A Study of Information Retrieval by Lovins Stemmer and Pattern Matching Approach

Wai Yan Lin Zaw, Khin Mar Myo
University of Computer Studies, Mawlamyine
alynnnet14@gmail.com, kmmyo09@gmail.com

Abstract

Nowadays, the World Wide Web technology is developed and it is a very large, distributed digital information collection and information contents in web is implemented by the different format such as HTML, XML etc. The role of information retrieval over the Web is important because web search users can make the best decision into their society if they get precise information from the Web. But as a result of huge amount of information over the Web and its growth rate, it is difficult for web search user to extract useful information. Field of information retrieval (IR) is born and several IR systems are used on everyday by a wide variety of users. This paper takes the advantages of IR and Brute Force pattern matching algorithm in the development of Web Document Retrieval System. This paper firstly describes how to pass the text preprocessing steps for web documents. Secondly, indexing technique for this system is described. Finally, how to match the query terms and document terms is described.

1.Introduction

Nowadays, advanced technology called internet have been developed and spread out everywhere to apply information such as documents, sound, news and animation etc. All online application based on the usage of World Wide Web. The Web provides world wide information services where information is linked from site to site. Web search users extract their needed information by browsing web directories, crawl the links from site to site and using search engines. To overcome the difficulties of getting the information among the spread out information contents, there are many developed search engines such as Google, Yahoo, AltaVista, etc., to improve searching over the Web.

But there is having the limitation of using the services of the World Wide Web. Sometimes, web search users do not know where to get their needed information over the Web. To overcome the difficulties of getting the information among the spread out information contents, there are developed many techniques. Among them Information Retrieval (IR) is one of the useful techniques for retrieving relevant information.

The information over the Web is represented as web pages and is mostly written with Hypertext Markup Language (HTML). HTML has two essential features - hypertext and universality [9]. With hypertext feature, web designer can create web page with hyperlinks to visit one page to another. With universality feature, every computer can read web pages. As a simple web page contains not only information but also tags for creating web page, it is difficult to use application area directly such as query and answering. Therefore, a system is needed for applying such web pages in application area directly by computer system and provides required information from it to the user.

In this paper, we implement the searching part of web technology. To implement the searching process, we use IR preprocessing steps as parsing HTML layouts, tokenization, remove stoplist from the tokens, and stemming them for achieving clean tokens from web pages. Then, match the results of parsing information from IR preprocessing steps against with user input queries from user interface. The pattern matching approach is used for obtaining the precise matching results of web searching process.

This paper is organized as follows: in Section 2, we presented related works. Web search Information retrieval processes are described in Section 3. In Section 4, we proposed architecture design of this system for searching process. Conclusion and future directions are presented in Section 5.

2. Related work

This section summarized the related works concerning with our system.

According to [8], Database Management and SQL can only handle structure database rather than free-form text. Information Retrieve (IR), XML and

other data mining techniques can handle semi-structure and unstructured database.

Now IR is used not only in text mining but also in other domain area, such as Library and Information Science. And it concerned with effective categorization of human knowledge and concerned with citation analysis and bibliometrics (structure of information).

[4] Provide the investigated techniques for extracting data from HTML sites through the use of automatically generated wrapper. This technique is compare the HTML pages and generated a wrapper based on their similarities and differences.

[1]Presentation of a new probabilistic model of information retrieval is there. The assumption is that documents and queries are defined by an ordered sequence of single terms and shows that the new probabilistic interpretation of tf x idf term weighting might lead to better understanding of statistical ranking algorithms.

C. Platzer and S. Dustdar [1] proposed a Vector Space Search Engine for Web Services. Their system used *a Vector Space Search Engine* to index descriptions of already composed services and presented a novel distributed Web service search engine based on the Vector Space Model for information retrieval.

3. Background Theory

This section describes the related background theory concerns with our system. There are four sub categories in this section: (i) IR, (ii) Text preprocessing, (iii) stemming algorithm and (iv) pattern matching algorithm.

3.1 Information Retrieval

IR is a field that has been developing in parallel with database systems for many years. Unlike the field of database systems which has focused on query and transaction processing of structure data, information retrieval is concerned with the organization and retrieval of information from a large number of text based documents. Typical information retrieval systems include online library catalog system and document management system.

