Metadata Based Web Mining for Relevance

Jeonghee Yi
Computer Science, UCLA
405 Hilgard Av.
LA, CA 90095
jeonghee@cs.ucla.edu

Neel Sundaresan
IBM Almaden Research Center
650 Harry Rd.
San Jose, CA 95120
neel@yahoo.com

Abstract

This paper presents a relevant term discoverer, a system that discovers relevant topics of a given topic from the World Wide Web. The system mines hyperlink metadata on the basis of the association of terms in the metadata. It also applies various filtering techniques to detect false positives and false negatives. The applications of the system include i) topic-specific information gathering systems that need to crawl resources of the relevant topic, ii) bibliography search system that need to extend their search to the articles of relevant topics, iii) classification systems that can categorize items of similar class together, and so on.

We report a successful application of the system to build a topic-specific search engine dedicated to eXtensible Markup Language (XML). Using the algorithms presented in this paper, we were able to identify the relevant topics that the search engine needs to cover. Together with effective topic-directed crawling algorithms, we were able to build a topic-specific search engine that require significantly less human labor but perform almost as well as topic-specific search engines whose content is maintained by humans.

1. Introduction

The World Wide Web (WWW) is a vast source of information. However, there are not many devices that sift useful information to the user from the web pages. Search engines play an important role in furnishing the information to the end user. However, the kinds of information one can expect using conventional search engines are very limited. Conventional search engines are designed to identify pages with specified phrases. Thus, for example, they are not able to search for relevant topics of a given topic. For instance, if a user wants to find out what are relevant topics to XML (eXtensible Markup Language), she/he has to search some documents about XML, perhaps using conventional search engines, and then learn what are the relevant to XML from the contents of the documents.

In this paper we present a data mining solution for identifying relevant things such as relevant topics. The main idea is to analyze the contents of the pages that are hyperlinked together by a page. If pages of two different topics are frequently hyperlinked together by many pages, we consider the two topics are relevant. The decision is based on the generally agreed observation that web page authors often hyperlink relevant pages together [3].

We utilize metadata of web pages in analyzing the contents (or topic) of web pages. Here metadata of a web page, y, is the description about the page, y, furnished by the creator of another web page, x, that hyperlinks to y. As the number and complexity (in terms of the scripts, graphics, animations, audios, and videos) of web pages grow, to study these pages based upon their entire contents can become expensive and complex [7]. As an alternative, it is possible to look at the pages that point to a page, and, to learn what the page is about on the basis of what other pages say about it. In this way, it is possible to learn from other people’s comments.

There are various applications that can be enhanced by the relevant term discovery: i) topic-specific information gathering systems, such as focused crawlers[9], that need to crawl resources of the relevant topic, ii) bibliography search system that need to extend their search to the articles of relevant topics, iii) classification systems that can categorize items of similar class together, and so on.

The Organization of the Paper The rest of the paper is organized as follows. Section 2 introduces a topic-specific information gathering system, a prototype system that demonstrates the utility of relevant topic mining. Section 3 lists metadata available in web pages and explains metadata extraction methods. Section 4 presents the relevant term mining algorithms and the experimental results. We review related work in section 5. Section 6 contains concluding remarks.
2 Topic-Specific Web Information Gathering

The WWW reached an estimated 800 million publicly indexable pages in Feb. 1999 [15]. It continues to grow at an alarming rate stretching most general-purpose search engines beyond their limits, both qualitatively and quantitatively. They suffer from low precision problem: they typically return too many irrelevant web pages in response to a user query. Moreover, the coverage of the major search engines is dwindling [14, 15]. The individual coverage by the largest search engine dropped from 33 % (Dec. 1997) to 16 % (Feb. 1999) of the estimated web size in each time of estimation, with the combined coverage of major search engines from 60 % (Dec. 1997) to 42 % (Feb. 1999) [14, 15]. Internet directories like Yahoo! 1 or topic-specific search engines like MathSearch 2 offer search results of higher quality. However they also have limitations: building and maintaining those engines require intensive human efforts which decreases their cost-efficiency ratio significantly.

