Proceedings of the International Workshop on Bioinformatics and Multi-A
Bologna, Italy, July 15, 2002.

Web Crawling Agents for Retrieving Biomedical Information

Padmini Srinivasan[acd]
padmini-srinivasan@uiowa.edu

Joyce Mitchell[ab]
mitchell@nlm.nih.gov

Olivier Bodenreider[a]
olivier@nlm.nih.gov

Gautam Pant[c]
gautam-pant@uiowa.edu

Filippo Menczer[c]
filippo-menczer@uiowa.edu

[a]National Library of Medicine
8600 Rockville Pike
Bethesda, MD 20894

[b]Health Mgmt. & Informatics
University of Missouri
Columbia, MO 65211

[c]Management Sciences
The University of Iowa
Iowa City, IA 52242

[d]Library & Info. Science
The University of Iowa
Iowa City, IA 52242

ABSTRACT

Autonomous agents for topic driven retrieval of information from the Web are currently a very active area of research. The ability to conduct real time searches for information is important for many users including biomedical scientists, health care professionals and the general public. We present preliminary research on different retrieval agents tested on their ability to retrieve biomedical information, whose relevance is assessed using both genetic and ontological expertise. In particular, the agents are judged on their performance in fetching information about diseases when given information about genes. We discuss several key insights into the particular challenges of agent based retrieval learned from our initial experience in the biomedical domain.

1. INTRODUCTION

Autonomous agents represent an emerging area of research. Agent based technologies are being applied to a wide range of complex problems from interface agents [14, 19] to recommender systems [3] and autonomous and comparative shopping agents [10, 17, 24]. Our interest is in the design of retrieval agents that seek out relevant Web pages in response to user generated topics [11, 22, 23].

Due to limited bandwidth, storage, and computational resources, and to the dynamic nature of the Web, search engines cannot index every Web page, and even the covered portion of the Web cannot be monitored continuously for changes. In fact a recent estimate of the visible Web is at around 7 billion "static" pages as of March 2002 [7]. This estimate is more than triple the 2 billion pages that the largest search engine, Google, reports at its Web site [12]. Therefore it is essential to develop effective agents able to conduct real time searches for users. This goal is reflected in our previous research in which we have explored a variety of Web crawling agents that operate using both lexical and link-based criteria [23]. We have assessed their performance with topics derived from the Yahoo and Open Directory (DMOZ) hierarchies. We used several alternative measures and have also compared those that are dominantly exploratory in nature with those that are more exploitative of the available evidence [28]. In ongoing research we are studying the scalability of various crawling agents and their sensitivity to different topic characteristics.

The particular goal in this paper is to examine the applicability of our agents to the challenge of retrieving biomedical information from the Web. The motivating question here is: How does one locate information from the Web that is related to a gene? This question is important not only from the aspect of scientific discovery, but also from the viewpoint of the general public. With the development of DNA microarrays accelerating the study of gene expression patterns, and progress in areas such as proteomics, the biomedical scientist is increasingly challenged by the growing body of relevant literature. This is particularly so, when considering the unexpected connections that become important across seemingly disjoint specializations. Thus when investigating a new gene or a new gene product, an integral part of the investigation is to identify what else is already known about it. Web retrieval agents, explored in this paper, are intended as a part of the overall solution to this problem. The goal is to scour the Web upon demand looking for relevant material, which might be located in sites far removed from well recognized resources such as those developed by the Human Genome Project (HGP) [16].

Such agents are also important from the consumers' point of view. A growing segment of the population is increasingly taking charge of their own health and therefore demanding more information about health problems and their underlying genetic basis. With the potential of genetic testing and engineering offering a greater range of individual choice

consumers are seeking to get better educated about the link between diseases and genes. In fact, many consumers are reacting to the press announcements of the completion of the Human Genome Project by asking, "What information is available from the Human Genome Project about my disease?" Unfortunately the path toward relevant information for the lay person is very thorny. In our previous study [26] we show four major challenges encountered when navigating from phenotype (disease) to genotype (gene(s)). These include the sheer complexity of the data, the accelerated rate of discovery in the biomedical domain, the diversity of knowledge resources and databases and finally the often idiosyncratic representation styles of the different databases.

Consider for instance the complexity of data, in particular the names of genes and gene products. The number of synonyms and the non-intuitive nature of the synonyms for various diseases, genes, and gene symbols make it difficult to find comprehensive information. The nomenclature committee of the Human Genome Organization (HUGO) decides upon an official gene name and gene symbol and makes this information available on-line. These names often include metadata that link to diseases, and give inheritance patterns or other information. For example, *fibrillin-1 (Marfan Syndrome)* is the official name for a gene that when mutated causes four diseases, even though the official gene name only mentions one of these. The official name for a gene that when mutated causes one type of polycystic kidney disease is *polycystic kidney disease 1 (autosomal dominant)*.

