www.netfx.ca/wsanl/">http://www.netfx.ca/wsanl/</a> <a href="http://www.nswwsa.org.au/">http://www.nswwsa.org.au/</a> <a href="http://www.isport.com.au/wheels/nswws/">http://www.isport.com.au/wheels/nswws/</a> <a href="http://www.britishwheelchairsports.org/">http://www.britishwheelchairsports.org/</a>

Table 2: A sample topic. For each DEPTH, only additional keywords, descriptions and targets are shown; the actual descriptions and target sets at each DEPTH  $d$  are inclusive of those for all DEPTH  $< d$ . Descriptions and target URLs are abridged for space limitations.

done with respect to the pool consisting of target descriptions down to DEPTH  $d$  in the topic subtree. We compute the tf\*idf weight of term  $i$  in page  $p$  for topic  $t$  and depth  $d$  as follows:

$$p_{t,d}(i) = f(i, p) \cdot \left( 1 + \ln \left( \frac{|D_d(t)|}{|\{q \in D_d(t) : i \in q\}|} \right) \right) \quad (6)$$

where  $f(i, p)$  is the number of occurrences of  $i$  in  $p$  and  $D_d(t)$  is the set of target description for topic  $t$  and depth  $d$ . The tf\*idf weights for topic vectors are computed analogously. For lexical similarity, the cosine formula in Equation 1 is used.

### 5.3 Topic Characteristics

In our case study we limit the analysis of topic characteristics to topic DEPTH = 2 only. We calculate topic popularity by searching each topic keywords against only the Google search engine, using the Google Web API. Searches are generated from the most inclusive interpretation of each topic, using just keywords at DEPTH=0. Topic cohesiveness has been fully specified in the discussion in section 4.2. For topic authoritativeness, when generating the base set we use  $I = 10$ , i.e., we add to the base set the top 10 inlinks as retrieved by Google. This is due to the API's limitation of 10 results per query. We then apply Kleinberg's algorithm to this base set and calculate the authority score for each page in the target set as described in Section 4.3. For seed-target similarity, the pages (after stop word removal and stemming) are represented using tf\*idf vectors (cf. Equation 6) and the cosine function defined in Equation 1 is used for similarity calculations.

### 5.4 Crawling algorithms

We use the framework just specified to compare four crawlers. The choice of crawling algorithms for this case study is based on crawlers that are well-known in the literature and that either have proved effective in prior research, or have been routinely used as baseline performers.