3.1.1 Methods of IR

According to [3], there are two methods for information retrieval, (i) Keyword-based and (ii) Similarity-based information retrieval. In Keyword-based, a document is represented by a string, which can be defined by a set of keywords. Similarity-based retrieval finds similar documents based on a set of common keywords. The output of such retrieval should be based on the degree of relevance, where relevance is measured based on the closeness of the keywords, the relative frequency of the keywords, and so on.

3.1.2 Models for IR

The classic models for IR are Boolean model (based on set theory), Vector space model (based on algebra), and Probabilistic models (based on probability theory). Further models are Browsing model, Filtering model; Fuzzy set model, Extended Boolean model, etc.

The problem with Information Retrieval is to locate relevant documents based on user input queries. Web Information Provider considered the measurement qualities of whether the retrieved document is relevant or not.

There are two basic measures of assessing the quality of text documents: Precision and Recall. Let P be the set of documents relevant to user input queries and Q be the set of retrieved documents. Thus, the measurement of Precision and Recall is as follow:

- Precision = { P ∩ Q } / Q
- Recall = { P ∩ Q } / P

3.2 Text Preprocessing for Web Document

Text preprocessing steps are HTML's layouts parsing for strip unwanted markup from web pages such as HTML tags, attributes, (CSS) properties etc. *Tokenization* breaks the sentence into tokens (keywords) on white space, *Stop-words Removing* removes the unnecessary and ineffective words for searching, *Stemming* stems the tokens to the original words (root-word) for matching effectiveness, for example from " used, using" to "use" etc.

Stemming algorithm has the advantage of reducing the corpus size thus making information retrieval a faster process. There are a number of stemmers available, notably the Lovins stemmer [5], Paice/Husk stemmer [6] and the Porter stemmer [7]

In our system design, Lovins stemming algorithm is used to enhance the stemming process of text preprocessing. The following sections describe the Lovins Algorithm in brief.

.

3.3 Lovins Stemming Algorithm

The Lovins stemmer has 294 endings, 29 conditions and 35 transformation rules. Each ending is associated with one of the conditions. In the first step, the longest ending is found which satisfies its associated condition, and is removed. In the second step, the 35 transform rules are applied to transform the ending. The second step is done whether or not an ending is removed in the first step.

3.4 Exact String or Pattern Matching Algorithms

String or pattern matching is a very important role in many application areas such as text mining, information retrieval, information extraction, pattern recognition, biometrics, error detection, etc. Another recognizing certain patterns within a text or sequence of symbols (might be a DNA, RNA or protein sequence), using so called exact pattern matching or string matching algorithms. Many pattern matching algorithms are used for pattern recognition from the random information such as Brute Force, Boyer-Moore, and Knuth-Morris-Pratt (KMP) etc.

3.4.1 Brute-Force Algorithm

The brute-force pattern matching algorithm compares the pattern ***P*** with the text ***T*** for each possible shift of ***P*** relative to ***T***, until either a match is found or all placements of the pattern have been tried.

In Brute-Force pattern matching algorithm, if the pattern is matched with the text token, then the sequence of pattern and text token is shifted one from left to right. When a mismatch is found after the numbers of subsequence match, realign the input text sequence against the pattern.

4. Design of Proposed System

The system design is constructed based on IR, Stemming Algorithm and Pattern matching algorithm. There are two main steps in system architecture as illustrated in Figure 1. These two main steps are (i) Document preprocessing steps and (ii) Matching step.