The diversity of data/knowledge base systems regarding the human genome also makes traversal of these systems difficult for novices. Most systems focused on consumers or non-specialty clinicians such as WebMD [31], MayoClinic.com [20], MedicineNet.com [21], and MEDLINE*plus* [25] do not refer directly to the scientific databases such as those produced by the Human Genome Project. And most of the HGP databases do not refer to consumer oriented pages. Furthermore, many of the genome databases have multiple species represented within them as the connection to human diseases is not the primary emphasis.

MEDLINE*plus* [25], the principal resource of the National Library of Medicine (NLM) focused on consumers, includes many single gene diseases as main topics or subtopics. MEDLINE*plus* is a curated resource providing a set of links to documents selected by health sciences librarians for consumers. As such, the linked documents are vetted for readability, currency and source. However, because of the lack of perceived readability for consumers, the connection to the gene information from the disease information is often not made. And yet a search of the HGP-related system of OMIM [13] and LocusLink [32] reveals over 1300 genes and 1700 diseases where a causal relationship has been definitively established. Other databases provide more details such as a catalog specific mutations and correlations with the disease severity [18]. These rich stores of information are largely hidden not only from the lay public but also from general health care providers.

Set within this context, our aim is to assess the ability of Web retrieval agents to find relevant biomedical information. In particular, given a specific gene the goal for our agents is to find information about the associated disease(s). The larger goal is to use Web agents to bridge the gulf between the various Web resources regarding genes and diseases.

This study is a natural merger of two strands of research that we have conducted. First, as a follow-up to our earlier study of manually connecting from phenotype (disease) to genotype [26], we decided in this study to attempt a navigation in the other direction (i.e., from genotype to phenotype) and to use intelligent agents in the search. Second, the emphasis on agents derives from our previous research in which we explore a wide variety of crawling agents with special emphasis on their underlying search algorithms and methods for evaluating their performance [23, 28].

In the next sections we present our Web crawling agents and describe our experimental design, including details about the dataset and our evaluation methods. Then we outline our experimental results and discuss a number of lessons learned in this project.

2. AGENTS FOR WEB INFORMATION RETRIEVAL

Agents for topic driven searching (also known as topic driven crawlers and focused crawlers) respond to the particular information needs expressed by topical queries or interest profiles. These could be the needs of an individual user or those of a community with shared interests. They support decentralizing the retrieval process, which is a more scalable approach when compared with centralized multipurpose search engines. An additional benefit is that such agents can be driven by a rich context (topics, queries, user profiles) with which to interpret pages and select the links to be visited.

Starting with the early breadth first [29] and depth first [8] crawling agents defining the beginnings of this research, we now see a variety of crawling algorithms. There is Shark Search [15], a more focused variant of Fish Search [8]. There are crawling agents whose decisions rely heavily on link based criteria [6, 9, 4]. Diligenti *et al.*, for example, use backlink-based context graphs to estimate the likelihood of a page leading to a relevant page, even if it is not relevant itself [9]. Others exploit lexical and conceptual knowledge. For example Chakrabarti *et al.* [5] use a hierarchical topic classifier to select links for crawling. Still others emphasize contextual knowledge [1, 22, 27] for the topic including that received via relevance feedback.

We are at present part of a highly creative phase regarding the design of topic driven retrieval agents. Almost all of this research is characterized by an emphasis on general topic queries, such as those derived from the Yahoo and DMOZ directories or those pertaining to topics such as 'bicycling' and 'gardening.' Our goal is to take what we now understand from these general domains and provide an important extension of this research into the specific context of biomedical information.

2.1 Architecture

As implemented, our agents share data structures and utilities to optimize efficiency without affecting fairness during evaluation. Examples of common facilities include a cache, an HTTP interface for the Web, a simple HTML parser, a stemmer [30], benchmarking and reporting routines. Our agent implementations are in Perl.

Each agent can visit up to `MAX_PAGES` = 1000 pages per topic, starting from a seed set. We use a timeout of 10 seconds for Web downloads. Large pages are chopped so that we retrieve only the first 100 KB. The only protocol

allowed is HTTP (with redirection allowed), and we also filter out all but pages with `text/html` content. Stale links yielding HTTP error codes are removed as they are found (only good links are used in the analysis).

We limit the memory available to each agent by constraining its buffer size. This buffer can be used to temporarily store links, typically a frontier of pages whose links have not been explored. Each agent is allowed to track a maximum of `MAX_BUFFER` = 256 links. If the buffer becomes full then the agent must decide which links are to be substituted as new ones are added.

We employ three crawling agents in this study. The first two are single-agent algorithms based on particular variations of Best-First traversals, called BFSN where $N = 1$ and $N = 256$. The former is the more commonly studied algorithm [6, 15], the latter is a more explorative agent. BFS256 has produced some of the best results in our previous research [28]. The third crawling agent, called InfoSpiders, is also one that we have developed in previous research [22]. InfoSpiders is a multi-agent approach based on an evolving population of learning agents.