It is our goal to develop topic-specific search engines that can be built and maintained with significantly less human labor while performing almost as well as those maintained by humans. Typical web crawlers crawl the web indiscriminately with little attention to the quality of search information. These crawlers have a goal of retrieving as many pages as possible. Topic-specific crawlers have a different objective. They focus on getting to as many pages related to their topic of interest and as fast as possible without deviating to unrelated pages. We identify major challenges of building a topic-specific system: (i) to define the target topic at an appropriate level of detail and (ii) to identify the web pages that qualify under each level of this topic definition. In addition, it has to be very efficient in dealing with the massive data on the web.

2.1 Target Topic Refinement

Typically a target topic consists of many sub-topics. In addition, it includes many other topics that are relevant because they share some property with the target topic. The system for a topic-specific search engine needs to identify such sub-topics and relevant topics in order to gather web pages completely for the given topic.

2.2 Topic-Directed Crawling

We need to direct the crawlers (i) in order to avoid the crawler drifting away from the given topic, and thus (ii) to search the web efficiently for web pages pertaining to the given topic. Crawling procedures based on breadth first

1 http://www.yahoo.com
2 http://www.maths.ual.edu.au/MathSearch.html

search from a given seed URL are too costly for topic-specific crawling because web pages often link to topic-irrelevant pages. This is out of the scope of this paper. In detail description of topic-directed crawling strategies can be found in [23].

2.3 Prototype System

We have built xCentral 3, a search engine for developers and users of XML [5]. xCentral indexes web pages about XML and XML-related technologies, as well as XML documents. XML is a meta language and only provides a framework within which to build domain-specific languages, such as CML for Chemistry, OFX for Finance, MathML for Mathematics, and so on. In addition, XML gives rise to various tools and systems related to each of these domain-specific languages and to XML in general. Thus, an ideal XML-specific search engine must index all the resources related to these languages, tools, and systems. Hence, a crawler must not only be able to identify web pages pertaining to the given topic; it must also learn all of the topic terms related to XML.

Figure 1 shows the architecture of the xCentral system. The system includes i) a gatherer that selectively crawls the web, ii) a summarizer that produces RDF[13] metadata for the crawled pages, and iii) a topic expander that incrementally expand topic definition as crawling of the web proceeds.

3 Metadata in Web Documents

Hyperlink metadata, or simply metadata, provide metadata information about the documents which they reference. Consider the citation of a research paper by another paper. In addition to providing information that the paper has been cited by another paper the citation offers a reader, without her having actually to read the paper, information about it through the metadata made by the citing paper. Web documents are of a similar nature. Citations in web pages are in the form of hyperlinks. Metadata for the citations are provided by the attributes of and the text around these hyperlinks.

Most web documents are in HTML form. Newer web documents would use XML structures with possibly associated schema. Our web crawler understands hyperlinks and its associated text in languages that have explicit syntax for hyperlinks (e.g. HTML, XML). To handle document types that do not have this explicit hyperlink syntax or document types that our crawler does not understand, it extracts out the strings in the documents, looks for hyperlink patterns, and gets the surrounding text as metadata. In this paper we

3 http://www.ibm.com/development/xml
3.1 HTML Metadata

In HTML documents we find four kinds of hyperlinks:
- anchor (<A>) tags
- Image (<IMG>) tags
- Map and Area tags
- Frame and iFrame tags

The most commonly used tags are the Anchor tags. Each of the tags have attributes associated with them. The main attributes of the anchor tags and the Area tags include name, title, alt, on-mouse-over, and href. The Image tags have attributes that include name, alt, src, dynsrc, lowsrc, onabort, onload, and onerror.

