A NOVEL TWO-PHASE PAGE FEATURE AND KTH KEY PHRASE FINGERPRINT BASED DUPLICATE DETECTION TECHNIQUE

Ashlesha Gupta  
Assistant Professor  
Computer Engineering Deptt., YMCAUST  
Faridabad, Haryana, India

Ashutosh Dixit  
Associate Professor  
Computer Engineering Deptt., YMCAUST  
Faridabad, Haryana, India

A.K. Sharma  
Professor & Dean(Research)  
Computer Engineering Deptt., BSAITM  
Faridabad, Haryana, India

Abstract: The World Wide Web is a huge repository of network-accessible information including text, image, audio, video and metadata. With rapid increase in information resources available via WWW and users of the Internet, it is becoming difficult to manage and access the desired information on the web. Therefore, majority of users use information retrieval tools like search engines to find the desired information from the WWW. Web search engines work by storing information about many web pages, which they retrieve from the WWW itself in search engine repositories. These repositories may contain duplicate or near duplicate pages which results in higher storage area and processing costs. To overcome these increased space /cost requirements and to provide redundant free results to the users duplicate detection and elimination algorithms are used. The proposed duplicate detection approach uses two phase page level and kth key phrase fingerprint based scheme to detect and eliminate near duplicate web pages so that quality of the result-sets may be improved.

Keywords: Duplicate page, Near-duplicate page, Filtering, Finger-print, Page-features

I. INTRODUCTION

World Wide Web is a huge, diverse and dynamic source of information which is expanding day by day. In order to trace relevant data, users depend on variety of search engines for finding suitable answers for their queries. A Search Engine is an information retrieval system which helps users find information on WWW by making the web pages related to their query available. It gathers, analyzes and organizes the data from the internet and offers users an interface to retrieve the network resources[1]. With the search engine user types in the query keywords and in response to it, Search Engine returns a list of clickable URL’s. This returned result set of a search engine, however, contains a mixture of both relevant and irrelevant information including duplicate and near-duplicate web pages. Duplicate web pages are mirrored copies of some web pages and near duplicates are not bit wise identical but differ in some parts of the web page like the advertisements, counters and timestamps [2]. Multiple versions of the same page, storing the document on multiple servers, creating documents using same templates or spamming are the probable reasons of existence of duplicate pages[3]. Early recognition of these duplicates help in reducing network bandwidth and storing costs.

To detect duplicate and near duplicate web pages a novel duplicate detection technique based on fingerprint similarity is being proposed. In the proposed technique firstly a web page feature based filtering is applied to eliminate the far similar documents and then kth key phrase fingerprint technique of the web page is computed and a bit-by-bit difference is measured between new crawled web page and other web pages in the repository. If the difference is less than the threshold value, the crawled page is a near duplicate and is discarded else the page is stored in the repository. The objective of the technique is to quickly and accurately identify duplicate and near duplicate web pages thereby improving the quality of search results and saving storage space and hence network bandwidth.

The rest of the paper is organized as follows: The Existing approaches to Duplicate detection are covered in Section II. Section III discusses the proposed duplicate detection technique. Experimental set-up and results are summarized in Section V. Section VI includes conclusion and Future scope.

II. LITERATURE REVIEW

Duplicate Detection techniques help search engines in providing quality and redundant free results to the user. A number of methods have been proposed for recognizing and eliminating such duplicates. Broder et al. [4] have suggested a technique, in which all sequences of adjacent words are extracted. If two documents contain the same shingles set they are treated as equivalent and if the shingles set overlaps, they are considered as exact similar. But this method does not work well on small documents. Theobald et al.’s [5] proposed SpotSigs method[6] for duplicate detection. In this method stop words in anchor tag were first identified then k tokens after an anchor leaving the stop words were grouped as k-gram and were referred as spot signatures. The document was then represented as a collection of these spot signatures. The
length of these spot signature vectors were reduced using hash function. The documents with similar spot signatures were identified as duplicates. Fetterly et al. [7] use five-gram as a shingle and sample 84 shingles for each document. Then the 84 shingles are built into six super shingles. The documents having two super shingles in common are considered as nearly duplicate documents. A sentence level duplicate detection technique for news-articles was proposed by Hung-Chi Chang and Ten-Hour Wang [8]. Hannaneh Hajishirzi et al. [9] developed duplicate detection technique for identifying duplicates in same domains. Every document in the proposed technique was treated as a k-gram vector. These k-gram vectors were then mapped to hash-values as document signatures through locality sensitive hashing scheme. Bingfeng Pi et al. [10] proposed the use of SimHash algorithm for finding near duplicate. Narayana et al. [11] proposed duplicate detection technique wherein the keywords of the document are extracted and keyword similarity score between newly crawled and stored web pages is used for duplicate detection. Salha Alzahrani et al. [12] suggested a fuzzy based semantic method for detecting plagiarism.