2.2 Best-First

The basic idea in Best-First crawlers [6, 15] is that 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. After completing the crawl of N pages the crawler decides on the next batch of N and so on. Typically an initial representation of the topic, in our case a set of keywords, is used to guide the crawl. More specifically this is done in the link selection process by computing the lexical similarity between a topic's keywords and the source page for the link. Thus the similarity between a page p and the topic is used to estimate the relevance of the pages pointed by p. The N URLs with the best estimates are then selected for crawling. Cosine similarity is used by the crawlers and the links with minimum similarity score are removed from the frontier if necessary in order not to exceed the buffer size `MAX_BUFFER`. Figure 1 offers a simplified pseudocode of a BFSN crawler. The `sim()` function returns the cosine similarity between topic and page:

$$sim(q,p) = \frac{\sum_{k \in q \cap p} w_{kq} w_{kp}}{\sqrt{\sum_{k \in p} w_{kp}^2 \sum_{k \in q} w_{kq}^2}} \quad (1)$$

where q is the topic, p is the fetched page, and w_{kd} is the frequency of term k in d.

2.3 InfoSpiders

In InfoSpiders [22], an adaptive population of agents search for pages relevant to the topic, using evolving query vectors and neural nets to decide which links to follow. This evolutionary approach uses a fitness measure based on similarity as a *local* selection criterion. The original algorithm was previously simplified and implemented as a crawler module [23]. Here we use a variant schematically illustrated in Figure 2.

There are at most `MAX_POP_SIZE` agents, maintaining a distributed frontier whose total size does not exceed `MAX_BUFFER`. InfoSpiders agents are independent from each other and crawl in parallel.

The adaptive representation of each agent consists of a vector of keywords (initialized with the topic keywords) and

```
BFS_N (topic, starting_urls, N) {
   foreach link (starting_urls) {
      enqueue(frontier, link);
   }
   while (visited < MAX_PAGES) {
      links_to_crawl := dequeue_top_links(frontier, N);
      foreach link (randomize(links_to_crawl)) {
         doc := fetch(link);
         score :=  sim(topic, doc);
         merge(frontier, extract_links(doc), score);
         if (#frontier > MAX_BUFFER) {
             dequeue_bottom_links(frontier);
         }
     }
   }
}
```

Figure 1: Pseudocode of BFSN crawling agents.

```
IS (topic, starting_urls) {
   foreach agent (1 .. MAX_POP_SIZE) {
      initialize(agent, topic);
      agent.frontier := random_subset(starting_urls);
      agent.energy := THETA / 2;
   }
   while (visited < MAX_PAGES) {
      foreach agent (1 .. pop_size) {
         link := pick_and_remove(agent.frontier);
         doc := fetch(link);
         newenergy = sim(agent.topic, doc);
         agent.energy += newenergy - COST;
         learn_to_predict(agent.nnet, newenergy);
         merge(agent.frontier, extract_links(doc), agent.nnet);
         if (#agent.frontier > MAX_BUFFER/MAX_POP_SIZE) {
             dequeue_bottom_links(agent.frontier);
         }
         delta := newenergy - sim(topic, agent.doc);
         if (boltzmann(delta)) {
            agent.doc := doc;
         }
         if (agent.energy < 0) kill(agent);
         if (agent.energy > THETA and pop_size < MAX_POP_SIZE) {
            offspring := mutate(clone(agent));
         }
      }
   }
}
```

Figure 2: Pseudocode of InfoSpiders crawling agents. The parameters are set as follows: `MAX_POP_SIZE`=8, `THETA`=0.1, `COST`=0.0125.

a neural net used to evaluate new links. Each input unit of the neural net receives a count of the frequency with which the keyword occurs in the vicinity of each link to be traversed, weighted to give more importance to keywords occurring near the link. The neural net computes link estimates, and based on these the agent uses a stochastic selector to pick one of the links.

After a new page has been fetched, the agent receives energy in proportion to the similarity between its query vector and this page (Equation 1). The constant `COST` charged for each fetched page is such that an agent will die after visiting a maximum of four pages that yield no energy. The agent's neural net is also trained to improve the link estimates by predicting the similarity of the new page, given the page that contained the link leading to it.

An agent moves to the newly selected page only if the `boltzmann()` function returns a true condition. This is de-

termined stochastically based on the probability

$$\Pr(\delta) = \frac{1}{1 + e^{-\delta/T}} \quad (2)$$

where δ is the difference between the similarity of the new and current page to the agent's keyword vector and $T = 0.1$ is a temperature parameter.