In addition to using these attributes for hyperlink metadata, we also get metadata information from surrounding text, from text contained inside these tags, and from text at the parent level of these tags. The text at the parent level of the anchor tags is useful when the page lists outgoing links using a set of <li> tags within a list structure (like ul or ol). The overall summary of these <li> elements is often found at the parent level with each list-item pertaining to a similar idea or topic. An example is a web page categorizing publications within a subject area. Figure 2 shows the excerpt from a publication web page of Almaden Research Center⁴.

Here all the papers related to "Association Mining" are listed together with a common metadata element (an H3 tag)

and all the papers related to "Classification" are listed together under a different metadata element.

We want to identify the attributes that are most appropriate for the purpose of metadata. In short, we would like to treat the HTML tags as structured XML elements and extract the right kind of metadata.

3.2 Metadata Extraction

In order to extract metadata from hyperlinks in an HTML document, we first convert the HTML documents to well-formed XML documents. The browsers tend to be very forgiving and accept and display documents that are poorly formed HTML. We have an HTML to XML filter that converts the HTML documents to well-formed XML documents performing extensive error recovery in case of poorly formed HTML documents. Once we have a well-formed XML document, we check the document for elements with names corresponding to the hyperlink elements (A for anchor tags, IMG for image tags, and so on), and extract their attribute values. In order to identify the surrounding text, we identify XML elements of type PCDATA which are left and right siblings of these tags. For text contained inside an metadata tag, we look for PCDATA inside the HTML tags. Often, HTML pages contain a list of links to pages on a topic. The information about these topics is often provided at the parent level of this list. In order to identify in the XML document we pick the text data at the ancestral level where the closest sibling of a previous ancestor that is a text node.

We studied a sample set of 20,000 HTML pages to analyze hyperlink metadata. In these pages, over 206,000 hyperlink references were found. Some of these hyperlinks

point to pages within the sample set, while others point to pages outside the sample set. Table 1 lists characteristics of several attributes and metadata types surrounding a hyperlink. It can be seen that, second to HREF, anchor text is the most common metadata type. In addition to the frequency of each metadata type, the quality of an metadata type is measured by the number of relevant topic terms discovered by its use during mining. Also, the absolute and relative reliabilities of metadata types are measured by what the metadata says about the page it references and what the page really contains. This measurement also takes into account outdated or expired references. From our experiments we found that anchor text, HREF and surrounding text, in that order, provide good metadata.

In the results reported in this paper we used anchor text that most frequently occur and is most reliable. Alternative schemes like choosing weighted averages of different metadata type occurrences may also be applied.

4 Topic Expansion

The discovery of relevant topics occurs in two phases.

1. Mining for topic terms potentially relevant to the target topic (Candidate Term Mining).

2. Refining the relevance measure of candidate terms to find the relevant ones (Filtering).

Figure 3 illustrates the overall scheme for relevant topic mining. Note that the components in the shaded box are out of the scope of this paper. In order to mine candidate terms more effectively, we can optionally apply domain knowledge to reduce (or increase) the set of terms to be mined. Some techniques of extracting such knowledge from the text of web pages is introduced in [6, 20, 21]. In our prototype system, we added acronyms and their expansions to add multi-word topic terms to the set of terms to be mined. For example, we treat “eXtensible Markup Language” as a unit of topic term rather than three individual terms. In
Table 1. This table lists various attributes associated with a hyperlink. The numbers in the column “Hyperlinks” gives the number (and the percentage) of hyperlinks that are referenced with a particular metadata type. The column “Pages” gives the number of pages which contain at least one particular metadata type.