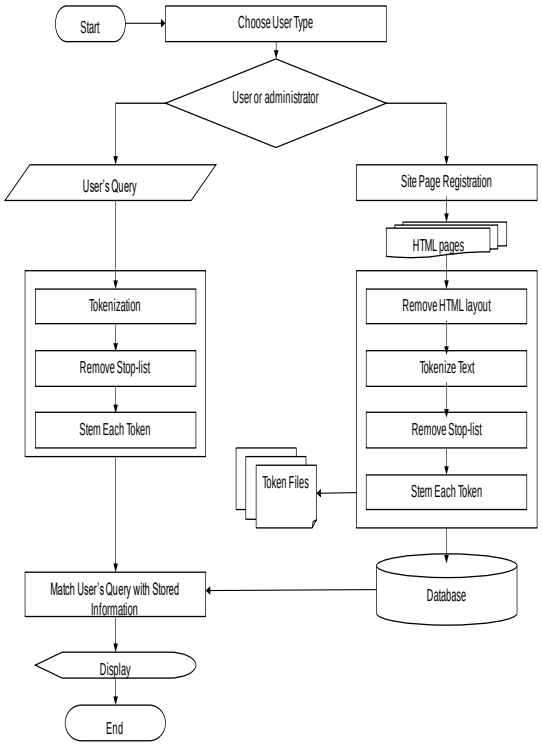


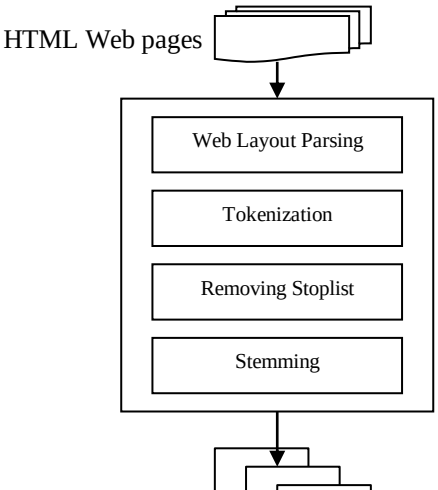
Figure1. System Design for information Retrieval

5. System Implementation

To evaluate the effectiveness of our system design, Web Document Retrieval system is implemented. The following section describes the detail of system implementation.

5.1. Web Document Preprocessing

The input of our document retrieval system is HTML web pages. Firstly the system is needed to parse the HTML pages to separate the necessary part and unnecessary part. After parsing and removing the unnecessary part from HTML content, the remaining content is passed to another process for tokenizing. Therefore preprocessing step contains sub processes of (i) HTML Layouts Parsing, (ii) Tokenizing, (iii) Stop words Removing and (iv) Stemming which is illustrated in Figure 2.



Clean Tokens

Figure2. Information Retrieval preprocessing document workflows

(i) HTML Layout Parsing

For HTML parsing, the pages of HTML documents are retrieved from some input of resources collection from the application source of the World Wide Web. And then analyze the structure of HTML pages and must plan to consider how to parse them by removing the unnecessary parts of HTML pages.

Firstly, non-essential parts from HTML pages are removed. The following table shows the non-essential parts from HTML pages.

Table 1: Non-Essential HTML's Information

Unnecessary parts of HTML	Description
Tags(elements)	!doctype, a, applet, area, input, code, frame, font, form, head, title…
Attributes	Width, height, color, align, cols, Content, behavior, direction, loop, language,…
CSS(properties and values)	Display, float, font-family, font-size, letter-spacing, list-style, margin…
Intrinsic Events	Onfocus, onkeypress, onload, onmouseout, onrest, Onselect, onsubmit, onunload…

As an example, the following sample source code of HTML page is used to remove the HTML layout as shown in Figure 3.

```
<div id="pp-cont">
 <h1 id="product-name">
 <span class="main-title">Brilliant Home <b>Computer</b> Book: Windows Vista Edition</span>
 <span class="sec-title">by P. K. Macbridge in Books
 <td>
 <div id="product-image-cont">
 <img src="c5_files/books.jpg" alt="Brilliant Home Computer Book: Windows Vista Edition" title="Brilliant Home Computer Book: Windows Vista Edition">
 </div>
 </td>
 <td id="product-price-rating-desc">
 <p class="product-price-cont">
 <span class="product-price">
 <span class="main-price">$23</span> new,&amp;nbsp;<span class="sec-price">$62</span> used
 </span>
 from
 <a>href="http://www.google.com/products/catalog?q=computer++books&amp;cid=14385913547954432147&amp;os=sellers#p"id="link-to-ps-sellers" onmousedown=
 'logTabClick("14385913547954432147","ps-sellers");'>12 sellers</a> </p>
 <div class="product-desc-cont"><p class="product-attr">
 By P. K. Macbridge - Pearson Prentice-Hall Prof&amp;nbsp;(2008) - Hardback - <nobr>320 pages</nobr> - <nobr>ISBN 0273715739</nobr> </p>
 <p class="product-desc">
 The ONLY book that you will need to show you how to do just about
 anything that you might want to do on your home computer – from browsing the internet and organising yur emails…...</u.></body></html>
```