An agent's energy level is used to determine whether the agent should die, reproduce, or survive. An agent dies when it runs out of energy and reproduces when the energy level passes the constant threshold `THETA`. At reproduction, offspring receive part of the parent's energy and link frontier. Offspring keyword vectors are also mutated by adding the term that is most frequent in the parent's current document. Such a mutation provides InfoSpiders with the unique capability to adapt the search strategy based on new clues captured from promising pages.

3. EXPERIMENTAL DESIGN

3.1 Topics and Seed Sets

Crawler agents need starting points, i.e., locations on the Web from which to start the crawl. These pages are referred to as *seeds.* To obtain some reasonable seeds for the crawl, our strategy is to utilize the top 5 URLs returned by a manual search on Google. All of our agents start from the same set of seeds. In selecting our seeds we remove any duplicates as well as any pointers to pages such as PDF files.

Since our goal is to find descriptions of diseases when given genes, we first defined a pool of 75 unique genes. Each of these 75 genes are known to cause at least one disease. In several instances a disease may be caused by more than one of our group of genes. We then selected a set of 32 diseases that are caused by these 75 genes. The connections between the diseases and genes were established using NLM's LocusLink resource [32]. It should be noted that although these 75 genes cause many more diseases, our study is restricted to these 32 diseases. MEDLINE*plus* has the diseases listed as either main topic pages or as subtopics on a broader topic page. Examples of diseases where the entire page is devoted to the disease include Marfan Syndrome, Phenylketonuria and Huntington Disease. Some diseases were subtopics within a MEDLINE*plus* page representing a family of diseases. For example, Duchenne, Becker, Limb-Girdle and Myotonic Dystrophy are subtopics within the main page of Muscular Dystrophy.

For the genes linked to these 32 diseases, data from LocusLink such as the official gene name, official symbol, alias symbols, gene products, preferred products, alias protein names, and phenotype (i.e., disease or syndrome) were manually extracted. A set of keywords was constructed for each gene from all of this information except the phenotype. Because the official gene name frequently includes a disease name, the keywords did not include the official gene name. These keywords were used to form the initial search query that was submitted to Google. These keywords were also used to guide the crawls as described in the previous section. However, since Google by default uses the conjunction of the search terms, often zero hits were obtained. In these cases we decided to limit the search to only the gene name (with disease names removed) plus symbols and aliases. In some cases fewer than five URLs were retrieved. We then eliminated one keyword at a time until a minimum of 5 usable URLs were obtained. These URLs were used as seeds by our crawler agents.

Topic

`Official Gene Name: fibrillin-1 (Marfan Syndrome)`
`Phenotype: Marfan Syndrome`
`Keywords: FBN1; FBN; MFS1; fibrillin 1; Fibrillin-1`

Seed URLs

`http://gdb.jst.go.jp/HOWDY/search.pl?Cls=LocusLink&Val=2200`
`http://cedar.genetics.soton.ac.uk/cgi-bin/runsearch22np?FBN1::15`
`http://www.gene.ucl.ac.uk/pub/nomen/month-up/2001/Nov01HGNC.txt`
`http://cardio.bjmu.edu.cn/List.asp?chr=15`
`http://www.pubgene.uio.no/cgi/tools/Network/Browser.cgi?gene=FBN1`

Description
`What is Marfan syndrome? The Marfan syndrome is a heritable disorder of the connective tissue that affects many organ systems, including the skeleton, lungs, eyes, heart and blood vessels. The condition affects both men and women of any race or ethnic group. It is estimated that at least 200,000 people in the United States have the Marfan syndrome or a related connective tissue disorder...`
Target URLs

`http://www.americanheart.org/presenter.jhtml?identifier=4672`
`http://www.marfan.org/index.html`
`http://www.marfan.org/list/chapters.html`
`http://www.marfan.org/pub/cardiac.htm`
`http://www.marfan.org/pub/emergency.html`
`http://www.marfan.org/pub/factsheet.html`
`http://www.marfan.org/pub/newsletter/features.html`
`http://www.marfan.org/pub/orthopedic.htm`
`http://www.marfan.org/pub/physed.html`
`http://www.marfan.org/pub/resourcebook/diagnosing.html`
`http://www.marfan.org/pub/resourcebook/eye.html`
`http://www.marfan.org/pub/teenagers.html`
`http://www.mayoclinic.com/invoke.cfm?id=HQ01056`
`http://www.modimes.org/HealthLibrary/334_604.htm`
`http://www.ncbi.nlm.nih.gov/disease/Marfan.html`
`http://www.niams.nih.gov/`
`http://www.niams.nih.gov/hi/topics/marfan/marfan.htm`

Table 1: A sample topic's seeds, description, and targets (abbreviated).