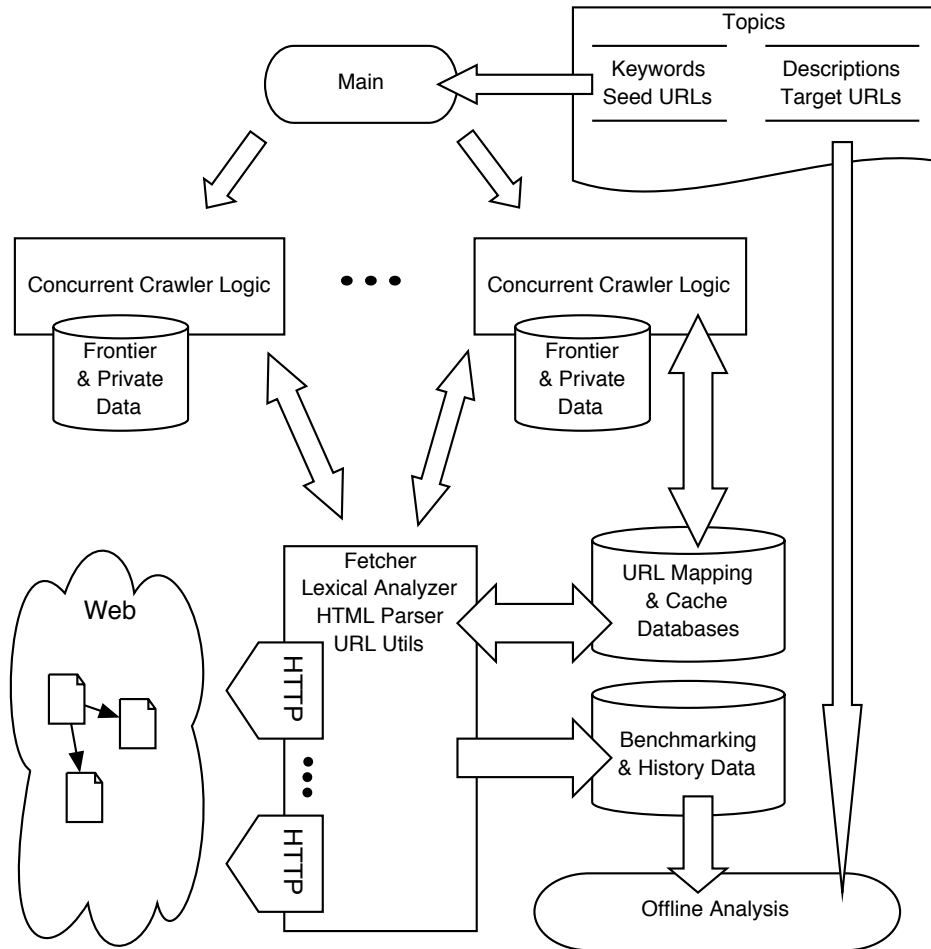


Figure 4: Architecture of our crawling system. Crawling algorithms run concurrently and are specified in modules that share common utilities (HTTP fetcher, HTML parser, URL processing, stopper, stemmer, lexical analysis, and benchmarking) and databases (cache and data collection). Each crawler module also maintains private data structures that are limited in size.

Figure 4 illustrates our architecture for crawling the Web according to the various algorithms. All crawlers are given the same topic keywords and seed URLs, and perform the following basic procedure:

```

crawler (keywords, seeds, N_PAGES, MAX_BUFFER) {
  frontier = seeds;
  repeat N_PAGES times {
    link = process(frontier, keywords);
    new_links = visit(link);
    push(frontier, new_links);
    maintain(frontier, MAX_BUFFER);
  }
}

```

The comparison is made under the constraint of limited resources, i.e., we limit the memory available to each crawler by constraining the size of its internal buffer. This buffer is used by a crawler to temporarily store link data, typically a frontier of pages whose links have not yet been explored. Each crawler is allowed to track a maximum of `MAX_BUFFER` links. If the buffer becomes full, the crawler must decide which links are to be substituted as the new ones are added. The value of `MAX_BUFFER` is set to 256 in our case study. Crawlers are given a lifespan of `N_PAGES = 4000` pages.

The crucial details that differentiate crawling algorithms are in the `process` function. The first crawler tested is a breadth-first crawler which is the simplest strategy for crawling. It uses the frontier as a FIFO queue, crawling links in the order in which it encounters them. The `BreadthFirst` crawler is used here because it provides us with a baseline performance level that can help gauge the effectiveness of more sophisticated algorithms.

The next two crawlers are variations of best-first search [12, 18]. In its basic version (`BFS1`), given a frontier of links, the best link according to some estimation criterion is selected for crawling. `BFSN` is a generalization in that at each iteration a batch of top  $N$  links to crawl are selected. Here we use `BFS256`, which has proved effective in our prior research [32, 27]. Topic keywords are used to guide the crawl. Link selection occurs by computing the cosine similarity between the keyword vector and the source page vector, for

each link. The  $N$  URLs with the best source page similarities are then selected for crawling.

The last crawler tested is an implementation of InfoSpiders [21, 24, 25, 27]. A population of agents crawls in parallel using adaptive keyword vectors and neural nets to decide which links to follow. An evolutionary algorithm uses a fitness measure based on similarity as a local selection criterion, and reinforcement learning to train the neural nets for predicting which links lead to the best pages based on their textual context in a source page. Agents that visit many pages similar to their internal keyword vectors get a chance to create offspring. An offspring inherits the keywords and neural net of the parent, modulo some mutations designed to internalize the features of the pages that led to the parent’s success. The algorithm is completely distributed, with no interaction between distinct agents. Therefore this IS crawler can maximally exploit our concurrent architecture for efficiency.

Further details of the four crawlers used in this case study are beyond the scope of this article. We refer the reader to a companion paper [27] where these crawlers are described and analyzed at much greater depth.

## 5.5 Performance Analysis

Figures 5 and 6 show the performance analysis results for the four crawlers using our evaluation framework’s effectiveness measures. The results are consistent across measures and with our prior experiments on these crawlers. In general we observe that BFS1 does well in the early stages of the crawl, but then pays a price for its greedy behavior [32]. BFS256 eventually catches up, and in the case of target pages it outperforms the other crawlers. IS is outperformed by both BFS crawlers based on descriptions, while it almost matches the performance of BFS1 based on target pages. As expected BreadthFirst displays the worst performance and provides us with a baseline for all measures. The main difference between these and our prior results is that in a more difficult task ( $\text{DIST} = 3$ ) we found IS to be competitive with BFS256 and better than BFS1 [27].

Precision and recall measures do provide complementary information in the evaluation.