Figure 3. Example Source Code for HTML page

Eliminating HTML layouts or unnecessary parts of HTML pages such as tags, attributes etc…. are not relevance for query matching. After parsing this HTML page, the output of information token is as follow:

Brilliant Home Computer Book: Windows Vista Edition Brilliant Home Computer Book: Windows Vista Edition $62 used from By P. K. Macbridge - Pearson Prentice-Hall Prof ;(2008) Hardback - 320 pages ISBN 0273715739 The ONLY book that you will need to show you how to do just about anything that you might want to do on your home computer – from browsing the internet and organising yur emails…..

Figure4. The resultant text from HTML layout parsing

(ii)Tokenization

Then, the system will tokenize that text information to improve matching effectiveness.

brilliant home computer book: windows vista editionb home computer book: windows vista

edition $62 used from by p. k. macbridge - pearson prentice-hall prof

;(2008) hardback - isbn 0273715739 the only book that you will need to show you how to do just about anything that you might want to do on your home computer – from browsing the internet and organising yur emails…..

Figure5. The resultant text token from Tokenization

(iii)Stoplist removing

And then removing stop-words from the output of information token, the clean information is obtained.

brilliant home computer book: windows vista editionb home computer book: windows vista edition $62 used p. k. macbridge - pearson prentice-hall prof

;(2008) hardback - isbn 0273715739 book need home computer –browsing internet organising emails…..

Figure6. The resultant text token from Stoplist removing

(iv)Stemming the tokens

Then, stem the information tokens or words from the remaining tokens to improve the effectiveness of information searching.

brilliant home computer book: window vist edit b home computer book: windows vist edition $62 us p. k. macbridg - pearson prenticehal prof

2008 hardback isbn 0273715739 book need home computer –browsing internet organ email

Figure7. The resultant clean token text from stemming

5.2 Web Document Retrieving

To retrieve the relevant pages from web application source, this system accepts the input query term from user via user interface.

For example the user input query term is "**Brilliant , the computer, Book** ", the query term is also need to conduct the preprocessing phase as explain in Section 5.1, except from parsing and layout removing step. After passing the preprocessing, the query term get three tokens as "**brilliant, computer, book** ".

After the input query is ready for retrieving their relevant pages, the tokens in the token text files is read from the application source. As an example, the token text files are shown in figure in Figure 8.

brilliant home computer book: window vist edit b home computer book: windows vist edition $62 us p. k. macbridg - pearson prenticehal prof

2008 hardback isbn 0273715739 book need home computer –browsing internet organ email

Figure8. The resultant token information

The tokens getting from existing document as shown in Figure 8 are structured with array representation as follows in Figure 9.

TokenArray={brilliant, **home**, **computer**, **book**, **window**, vist, edit, **home**, **computer**, **book**, **window**, vist, edition, etc}

brillia nt	home	comp uter	book	windo w	visit	visit

Figure9. Tokens Array

To be efficiently matching process, this array representation is sorted alphabetically and removing duplicate words. This is shown in Figure 10.

brillia nt	book	comp uter	home	windo w	visit	

Figure10. Tokens Array (After sorting and remove duplication)

And then this string word of array representation is put into the Hash map alphabetically.

With Hash Map Indexing, the query matching process is speed up and more accurately. This hash map indexing design is shown in Figure 11.

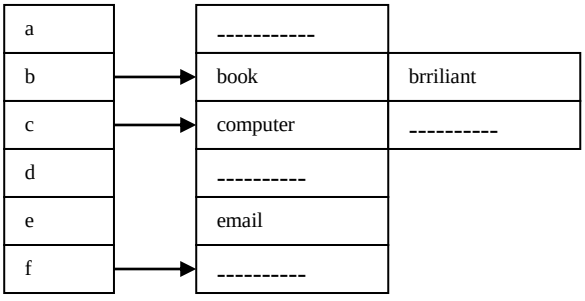


Figure11. Hash Map Indexing Design

After the indexing process is done, the last process for retrieving relevant pages is conducted. This retrieving process design is shown in Figure 12.