3.2 Page Importance and Target Sets

Retrieval agents are generally evaluated on the relevance of the retrieved pages. Since it is difficult to obtain relevance judgments from humans because of the scale of the problem, we need some other automated mechanism to estimate the relevance of a page. A common approach for this is to assess the similarity between the retrieved page and a gold standard that represents a relevant page. For instance in [6] the authors explore a rather simple version of this 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 [5]. In our own research we have used as the gold standard the concatenation of text passages describing relevant sites written by Yahoo and DMOZ editors [23, 28].

For this biomedical application the geneticist in our group

extracted a description of each target disease from the Web. Each description was limited to about a page in length. Some were as short as half a page. These descriptions most often came from the Medline*plus* Web site and for the remainder were a combination of texts extracted from different Web sites. In general, each description provides an overview of the disease as well as a summary of the major symptoms associated with it. An abbreviated example is provided in Table 1. Thus one measure that we use to gauge the importance of a page is the lexical similarity between the page and the corresponding disease description.

We also use a second source of evidence for relevance. We take the MEDLINE*plus* page for the disease as the source of "good" pages. In other words all pages that are pointed to by this source are viewed as relevant. This set is shown in Figure 1 under 'Target URLs.' The advantage with this approach is that since MEDLINE*plus* is a curated resource we are very confident in the value of its outlinks. The disadvantage is that we are then limited to the constraints that come implicitly with a single curated resource. Thus the underlying policy determines which pages are linked to a disease page. For example, the disease description for the Ellis-van Creveld syndrome was very brief as linked directly from MEDLINE*plus*. Several more extensive disease descriptions exist at sites that are usually avoided by MEDLINE*plus* policy, such as those hosted by educational institutions or those that do not display update dates.

We believe that the combination of our two methods for assessing page importance provides a basis for a reasonably comprehensive evaluation strategy. Our assessment method is based on the idea that a good agent will be able to find important pages, where these match the pages pointed to by MEDLINE*plus* or where they match the gold standard description provided by our expert. In order to make the assessment meaningful, we block the source MEDLINE*plus* page itself from the agent. In other words, the agent will have to find the relevant pages by other means.

3.3 Evaluation Metrics

Given the above two mechanisms for assessing page importance, we need to summarize in some reasonable way the performance of the agents. 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 or precision is complicated by the fact that crawlers have 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 agents that return speedy dividends. For an agent 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.

In response to these differing needs we adopt two approaches for summarization: static and dynamic. The static approach examines retrieval quality based upon the full set of (up to 1000) pages retrieved during the lifespan of an agent. That is, a static measure is calculated at the end of the agent's crawl. In contrast the dynamic measures provide a temporal characterization of the agent's strategy, by considering the pages fetched while the crawl is in progress. Dynamic plots offer a trajectory over time that displays the dynamic behavior of the crawl. Thus the following measures are used in this study.

Dynamic Similarity: Each retrieved page is compared with the description page created by the expert. We measure the average cosine similarity between the TFIDF vector of this description page and the TFIDF vector of each page visited up to a certain point in the crawl:

$$sim(q, S(t)) = \frac{1}{|S(t)|} \sum_{p \in S(t)} sim(q, p) \qquad (3)$$

where q is the topic and $sim()$ is the cosine similarity function of Equation 1, except that term frequencies w_{kd} are replaced by TFIDF weights:

$$w_{kd}^{tfidf} = w_{kd} \cdot \left(1 + \ln\left(\frac{|S|}{n_k}\right)\right) \qquad (4)$$

where n_k is the number of documents in which term k occurs and S is the set of all pages crawled for a particular topic. We do this incrementally as the set of pages $S(t)$ for each crawler grows with time t. This way we can plot a trajectory over time and assess the crawlers based on their capability to remain on topic. In general, this measure assesses the cohesiveness of the retrieved set with the topic as the core. The underlying assumption is that the more cohesive the crawled set, the more relevant its pages.

Dynamic Target Recall: Given a set of target pages for a topic, this measure simply looks at the percentage of the target set that is crawled by the agent, as a function of the number of pages crawled up to time t.

Static Similarity: Here we first pool all the pages retrieved by the different agents for a given topic. We then compute similarity as shown above between each page and the description page. Next we identify the top N most similar pages. Given this 'relevant' pool of N pages, the measure assesses the percentage of this pool that was crawled by each agent. We do this analysis for different values of N.

4. PRELIMINARY RESULTS

4.1 Crawl Runs

Out of the 75 initial topics (one for each gene), there were a total of 58 topics that we were able to use for our similarity analysis. Most of the eliminations were because either the crawls did not complete with integrity (due to hardware problems) or because the seeds obtained from Google were not appropriate. Of these 58 topics, only 23 had target sets provided by MEDLINE*plus* that allowed for recall analysis.

As mentioned before each retrieved page is compared with the description page created by the expert. For the dynamic perspective we measure the average similarity between this description page and the pages visited up to a certain point in the crawl. Figure 3 shows these results as the set of pages retrieved grows incrementally. The performance was not statistically different across the three crawling agents. We have not plotted error bars in order to keep the graphs visually clear. BFS1 shows a slight edge over the others in the early part of the crawl, but its exploitative behavior does not appear advantageous over the longer run.