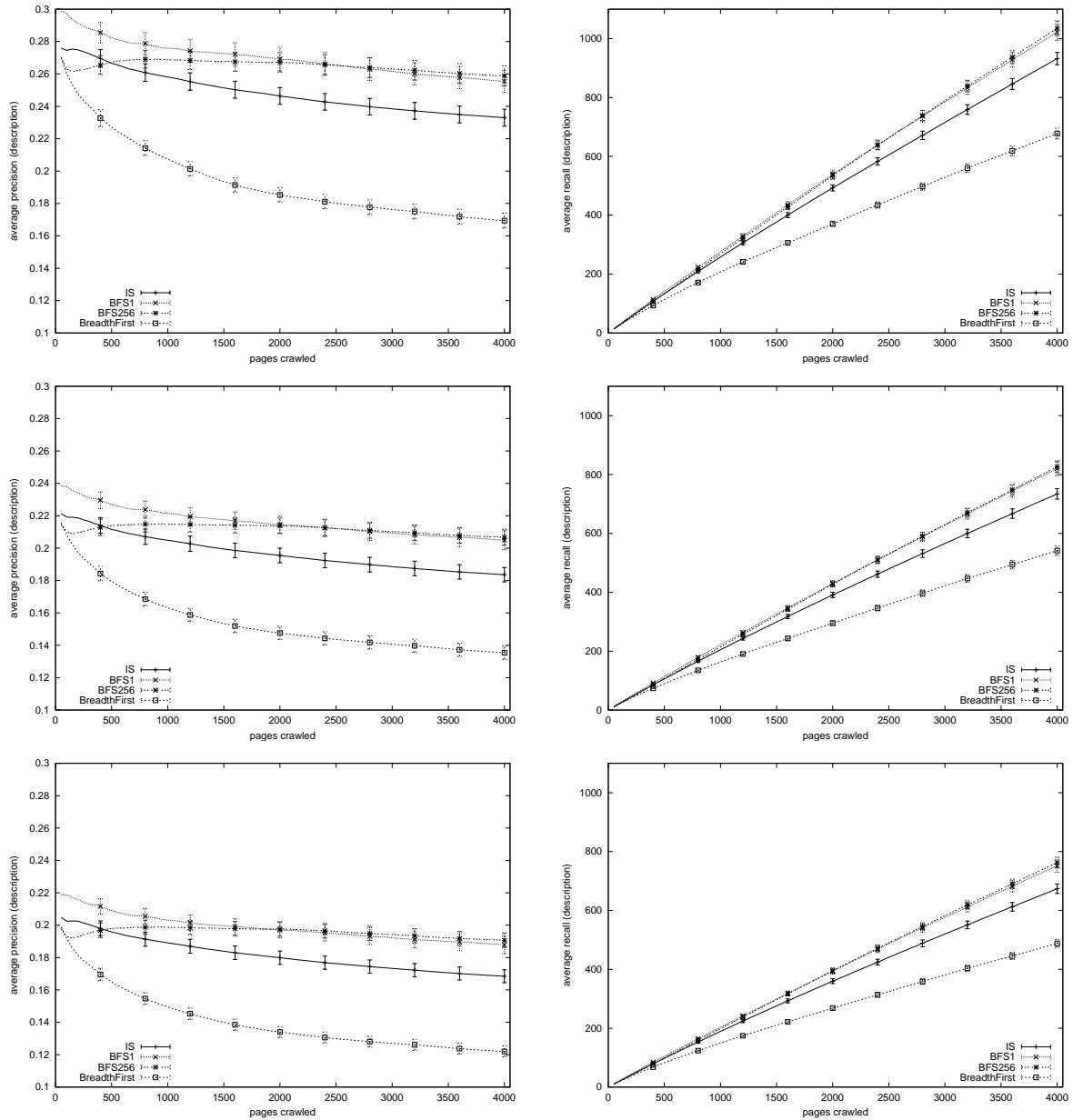


Figure 5: Dynamic plots of precision (left) and recall (right) versus crawled pages for relevance assessments based on target descriptions at DEPTH 0 (top), 1 (center), and 2 (bottom). Performance is averaged across topics and standard errors are also shown.

Precision captures a more dynamic and textured view of the behavior of the different crawling algorithms, especially in the early stages of the crawls. Recall provides for a clearer picture of the overall difference in the crawlers’ asymptotic performance. Performance generally decreases with increasing DEPTH, indicating that more inclusive interpretations of topics make for more difficult crawling applications. The one exception to this trend is provided by the target precision measure, which increases with DEPTH. This is because for most topics the number of target pages increases very quickly with DEPTH. So even though more targets are visited, they represent a smaller fraction of the target set.