<table>
<thead>
<tr>
<th>Metadata Type</th>
<th>Hyperlinks</th>
<th>Pages</th>
</tr>
</thead>
<tbody>
<tr>
<td>ALT tag</td>
<td>1,890</td>
<td>281 (1.5%)</td>
</tr>
<tr>
<td>Anchor Text</td>
<td>147,745</td>
<td>14320 (76%)</td>
</tr>
<tr>
<td>HREF tag</td>
<td>176,412</td>
<td>16313 (87%)</td>
</tr>
<tr>
<td>NAME tag</td>
<td>5,487</td>
<td>779 (4.1%)</td>
</tr>
<tr>
<td>ONHOUSEOVER tag</td>
<td>9,383</td>
<td>1523 (8.1%)</td>
</tr>
<tr>
<td>Surrounding Text</td>
<td>49,138</td>
<td>8424 (45%)</td>
</tr>
<tr>
<td>Title</td>
<td>885</td>
<td>249 (1.3%)</td>
</tr>
</tbody>
</table>

addition, the expansion “eXtensible Markup Language” is treated the same as the abbreviation “XML”. [22] discusses a technique to build generalization hierarchy from the term descriptions.

We set up notations used in this section in table 2. The following subsections provide in-depth explanations of the algorithms used in candidate term mining and filtering phases.

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>τ</td>
<td>the target topic</td>
</tr>
<tr>
<td>t</td>
<td>a topic term</td>
</tr>
<tr>
<td>ct</td>
<td>a candidate topic</td>
</tr>
<tr>
<td>T</td>
<td>the set of topic terms</td>
</tr>
<tr>
<td>CT</td>
<td>the set of candidate topics</td>
</tr>
<tr>
<td>L</td>
<td>a lexicon</td>
</tr>
<tr>
<td>r</td>
<td>a sampling metric</td>
</tr>
</tbody>
</table>

Table 2. Symbols used in section 4

4.1 Discovering Candidate Topic Terms

Association mining is used to extract the subset of terms that are highly co-occurred with the target topic, what we call the candidate topics, from the metadata of hyperlink references. The use of metadata rather than the full-text of the linked page is based on two assumptions: (i) The hyperlink metadata provide good summary descriptions of the page it points to provided by the creator of the web page that contains the hyperlink, and (ii) Web pages reference pages of related content with some noise. [12, 4, 9, 16].

Hyperlink metadata is preferred to the remaining content of the page itself for the purpose of candidate topic mining in various reasons:

1. Mining based on link metadata requires significantly less number of web page visits, because it does not require lookup of all linked pages. This saves us from downloading irrelevant pages.

2. The association mining using entire web pages not necessarily guarantees the improved accuracy. In reality, it not only increase computational complexity, but also, it might well degrade the accuracy due to the curse of dimensionality problem.

Our algorithm identifies candidate terms (ct) by finding strong association between the target topic, τ, and terms appearing in the hyperlink metadata. Strong association rules are the rules that have confidence and support values higher than the threshold values. The terms identified by these rules are surmised to be relevant to the target topic.

4.1.1 Relevance Metrics

Candidate terms are the terms strongly associated with the target topic, τ. This means that, association between a candidate term, ct, and τ should have both high support and high confidence. This premise leads us to use support and confidence as metrics of relevance.

Due to the space limitations, we avoid a detailed discussion of the support and confidence measures. Interested readers may refer to [1, 2, 18]. Instead, we simply make note of the following points:

(i) Confidence, \( p(τ|t) = \frac{w(t, τ)}{p(t)} \), is a good measure of the implications of a term to τ. It is biased, however, in that it favors rare but strongly associated topics and disfavors commonly co-occurring terms.

(ii) Support of an association rule, \( p(τ, t) \), is a good metric for finding major topics related to the target topic. It favors popular topics associated with τ but exhibits the drawback of indicating only rules that are already well-known.

4.1.2 Algorithm: Candidate Topic Term Discovery

1. Each web page is considered to be a market-basket (i.e., a transaction).

2. Each term in the hyperlink metadata of the page is considered to be a transaction item.

3. \( R \) is a set of significant association rules: \( R = \{ t_i \Rightarrow τ \mid \text{relevance}(t_i \Rightarrow τ) \geq c \} \)

   where \( c \) is a user defined threshold, and relevance(·) is a relevance measure.