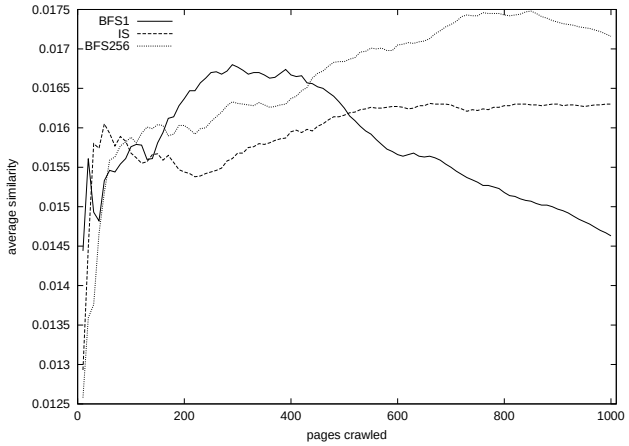

Figure 3: Dynamic performance. The plot is based on 58 topics. The differences in average similarity between the three crawlers are not statistically significant.

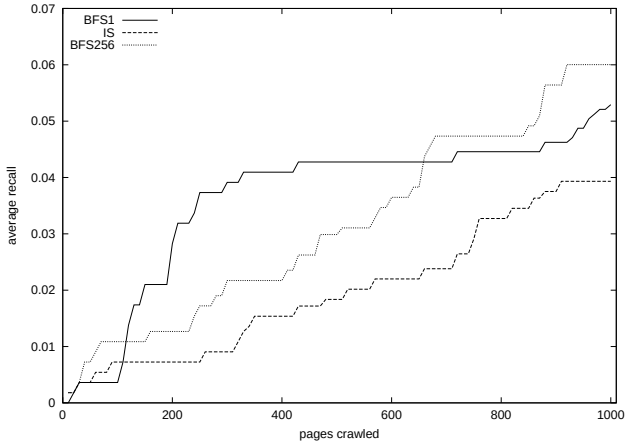

Figure 4: Dynamic recall of target pages. The plot is based on 23 topics. The differences in average recall performance between the three crawlers are not statistically significant.

Figure 4 shows the recall of target pages identified for each topic. Unfortunately, the recall levels reached are quite low for all three agents. We propose some explanations in the discussion section below. Differences in performance are not statistically significant between crawling agents.

Finally Figure 5 shows the results using the static measure of similarity. Remember that we first pool all the pages retrieved by the different agents for a given topic. We then identify the N pages most similar to the target disease description and measure the coverage of this 'relevant' pool by each agent, for various values of N. In this analysis differences in performance appear to be statistically significant. InfoSpiders achieves the best coverage, followed by BFS256. Again BFS1 appears to be too greedy. The trend across N indicates that the InfoSpiders agents are particularly good at pinpointing the few highest-quality pages.

4.2 Discussion

These preliniary results offer many valuable lessons that may hopefully be used effectively in our future efforts. First, it is clear from the low recall levels that the problem of starting our crawl agents from genes and looking for disease information is a hard problem. Based upon our own experience this domain has proved much more challenging than the Yahoo and DMOZ based topics that we have thus far considered. In these previous efforts recall levels, for the target pages, of 0.35 to 0.4 on average have been achieved.

A key characteristic of the present problem domain is the cryptic nature of the gene names, gene product names and gene symbols. These are frequently acronyms that are likely to duplicate other words or acronyms in common usage. Even within the medical language these are often confused with other concepts. An example topic in which this was observed is the BACH1 gene that causes breast cancer. BACH1 is the symbol for the gene product BRCA1-associated C-terminal helicase. As mentioned before, following the algorithm for gathering seed URLs, the symbols and synonyms were submitted to Google until a subset of them resulted in at least five URLs. In this case, the subset used was the BACH1 symbol. Unfortunately, the Google search matched this gene symbol with the composer Bach and the Web crawlers delved into the music world instead of the breast cancer world. In fact in looking back at the data we noticed that for eight topics not a single seed was even remotely pertinent to the topic of the gene! These were some of the topics that were eliminated from our initial pool of 75 genes when doing the analysis. Even with the 58 topics analyzed, we found that 21 had at least 1 seed that was not appropriate for the topic. Such seeds are likely to misdirect the agents. We now appreciate the risks involved when obtaining seeds from a search engine.

Other problems that are now recognized include seeds whose pages no longer exist as well as seeds whose pages have no outlinks. This latter aspect was unexpected since in our earlier studies the methodology used to identify seeds ensured the existence of outlinks [23, 28].