These results provide us with a clear picture of the effectiveness of the different crawlers, but do not account for the computational complexity of the crawling algorithms. To gauge performance by the efficiency of the crawlers, Figure 7 shows the results of a performance/cost analysis based on our evaluation framework. Here we focus on recall measures, and on target descriptions and pages at DEPTH = 0. The results are quite interesting. Due to its efficiency BreadthFirst displays the best performance/cost ratio in the early stages of the crawl — if we need a few results really fast the simplest strategy may be the way to go. In the long run, IS achieves the highest performance/cost ratio thanks to its competitive performance and efficient use of concurrency. The BFS crawlers are penalized by our less efficient implementations of their algorithms, which require frequent sorting and synchronization operations.

## 5.6 Topic Analysis

To analyze how crawler behavior is affected by different topics let us consider the correlation between performance and the various topic characteristics defined in Section 4. Here we need to pair a topic’s characteristic with a crawler’s performance, so we use the cohesiveness, authoritativeness, popularity, and seed similarity measures for the former and the recall levels achieved by each crawler after 4000 pages for the latter. Since the distributions of all these measures are unknown, we need a distribution-free correlation measure and to this end we use Spearman’s rank correlation coefficient  $\rho$ .

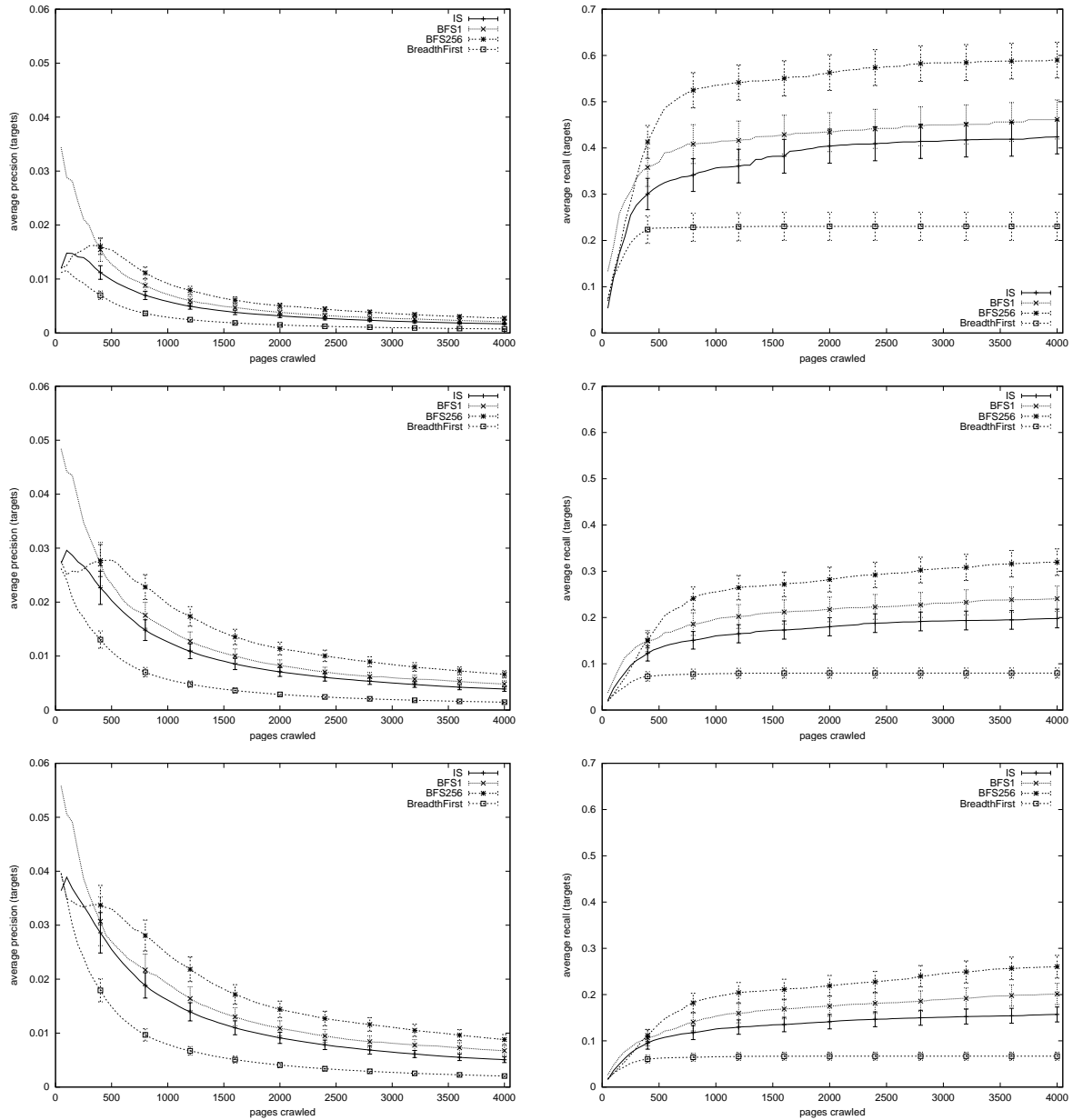


Figure 6: Dynamic plots of precision (left) and recall (right) versus crawled pages for relevance assessments based on target pages at DEPTH 0 (top), 1 (center), and 2 (bottom). Performance is averaged across topics and standard errors are also shown.