4. From \( R \), the set of candidate topics, \( CT \), is built as follows: \( CT = \{ ct_i \mid ct_i \Rightarrow τ, \{ ct_i \Rightarrow τ \} \in R \} \)
4.2 Discovering Relevant Topics

Though the candidate topics are highly associated with \( \tau \), their relevance to \( \tau \) is not 100% guaranteed because it allows the presence of potentially false associations (strong association rules that involve irrelevant terms). To compensate, we apply two filtering techniques, specialization and sampling.

4.2.1 Sampling

In topic-specific crawling, the gathered web pages, \( P \), are biased towards \( \tau \). Our sampling techniques exploit this bias to determine whether each candidate topic, \( ct \), is indeed relevant as follows: for each \( ct \) to test, a sample set \( S \) is gathered targeted not for \( \tau \) but for \( ct \). Ideally, this set represents the actual distribution of all pages in the entire WWW which contain \( ct \).

A new topic, \( ct \), is not relevant to \( \tau \), if

\[
r = \frac{p_S(\tau | ct)}{p_P(\tau | ct)} < c,
\]

where \( p_S(\tau | ct) \) and \( p_P(\tau | ct) \) are the confidence of \( ct \) in \( S \) and \( P \), respectively, and \( c \) is the user-defined threshold. Assuming that \( S \) is a good reflection of the distribution of \( ct \), the ratio \( r \) reflects the degree of bias of \( ct \) in \( P \). If the ratio \( r > 1 \), \( P \) underestimates the actual confidence of association between \( \tau \) and \( ct \). If the ratio \( r < 1 \), \( P \) overestimates it.

4.2.2 Specialization

The specialization filter concludes that a topic, \( t_1 \), is relevant to \( \tau \), i) if \( t_1 \) is specialization of another topic, \( t_2 \), and \( t_2 \) is relevant to \( \tau \), or ii) if \( t_1 \) is specialization of \( t_2 \), and \( t_2 \) is generalization (but not over-generalization) of \( \tau \). A term \( t_1 \) is over-generalization of \( t_2 \), if \( t_1 \) is a generalization of \( t_2 \), but the generalization relationship does not hold in sample set. In order to detect over-generalization, the sampling technique discussed in section 4.2.1 is applied to the generalized topics.

Specialization is less expensive than sampling. Sampling calls for potentially irrelevant pages (which will ultimately be discarded) to be crawled. We apply specialization in the earlier stage of filtering. Fortunately, the proportion of topics to be filtered by sampling diminishes over iterations, as shown in table 3.

4.3 Experiments

Table 3 summarizes the results of topic expansion experiments. Association mining, applied to the link metadata of 17000 web pages, produced a high quality set of can-
Table 3. Number of Candidate Topics Discovered by Association Mining and Filtering. Thresholds for support = 0.1%, 0.04%, 0.02%, and 0.012% for 1st to 4th iterations, respectively, and threshold for confidence = 0.01% in all cases. The actual number of relevant topics (item 3) in the candidate topic set is computed by manual inspection. The numbers in parentheses in item 4 indicate the percentage of relevant terms discovered by specialization filter among candidate topics (item 2). Items 5-8 & 9-10: Number of relevant topics by sampling with $c=0.1$ and $c=0.01$, respectively, using sampling only (5,9), using sampling with specialization (6,10). Items 7-8 & 11-12: Number of false inclusions and exclusions. (Note that all relevance numbers, (4,5,6,7,10,11) are corrected for false inclusion.)