III. PROPOSED TECHNIQUE

For fast and efficient detection of duplicate pages, a novel duplicate detection approach is being proposed that uses two phase filtering techniques to detect duplicates and near duplicates. In the proposed approach first a web page level feature based comparison among the new and stored web pages is done. This action eliminates the far similar documents and thus reduces the number of web documents to the second level of filtering. Since only numerical values are compared so this filtering requires less time and disk space. In the second stage fingerprint of all kth keywords in the web pages are computed and a bit by bit difference between the fingerprints of the crawled and stored web pages is made. If the difference is less than the specified threshold value, the crawled page is near duplicate and dropped otherwise the crawled page is stored in the repository. Fingerprints of only limited web pages need to be created due to initial page level feature filtering, the proposed technique incurs less time for duplicate detection and provides precise results. The working of the proposed technique is given in Figure 1

A. Preprocessing

The newly crawled page is parsed and preprocessed for page level feature extraction. The Preprocessing includes stop-word removal and stemming. Following web page features are extracted from the parsed and preprocessed web page: No. of tables in the web page, No.of images/figures in the web page, No. of sentences, No. of Anchor tags and keywords of the web page. The information is then passed to Level 1 filtering.

B. Level-1 Filtering

After preprocessing, Level 1 filtering is used for finding the duplicate web pages. The filtering involves comparing the extracted page level features of the newly added web page with the similar extracted features of already stored web pages and assigning a score value to the compared page. The score value is assigned as per the details given in Table 1.

<table>
<thead>
<tr>
<th>Feature</th>
<th>Threshold Value</th>
<th>Computed Value</th>
<th>Score</th>
</tr>
</thead>
<tbody>
<tr>
<td>No. of tables</td>
<td>A1</td>
<td>Tn-To where Tn and To refer to number of tables in new and old web page</td>
<td>If Tn-To is less than A1 a score of 1 is assigned else score value is 0</td>
</tr>
<tr>
<td>No. of Images/Figures</td>
<td>A2</td>
<td>In-Io where In and Io refer to number of images in new and old web page</td>
<td>If In-Io is less than A2 a score of 1 is assigned else score value is 0</td>
</tr>
<tr>
<td>No. of Anchor tags</td>
<td>A3</td>
<td>An-Ao where An and Ao refer to number of anchor tags in new and old web page</td>
<td>If An-Ao is less than A3 a score of 1 is assigned else score value is 0</td>
</tr>
<tr>
<td>No. of Sentences</td>
<td>A4</td>
<td>Sn-So where Sn and So refer to number of sentences in new and old web page</td>
<td>If Sn-So is less than A4 a score of 1 is assigned else score value is 0</td>
</tr>
<tr>
<td>Keyword Similarity</td>
<td>A5</td>
<td>No. of Common</td>
<td>If KS is less than</td>
</tr>
</tbody>
</table>

Figure 1: Working of Proposed Technique
words in new and old web page
Total number of words in new and old page
A5 a score of 0 is assigned else score value is 1

The scores thus assigned for each feature are combined and a final score value is calculated. If the total score value is greater or equal to the specified threshold score, then the compared page is filtered in. The advantage of this step is that it is fast and consumes less disk space since only numbers are compared and reduces the number of web pages for next level of filtering. Hence, the number of inputs to the fingerprint comparison will get reduce, so that fingerprint of only a limited number of web page is to be found out, rather than all the web pages in the database.

C. Fingerprint Comparison
Fingerprints of the filtered web pages is then computed. Instead of finding the fingerprint of the entire document, kth keyphrases are selected and fingerprint of these kth keyphrases are computed. To compute the fingerprint first character of each kth key phrase is taken and is converted to its ascii value. The sum of all ascii values is computed and converted to binary form. This process of finger-print comparison is repeated with each document filtered in first phase.

D. Near Duplicate Detetion
For near duplicate detection first a threshold is set. The kth keyphrase fingerprint of the newly crawled page is compared with fingerprint of all other stored pages. The comparison is done bit-by-bit. If the difference is less than the threshold value, the document is discarded as it is a near-duplicate else the newly crawled page is added to the search engine repository.

IV. RESULTS & DISCUSSION
The proposed duplicate detection approach has been implemented in Java with Net beans as frontend and MS Access as the backend. The query terms “SEO”, “Web-Crawler” and “I-Phones” were given to Google search engine. The results of first 5 pages are being used for experimentation. The results of the study are being given in following sub-sections. The analysis of the experimental results confirmed that the proposed approach is able to achieve its objective of time and space reduction.

A. Experimental Results
For providing sample results, we take 3 web page documents from web crawling procedure. The web pages were first processed to remove stop words and then process of stemming was applied. The web page documents along with the extracted keywords are given in the Table 2.