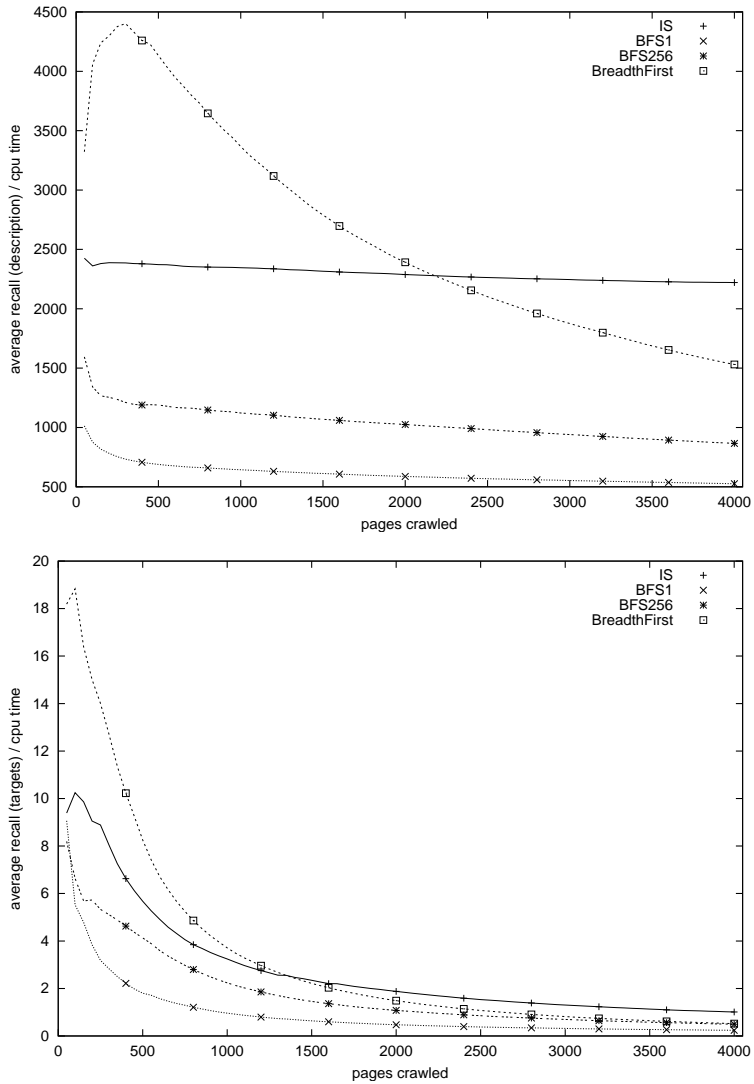


Figure 7: Dynamic plots of recall over relative CPU time for relevance assessments based on target descriptions (top) and target pages (bottom) at  $\text{DEPTH} = 0$ . CPU time is relative to its mean across crawlers to account for differences in the CPU speeds of machines used in our experiments. Performance and CPU times are then averaged across topics before their ratio is computed.

Table 3 shows the values of  $\rho$  for each crawler and topic characteristic, based on recall performance from target pages and target descriptions. Seed-target similarity is the topic characteristic that most significantly affects performance across crawlers. Higher seed-target similarity not only improves performance based on topic description, but also helps in reaching more predefined targets. The strong correlation may be indicative of the generally accepted principle that Web pages tend to point to lexically similar pages [23]. With that in mind, we also note that all of the topical crawlers (IS, BFS1 and BFS256) are more exploitative of seed-target similarity and hence show higher correlation than BreadthFirst.

While topic cohesiveness has no significant effect on target page recall, it does have a significant influence on description based performance. We interpret this observation by arguing that a cohesive topic may provide many paths to lexically similar pages even while identifying target pages may remain non trivial.

A topic’s authoritativeness topic does not significantly influence any crawler other than BreadthFirst. Since the latter is not a topical crawler, it is able to improve its performance in reaching the targets simply because there are more paths leading to them — authoritative targets are like attractors because they have many inlinks. This is consistent with observations that BreadthFirst crawlers effectively retrieve pages with high PageRank [29].