<table>
<thead>
<tr>
<th></th>
<th>1st Iteration</th>
<th>2nd</th>
<th>3rd</th>
<th>4th</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. # of pages crawled</td>
<td>2000</td>
<td>5000</td>
<td>10000</td>
<td>17000</td>
</tr>
<tr>
<td>2. candidate topics</td>
<td>130</td>
<td>110</td>
<td>79</td>
<td>67</td>
</tr>
<tr>
<td>3. Actual Relevant topics</td>
<td>29</td>
<td>37</td>
<td>41</td>
<td>54</td>
</tr>
<tr>
<td>4. Relevant topics by Specialization</td>
<td>15 (12.5%)</td>
<td>32 (30%)</td>
<td>37 (47%)</td>
<td>49 (73%)</td>
</tr>
<tr>
<td>5. Relevant topics by Sampling ($c=0.1$)</td>
<td>9</td>
<td>2</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>6. Relevant topics by Specialization &amp; Sampling ($c=0.1$)</td>
<td>24</td>
<td>34</td>
<td>37</td>
<td>49</td>
</tr>
<tr>
<td>7. False Exclusion ($c=0.1$)</td>
<td>5</td>
<td>3</td>
<td>4</td>
<td>5</td>
</tr>
<tr>
<td>8. False Inclusion ($c=0.01$)</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>9. Relevant topics by Sampling ($c=0.01$)</td>
<td>14</td>
<td>4</td>
<td>4</td>
<td>5</td>
</tr>
<tr>
<td>10. Relevant topics by Specialization &amp; Sampling ($c=0.01$)</td>
<td>29</td>
<td>37</td>
<td>41</td>
<td>53</td>
</tr>
<tr>
<td>11. False Exclusion ($c=0.01$)</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>12. False Inclusion ($c=0.01$)</td>
<td>43</td>
<td>30</td>
<td>15</td>
<td>19</td>
</tr>
</tbody>
</table>

Candidate topics. Of the 67 candidate topics mined, 54 were XML-related (81%). The specialization filter also produced impressive results. 49 of the candidate topics passed the specialization filter (73%). These learned topics were, in fact, all XML-related and previously unknown to us.

By contrast to its impressive performance on the 4th iteration, the specialization filter's rate of discovery was low initially (12.5%). This number reflects the large number of irrelevant topics found in the initial candidate set (70%). Over iterations, the effectiveness of both the mining algorithm and the specialization filter improved significantly. As expected, the effectiveness of the specialization filter further increased as the number of known relevant topics increased reflecting the expansion of the topic hierarchy. By the 4th iteration, over 80% of the candidate topics were relevant, 90% of this subset distilled by the specialization filter. Moreover, without a good topic hierarchy, sampling accomplishes a large proportion of the learning (item 5).

Topic discovery by sampling was tested using various threshold values (equation 1). We measure the quality of learning in terms of relevant topics discovered, false inclusion, and false exclusion. False inclusion refers to the irrelevant topic terms that are included in the set of relevant topics by the algorithm. False exclusion refers to the relevant topic terms that are not included in the set of relevant topics by the algorithm. Results using the values $c=0.1$ and $c=0.01$ (items 5-8, 9-12) show that a decrease in $t$ causes noise levels to increase at a faster rate than the increase in the discovery rate. To be specific, the increase in the number of false inclusions at $c=0.01$ vs. 0.1 (items 12,8) is much greater than the increase in the number of relevant topics found (items 9,5). In the same respect, the number of false exclusions (item 11) shows a negligible decrease (item 7) compared with the parallel increase in false inclusion.

Figure 4 summarizes the effectiveness of candidate topic learning and filtering, respectively. Over iteration, as demonstrated by figure 4 (a), the accuracy of the candidate topic mining algorithm is significantly improved. In the first iteration, only 24% of candidate topics are relevant to $t$. In the 4th iteration, 81% of candidate topics are relevant.