A major aspect that we recognize retrospectively, is that a good portion of the pages retrieved, and even a few of the seed pages were non English pages. Since our target pages were limited to the English language, this causes a very significant degree of mismatch. Language recognition is therefore a necessary component of such an agent. Even within English pages, general purpose stemming rules such as those employed by our agents [30] ought to be adapted to the specialized terminology of the biomedical domain.

Another observed problem is that often the agent becomes bogged down in a particular site and does not emerge from it, at least within its given lifespan of 1000 pages. For instance a seed for the ACE gene (Angiotensin I Converting Enzyme) leads to a GDB-mediated (Genome Database) and pre-computed LocusLink search. The returned page is essentially a recapitulation of the LocusLink information of which only a very small portion of the output data pertains to the disease(s). The remaining links point to DNA and protein sequence data. Our intuition is that this situation happens rather frequently in the biomedical domain. Consequently we may need to augment our agents with a strategy to recognize when the amount of exploration within a site has reached a maximum limit.

Finally, the above challenges are compounded by the fact

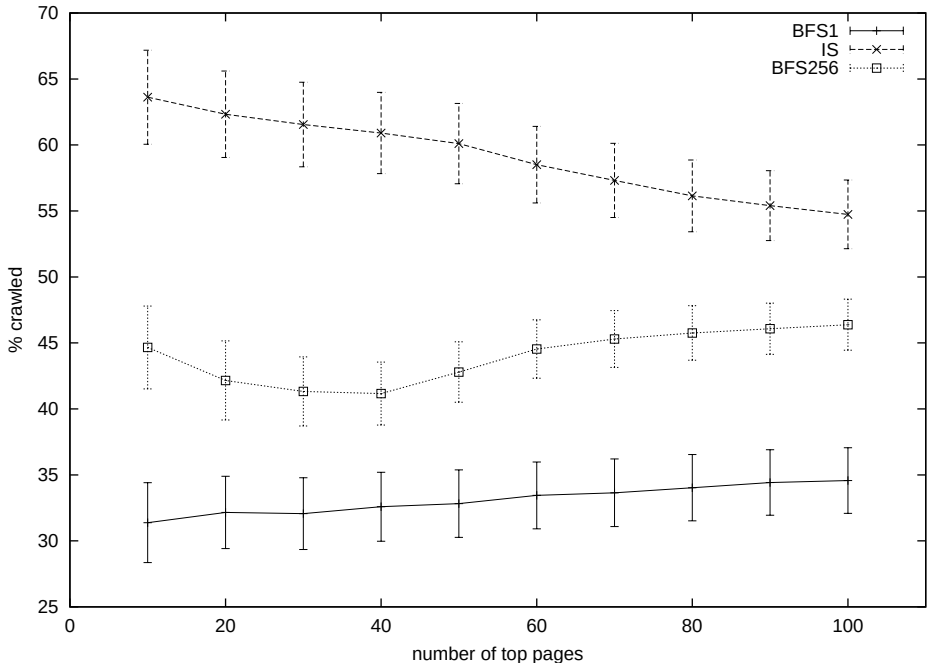

Figure 5: Static performance. Error bars correspond to ± 1 standard error (standard deviation of mean coverage) across 58 topics.

that we deliberately did not give the agents any clues about the fact that we were looking for diseases. To remind the reader, we did not add the phenotype information to the keywords and also eliminated gene names that included disease names. This was done in order to keep the evaluation as unbiased as possible. At the same time our relevance criteria were built upon the disease descriptions and disease-based URLs. Perhaps a more realistic strategy is an intermediate one, where the agent is given some minimal guidelines on how to recognize a disease page. This could possibly be composed of generic information about how to recognize a page containing disease descriptions. In retrospect we believe that our decision to avoid any disease clues in the keywords may have been too extreme.

5. CONCLUSIONS

To the best of our knowledge this has been the first formal study of an agent crawling for Web pages containing biomedical information[1] and with genes defining the starting points. Our intent was to capitalize on our experience with Web crawling agents, taking some of the best agents and applying them to a problem in biomedicine. The goal of seeking out disease information for a given gene in real time is of growing importance to scientists, health care professionals and the general public.

Among the crawling agents that we applied to this biomedical information retrieval problem, InfoSpiders was the one that displayed the best performance in the static similarity metric — the only measure for which a significant difference in performance could be observed. We conclude that adaptability at the population and individual level may be key features in designing crawling agents to cope with complex search domains such as biomedical information.

Based on the present research we now recognize several characteristics that make agent-based retrieval of biomedical information distinct from retrieval from the more general domains. Our plan is to further explore these issues in follow up research and thereby continue our contribution to research on agents for tracking biomedical information.

[1]We exclude studies of the deep Web in this statement, i.e., studies of mechanisms for searching databases with Web interfaces, such as LocusLink.

6. ACKNOWLEDGMENTS

This work was funded in part by the National Science Foundation under CAREER grant No. IIS-0133124 to FM.