Topic popularity seems to have contradicting effects on the two evaluation measures. This is highlighted by the performance of InfoSpiders; IS is hindered by topic popularity when relevance is assessed through target pages, and helped when performance is based on target descriptions. Information overload due to high topic popularity makes it hard to identify a relevant subset such as the target set. On the other hand the large relevant set of a popular topic makes it easy to find many relevant pages.

As an illustration of the correlations in this data, Figure 8 shows a scatter plot of performance versus seed-target similarity for IS. For comparison, linear regressions are plotted for both IS and BreadthFirst. The plot makes it evident that IS tends to visit more relevant target pages when it starts from seeds that are lexically similar to the target descriptions.

Crawler	Target pages recall				Target description recall			
	$C_2$	$A_2$	$P_2$	$L_2$	$C_2$	$A_2$	$P_2$	$L_2$
IS	+0.15	+0.17	-0.19	<b>+0.54</b>	<b>+0.41</b>	-0.08	<b>+0.20</b>	<b>+0.37</b>
BFS1	+0.03	-0.01	-0.18	<b>+0.41</b>	<b>+0.31</b>	-0.06	+0.07	<b>+0.35</b>
BFS256	+0.12	+0.05	-0.14	<b>+0.53</b>	<b>+0.32</b>	-0.02	+0.10	<b>+0.41</b>
BreadthFirst	+0.15	<b>+0.27</b>	-0.18	<b>+0.31</b>	<b>+0.36</b>	-0.14	+0.12	<b>+0.28</b>

Table 3: Rank correlation coefficients between each crawler’s recall after 4000 pages and the four topic characteristics. Recall is based either on target pages (left) or target descriptions (right). Values of  $\rho$  in bold indicate significant correlations at the 95% confidence level, based on a two-tailed Spearman rank correlation test [13]. In these 14 cases we can refute the null hypothesis that there is no monotonic relationship between performance and topic characteristic.

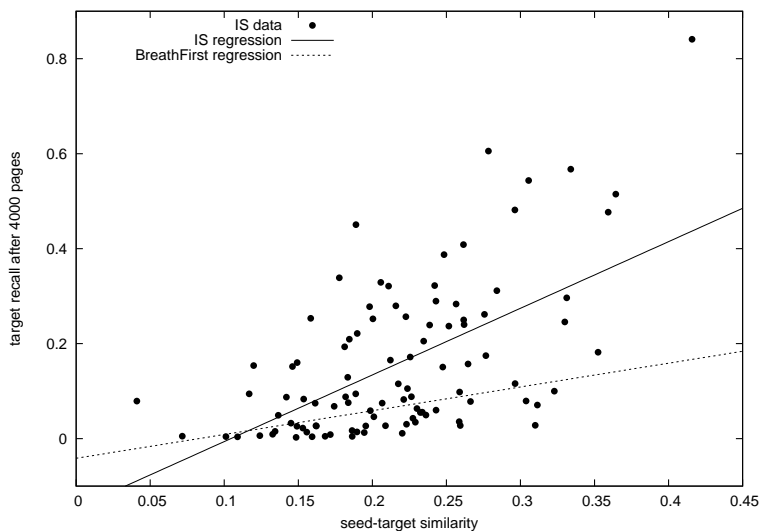


Figure 8: Scatter plot of target page recall for IS versus  $L_2$ . A linear regression is also shown for both IS and BreadthFirst.

## 6 Conclusions

In this paper we presented a general framework to evaluate topical crawlers. We identified a class of tasks that model crawling applications of different nature and difficulty. By relying on Web directories, topics with the desired mix of specificity and inclusiveness can be easily identified.<sup>1</sup> One important limitation of this approach is its dependence on the availability of a hierarchical directories as a topic source. The Open Directory currently provides us with such a public resource, while other directories may be less open due to commercial concerns.

The framework also specifies a procedure for defining crawling tasks of variable difficulty by selecting seed pages at appropriate distances from targets. The goal of such a formal and systematic characterizations of crawl tasks is to foster quantitative experiments that may allow researchers to better understand the differences between the many crawling applications found in the literature.

We introduced a set of performance measures to evaluate Web crawlers along several dimensions: precision versus recall, relevance criteria based on target pages versus human-compiled target descriptions, topic breadth, algorithmic efficiency, and dependence on diverse topic characteristics. This is the most comprehensive treatment of topical crawler evaluation issues to date. Such a framework should assist researchers in making objective comparative evaluations between crawlers and across studies.