In figure 4 (b), the specialization filter demonstrates a significant capacity for learning. Initially, it detects only 50% of relevant topics, thus the rest of 50% of relevant topic terms is passed on to the sampling filter. In the 2nd iteration, the specialization filter detected 83% of relevant topics. By the 3rd iteration, the specialization by itself successfully identifies all of the possible relevant topics, eliminating the need for the less efficient sampling filter. Finally, at all stages, the combination of specialization and sampling discovers 80 - 90% of the relevant topics (the remaining 10-20% lost to false inclusion and exclusion).
Figure 4. Performance of Relevant Topic Learning (a) The Effectiveness of Candidate Topic Mining: This graph depicts the accuracy of association mining, i.e., the percentage of relevant topics among candidate topics. Initially, only 24% of candidate topics were indeed relevant to the target topic. After 4th iteration, the accuracy was improved to 81%. (b) The Effectiveness of Filters: This graph shows the accuracy of filters, the percentage of relevant topics identified by (i) specialization filter only, and (ii) specialization and sampling filters combined, respectively. Initially, specialization filter alone detected only 50% of relevant terms in the set of candidate topics, while specialization and sampling filters combined achieved 83% accuracy. After 3rd iteration, both schemes converged to above 90% accuracy.

5 Related Work

Work in scalable data mining technology [1, 2] has been successfully applied to identify co-occurring patterns in many real world problems like market basket analysis. Q-Pilot [19] attempts to dynamically route each user query to the appropriate topic-specific search engines. In order to increase the accuracy, Q-Pilot applies query expansion, a technique for obtaining the terms relevant to a query. It finds relevant terms from the web pages containing the query term on the basis of co-occurrence of the terms.

For the purpose of topic-centric query routing, it is used mainly for evaluating the relevance of a query to the identified topic terms of search engines.

expands user queries on the basis tries to discover the [11] guides web crawlers to gather a set of similar web pages by ranking web pages based on both content and link similarity measures. Similar to our system, theirs defines the ranking measure by the presence of a topic word. It differs from our system, however, in that the measure does not account for topic relevance. Moreover, their similarity decision requires the entire text of the page.

In [12], Kleinberg proposes using the link structure in hypertext documents to discover authoritative pages. The techniques used in [4, 8] use edge/link weighting based on relevance in order to enhance Kleinberg's method. Chen et. al. [10] also explore and refine the techniques of utilizing metadata for information retrieval.

Focused crawler (FC) [9] utilizes both web link structure information and content similarity based on document classification. FC includes all URLs that are pointed to by a relevant page regardless of their relevance to the topic. This results in unnecessary visits of many irrelevant pages. Our system minimizes the visits of irrelevant pages by making the decision based mostly on the metadata of individual URL. In addition, the topic scope of FC is limited to that of example (or seed) pages and can be refined only by user intervention, i.e., by providing more example web pages. Our system, by contrast, incrementally extends the topic term set through fully automated means. Hence, requiring only a limited scope of seed pages, our system achieves a comprehensive coverage of the target pages.

Cora [17] is a search engine for research papers in computer science. It applies reinforcement learning to gather papers (in postscript format) starting from computer science department homepages. It focuses on analyzing papers for title, author, and cross reference information to augment the power of the search. The problem of guiding a crawler to
find papers in postscript format is relatively straightforward (compared to our case). The URL of a postscript document serves to identify it as such.

6 Concluding Remarks

We presented a web mining algorithm that discovers relevant topics of a given topic, called topic expansion. We built topic-specific information gathering system that utilizes the topic expansion algorithm. The topic expansion algorithm is very effective. The experimental results of our prototype system demonstrated that the algorithm achieved an impressive coverage of domain-specific languages, tools, and systems related to XML, the target topic.

Thanks to the topic expansion, our information gathering system does not require additional intervention in order to expand the search scope, unlike some other topic-specific crawling systems that require additional seed pages in order to add a new topic. When the learning component identifies a new relevant topic, that raises the relevance of the pages of which metadata is signified by the new topic. Thus, the system automatically go after the pages, which was previously ignored due to the lack of the knowledge of the relevance of the topic.

References