7. REFERENCES

[1] C. Aggarwal, F. Al-Garawi, and P. Yu. Intelligent crawling on the World Wide Web with arbitrary predicates. In *Proc. 10th Intl. 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 *Proceedings of the 23rd International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 296–303, 2000.

[3] M. Balabanović and Y. Shoham. Content-based, collaborative recommendation. *Communications of the ACM*, 40(3):67–72, 1997.

[4] K. Bharat and M. Henzinger. Improved algorithms for topic distillation in hyperlinked environments. In *Proceedings of the 21st International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 104–111, 1998.

[5] 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.

[6] J. Cho, H. Garcia-Molina, and L. Page. Efficient crawling through URL ordering. In *Proceedings of the Seventh International World Wide Web Conference, Brisbane, Australia*, 1998.

[7] Cyveillance. Sizing the internet. White paper, July 2000. http://www.cyveillance.com.

[8] P. De Bra and R. Post. Information retrieval in the World Wide Web: Making client-based searching feasible. In *Proceedings of the First International World Wide Web Conference*, 1994.

[9] 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.

[10] R. Doorenbos, O. Etzioni, and D. Weld. A scalable comparison-shopping agent for the World-Wide Web. In *Proceedings of the First International Conference on Autonomous Agents*, pages 39–48, 1997.

[11] D. Eichmann. The RBSE spider — Balancing effective search against Web load. *Computer Networks*, 4(2):281–288, 1994.

[12] Google. http://www.google.com.

[13] A. Hamosh, A. Scott, J. Amberger, et al. Online Mendelian Inheritance in Man (OMIM), a knowledgebase of human genes and genetic disorders. *Nucleic Acids Research*, 30(1):52–55, 2002.

[14] T. Helmy, S. Amamiya, and M. Amamiya. User's ontology-based autonomous interface agents. In *The 2001 International Conference on Intelligent Agent Technologies*, Maebashi City, 2001.

[15] M. Hersovici, M. Jacovi, Y. S. Maarek, D. Pelleg, M. Shtalhaim, and S. Ur. The shark-search algorithm — An application: Tailored Web site mapping. In *Proc. 7th International World-Wide Web Conference*, 1998.

[16] Human Genome Project. The human genome. *Nature*, 409(6822):813–958, 2001.

[17] J. O. Kephart and A. R. Greenwald. Shopbot economics. *Autonomous Agents and Multi-Agent Systems*. forthcoming.

[18] M. Krawczak, E. V. Ball, I. Fenton, et al. Human gene mutation database — A biomedical information and research resource. *Hum Mutat*, 15(1):45–51, 2000.

[19] H. Lieberman. Autonomous interface agents. In *Proc. ACM Conference on Computers and Human Interface*, Atlanta, GA, 1997.

[20] MayoClinic. http://www.mayoclinic.com.

[21] MedicineNet. http://www.medicinenet.com.

[22] F. Menczer and R. Belew. Adaptive retrieval agents: Internalizing local context and scaling up to the Web. *Machine Learning*, 39(2–3):203–242, 2000.

[23] F. Menczer, G. Pant, M. Ruiz, and P. Srinivasan. Evaluating topic-driven Web crawlers. In *Proc. 24th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, 2001.

[24] F. Menczer, W. Street, N. Vishwakarma, A. Monge, and M. Jakobsson. IntelliShopper: A proactive, personal, private shopping assistant. In *Proc. 1st ACM Int. Joint Conf. on Autonomous Agents and MultiAgent Systems (AAMAS 2002)*, 2002.

[25] N. Miller, E. Lacroix, and J. E. B. Backus. MEDLINE*plus*: Building and maintaining the National Library of Medicine's consumer health Web service. *Bull. Med. Libr. Assoc.*, 88(1):11–27, 2000.

[26] J. Mitchell, A. T. McCray, and O. Bodenreider. From phenotype to genotype: Navigating the available information resources. In *Proceedings of the AMIA Symposium*, 2002.

[27] M. Najork and J. L. Wiener. Breadth-first search crawling yields high-quality pages. In *Proc. 10th International World Wide Web Conference*, 2001.

[28] G. Pant, P. Srinivasan, and F. Menczer. Exploration versus exploitation in topic driven crawlers. In *Proc. Second International Workshop on Web Dynamics*, 2002.

[29] B. Pinkerton. Finding what people want: Experiences with the WebCrawler. In *Proceedings of the First International World Wide Web Conference, Geneva, Switzerland*, 1994.

[30] M. Porter. An algorithm for suffix stripping. *Program*, 14(3):130–137, 1980.

[31] WebMD. http://www.webmd.com.

[32] D. Wheeler, D. Church, A. Lash, et al. Database resources of the National Center for Biotechnology Information: 2002 update. *Nucleic Acids Res*, 30(1):13–16, 2002.