The results of our case study clearly demonstrate that the proposed framework is effective at evaluating, comparing, differentiating, and interpreting the performance of diverse crawlers along all the above-mentioned dimensions.

Topic analysis gives further insight into the behavior of crawling algorithms. Given a particular crawler, we may be able to predict its performance from the value of a topic characteristic, based on its sensitivity to that characteristic. For example we have shown that the IS crawler is most sensitive to the popularity of topics, visiting more relevant pages

---

<sup>1</sup>A script that selects topics from the Open Directory based on a number of parametric specifications, and generates a file containing topic keywords, descriptions and target URLs at various depths, is available from the authors upon request.

(as assessed based on target descriptions) when topics are more popular. In future research we plan to develop and test additional topic characteristics, such as recency and update frequency of topic target pages.

Since the main emphasis of this paper is on presenting our general evaluation framework, we did not perform the many experiments suggested by the possibility of varying parameters such as `TOPIC_LEVEL` and `MAX_DEPTH`. Such experiments, together with the evaluation of the many other crawlers in the literature, are left for future research.

## Acknowledgments

Thanks to Alberto Segre, Dave Eichmann, Miguel Ruiz and other colleagues for their support and contributions to this and our prior work. We are grateful to the Open Directory and its editors for their work and for making their data publicly available. This work is supported in part by the National Library of Medicine under grant No. RO1-LM06909 to PS and by the National Science Foundation under CAREER grant No. IIS-0133124 to FM.

## References

- [1] CC Aggarwal, F Al-Garawi, and PS Yu. Intelligent crawling on the World Wide Web with arbitrary predicates. In *Proc. 10th International World Wide Web Conference*, pages 96–105, 2001.
- [2] B Amento, L Terveen, and W Hill. Does "authority" mean quality? Predicting expert quality ratings of Web documents. In *Proc. 23rd ACM SIGIR Conf. on Research and Development in Information Retrieval*, pages 296–303, 2000.
- [3] M Beaulieu, H Fowkes, and H Joho. Sheffield interactive experiment at TREC-9. In *Proc. 9th Text Retrieval Conference (TREC-9)*, 2000.
- [4] I Ben-Shaul et al. Adding support for dynamic and focused search with Fetuccino. *Computer Networks*, 31(11–16):1653–1665, 1999.
- [5] I Ben-Shaul, M Herscovici, M Jacovi, YS Maarek, D Pelleg, M Shtalhaim, V Soroka, and S Ur. Adding support for dynamic and focused search with Fetuccino. *Computer Networks*, 31(11–16):1653–1665, 1999.
- [6] K Bharat and MR Henzinger. Improved algorithms for topic distillation in hyperlinked environments. In *Proc. 21st ACM SIGIR Conf. on Research and Development in Information Retrieval*, pages 104–111, 1998.

- [7] S Brin and L Page. The anatomy of a large-scale hypertextual Web search engine. *Computer Networks*, 30(1–7):107–117, 1998.
- [8] S Chakrabarti, B Dom, P Raghavan, S Rajagopalan, D Gibson, and J Kleinberg. Automatic resource compilation by analyzing hyperlink structure and associated text. *Computer Networks*, 30(1–7):65–74, 1998.
- [9] S Chakrabarti, MM Joshi, K Punera, and DM Pennock. The structure of broad topics on the Web. In *Proc. 11th International World Wide Web Conference*. ACM Press, 2002.
- [10] S Chakrabarti, K Punera, and M Subramanyam. Accelerated focused crawling through online relevance feedback. In *Proc. 11th International World Wide Web Conference*. ACM Press, 2002.
- [11] S Chakrabarti, M van den Berg, and B Dom. Focused crawling: A new approach to topic-specific Web resource discovery. *Computer Networks*, 31(11–16):1623–1640, 1999.
- [12] J Cho, H Garcia-Molina, and L Page. Efficient crawling through URL ordering. *Computer Networks*, 30(1–7):161–172, 1998.
- [13] WJ Conover. *Practical Nonparametric Statistics*, chapter 5, pages 213–343. Wiley, New York, 1980.
- [14] PME De Bra and RDJ Post. Information retrieval in the World Wide Web: Making client-based searching feasible. In *Proc. 1st International World Wide Web Conference*, 1994.
- [15] M. Diligenti, F. Coetzee, S. Lawrence, C. L. Giles, and M. Gori. Focused crawling using context graphs. In *Proc. 26th International Conference on Very Large Databases (VLDB 2000)*, pages 527–534, Cairo, Egypt, 2000.
- [16] GW Flake, S Lawrence, and CL Giles. Efficient identification of Web communities. In *Proc. 6th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pages 150–160, Boston, MA, August 20–23 2000.
- [17] MR Henzinger, A Heydon, M Mitzenmacher, and M Najork. Measuring search engine quality using random walks on the Web. In *Proc. 8th International World Wide Web Conference*, pages 213–225, 1999.
- [18] M. Hersovici, M. Jacovi, Y. S. Maarek, D. Pelleg, M. Shtalhim, and Sigalit Ur. The shark-search algorithm — An application: Tailored Web site mapping. In *Proc. 7th Intl. World-Wide Web Conference*, 1998.
- [19] BJ Jansen, A Spink, and T Saracevic. Real life, real users and real needs: A study and analysis of users queries on the Web. *Information Processing and Management*, 36(2):207–227, 2000.