<table>
<thead>
<tr>
<th>Web page documents</th>
<th>Keywords</th>
</tr>
</thead>
<tbody>
<tr>
<td>WP1</td>
<td>Web, Search, Engine, Software, information, mine, data, editor, crawler, database, WWW, System, design, user, index, huge, site, directory</td>
</tr>
<tr>
<td>WP2</td>
<td>Search, Engine, document, keywords, list, Google, Bing, list, specific, WWW</td>
</tr>
<tr>
<td>WP3</td>
<td>Internet, Search, Engine, program, information, crawler, index, Bing, WWW, directory</td>
</tr>
</tbody>
</table>

Table 2: Web documents along with extracted keywords

<table>
<thead>
<tr>
<th>Web Page</th>
<th>No. of Tables</th>
<th>No. of Figures</th>
<th>No. of Anchor tags</th>
<th>No. of Sentences</th>
</tr>
</thead>
<tbody>
<tr>
<td>WP1</td>
<td>0</td>
<td>2</td>
<td>6</td>
<td>6</td>
</tr>
<tr>
<td>WP2</td>
<td>0</td>
<td>1</td>
<td>9</td>
<td>4</td>
</tr>
<tr>
<td>WP3</td>
<td>0</td>
<td>2</td>
<td>5</td>
<td>5</td>
</tr>
</tbody>
</table>

Table 3: Page Level Features of web Documents

<table>
<thead>
<tr>
<th>Web Document</th>
<th>Fingerprint</th>
</tr>
</thead>
<tbody>
<tr>
<td>WP1</td>
<td>001001001100111</td>
</tr>
<tr>
<td>WP2</td>
<td>1001011100011110</td>
</tr>
<tr>
<td>WP3</td>
<td>00011100011110</td>
</tr>
</tbody>
</table>

Table 4: Fingerprint of web documents

For near duplicate web page detection, Cw is taken as newly crawled web page. The page-level features of Cw were extracted and compared with similar features of all three pages. The threshold values for A1,A2,A3,A4 and A5 were set equal to 2,2,2,2 and 50% respectively.

Table 5: Page level features and Fingerprint of new web page Wpnw

<table>
<thead>
<tr>
<th>Web Page</th>
<th>Keywords</th>
<th>No. of Tables</th>
<th>No. of Figures</th>
<th>No. of Anchor tags</th>
<th>No. of Sentences</th>
<th>Fingerprint</th>
</tr>
</thead>
<tbody>
<tr>
<td>new</td>
<td>Search, Engine, Mine, data, informati on, user</td>
<td>0</td>
<td>2</td>
<td>8</td>
<td>6</td>
<td>00101110011110</td>
</tr>
</tbody>
</table>

© 2015-19, IJARCS All Rights Reserved
Since WP1 and WP2 match Cw in page level 1 filtering, they are selected for kth keyphrase filtering. The next process is comparing bit-by-bit the kth keyphrase fingerprints of Cw with fingerprint of WP1 and WP3. The threshold value is set to 3. By comparing the fingerprint of Cw with WP1 and WP3 it was found that the difference between Cw and WP1 is 1 which is less than the specified threshold value of 3. Hence, Cw is considered as near duplicate web page and it is not added to the database.

Table 6: Comparison Table

<table>
<thead>
<tr>
<th>Dataset</th>
<th>Count of web pages considered</th>
<th>Precision of NDupDet</th>
<th>Recall of NDupDet</th>
<th>Time to detect duplicates (ms)</th>
<th>Precision of Proposed Technique</th>
<th>Recall of Proposed Technique</th>
<th>Time to detect duplicates (ms)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Search Engine</td>
<td>50</td>
<td>0.7</td>
<td>0.4</td>
<td>180</td>
<td>0.8</td>
<td>0.8</td>
<td>120</td>
</tr>
<tr>
<td>Crawler</td>
<td>45</td>
<td>0.58</td>
<td>0.5</td>
<td>160</td>
<td>0.8</td>
<td>0.75</td>
<td>80</td>
</tr>
<tr>
<td>SEO</td>
<td>35</td>
<td>0.5</td>
<td>0.5</td>
<td>120</td>
<td>0.7</td>
<td>0.7</td>
<td>60</td>
</tr>
</tbody>
</table>

B. Performance Evaluation

The proposed duplicate detection approach is compared with Near Duplicate Web Page Detection using NDupDet Algorithm in terms of Precision, Recall and Computation time where Precision and Recall are defined as:

Precision=No. of true duplicates detected/Total number of duplicate detected.
Recall= No. of true duplicate detected/Total number of duplicates in the dataset
Computation time= Time required to identify near duplicates.

Table 6 lists out the precision, recall and computation time values for NDupDet algorithm[13] and proposed technique for duplicate detection. The precision, Recall and Computation plots are shown in Figure 2, 3 and 4 respectively.

It can be viewed from the Precision and Recall plots that the proposed technique is better and accurate in identifying near duplicates. Also time incurred for identifying near duplicates has been significantly reduced by using the proposed technique (as shown in Fig 4) since the new web page have to be compared with less number of web documents rather than all web documents in the database. Hence the proposed technique outperforms in accuracy and incurs less time and disk space.

V. Conclusion

A new duplicate detection technique is being proposed based on two phase filtering techniques that are applied serially one after the other. The first phase of page level feature
comparison is fast enough as it involves comparing numerical values only and fingerprint technique is used for providing high precision results and fast computation with limited storage. The proposed technique is efficient enough and is able to provide effective, precise and less time consuming results.

VI. REFERENCES