- [20] J Kleinberg. Authoritative sources in a hyperlinked environment. *Journal of the ACM*, 46(5):604–632, 1999.
- [21] F Menczer. ARACHNID: Adaptive Retrieval Agents Choosing Heuristic Neighborhoods for Information Discovery. In *Proc. 14th International Conference on Machine Learning*, pages 227–235, 1997.
- [22] F Menczer. Complementing search engines with online Web mining agents. *Decision Support Systems*, 2002. Forthcoming.
- [23] F Menczer. Lexical and semantic clustering by web links. *IEEE Trans. on Knowledge and Data Engineering*, Submitted, 2002. Shorter version available as Computing Research Repository (CoRR) Technical Report arXiv.org:cs.IR/0108004.
- [24] F Menczer and RK Belew. Adaptive information agents in distributed textual environments. In *Proc. 2nd International Conference on Autonomous Agents*, pages 157–164, Minneapolis, MN, 1998.
- [25] F Menczer and RK Belew. Adaptive retrieval agents: Internalizing local context and scaling up to the Web. *Machine Learning*, 39(2–3):203–242, 2000.
- [26] F Menczer, G Pant, M Ruiz, and P Srinivasan. Evaluating topic-driven Web crawlers. In *Proc. 24th Annual Intl. ACM SIGIR Conf. on Research and Development in Information Retrieval*, 2001.
- [27] F Menczer, G Pant, and P Srinivasan. Topic-driven crawlers: Machine learning issues. *ACM TOIT*, Submitted, 2002. <http://dollar.biz.uiowa.edu/~fil/Papers/TOIT.pdf>.
- [28] M Mitra, A Singhal, and C Buckley. Improving automatic query expansion. In *Proc. 21st ACM SIGIR Conf. on Research and Development in Information Retrieval*, pages 206–214, 1998.
- [29] M. Najork and J. L. Wiener. Breadth-first search crawling yields high-quality pages. In *Proc. 10th International World Wide Web Conference*, 2001.
- [30] MJ Nelson. The effect of query characteristics on retrieval results in the TREC retrieval tests. In *Proc. Annual Conference of the Canadian Association for Information Science*, 1995.
- [31] G Pant and F Menczer. MySpiders: Evolve your own intelligent Web crawlers. *Autonomous Agents and Multi-Agent Systems*, 5(2):221–229, 2002.
- [32] G Pant, P Srinivasan, and F Menczer. Exploration versus exploitation in topic driven crawlers. In *Proc. WWW-02 Workshop on Web Dynamics*, 2002.
- [33] B Pinkerton. Finding what people want: Experiences with the WebCrawler. In *Proc. 1st International World Wide Web Conference*, 1994.
- [34] M Porter. An algorithm for suffix stripping. *Program*, 14(3):130–137, 1980.

- [35] Jason Rennie and Andrew Kachites McCallum. Using reinforcement learning to spider the Web efficiently. In *Proc. 16th International Conf. on Machine Learning*, pages 335–343. Morgan Kaufmann, San Francisco, CA, 1999.
- [36] T Saracevic and P Kantor. A study of information seeking and retrieving. II. Users, questions, and effectiveness. *Journal of the American Society for Information Science*, 39(3):177–196, 1998.
- [37] IR Silva, B Ribeiro-Neto, P Calado, N Ziviani, and ES Moura. Link-based and content-based evidential information in a belief network model. In *Proceedings of the 23rd International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 96–103, 2000.
- [38] A Spink, D Wolfram, BJ Jansen, and T Saracevic. Searching the Web: The public and their queries. *Journal of the American Society for Information Science*, 52(3):226–234, 2001.
- [39] P Srinivasan, J Mitchell, O Bodenreider, G Pant, and F Menczer. Web crawling agents for retrieving biomedical information. In *Proc. Int. Workshop on Agents in Bioinformatics (NETTAB-02)*, 2002.
- [40] CJ van Rijsbergen. *Information Retrieval*. Butterworths, London, 1979. Second edition.