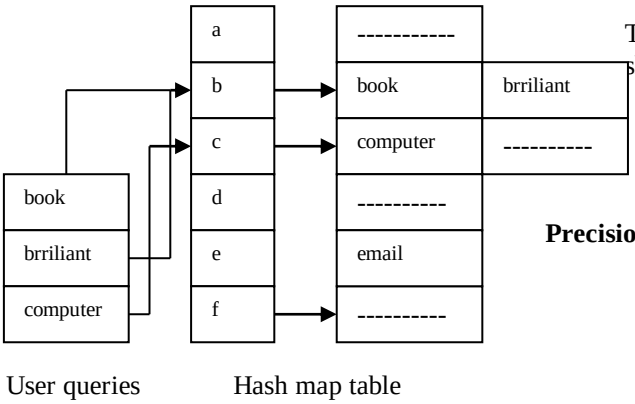


Figure12.Searching process of the system

As describe in Figure 6, the query terms contain 3 tokens. These 3 tokens are matched with existing document terms in hash base index table by using the Brute Force pattern matching algorithm.

After the matching process is complete, the relevant site URL is give to user.

6. Performance Measure

In this section, the performance of our system is evaluated. We first compute the precision and recall for this system. To access the "accuracy" or "correctness" of the system, there are two measures of IR success, both based on the concept of relevance [to a given query or information need], are widely used: "precision" and "recall.

The precision and recall rate is measured by using the formula as described in Section 3.1.2 using the different length of Query terms. For example, the query term is "Computer", "Knowledge of Computer", "Networking and Computer Windows Administration ", " Data mining approach Basic for Master Students " , " Accuracy Measurement for Information Retrieval by Precision and Recall Concepts". The results show that of the query term is short, the recall rate is high and precision rate is low. If the query is too long, the recall rate is low and precision rate is high. The results for precision and recall are described in Table 2.

Table 2. Precision and Recall

No of token in query	1	2	4	5	7		
Recall	0.9	0.8	0.7	0.5	0.2		
Precision	0.2	0.5	0.7	0.8	1.0		

The Table is also described with line graph as shown in Figure 13.

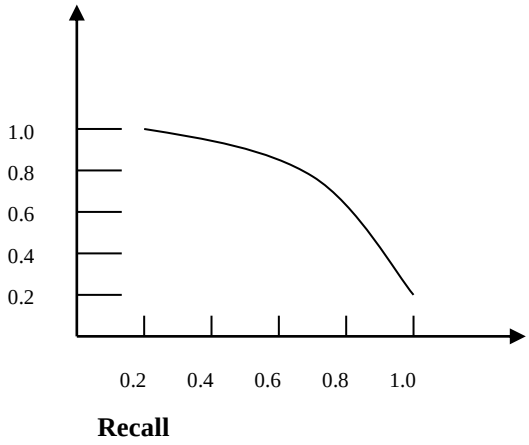


Figure13.Line graphs for accuracy measurements

7. Conclusion

In this paper, we described the information retrieval system is intended to give more precise information to web search user by using pattern matching method. In this paper we describe the detail implementation web document retrieval system. There are two main parts in our system: (i) preprocessing and indexing and (ii) retrieving. In indexing phase, this system uses the hash-based indexing to speed up the searching time instead of other indexing approach. Beside these, Lovins Algorithm is used to improve the effectiveness of retrieving. The Brute Force Pattern matching is also used to match the query term and existing document terms. The experiment results show that our document retrieval system can search accurately and efficiently. Thus, web users can get their needed information by spending a little time over the web.

References

[1] R. Baeza-Yates and B. Ribeiro-Net," Modern Information Retrieval," ACM Press, AddisonWesley, 1999.

[2] S. Brin and L. Page, "The Anatomy of a Large-Scale Hypertextual Web Search Engine", Elsevier Science Publishers B. V. Publisher

[3]J.iawei Han and M. Kamber, "*Data Mining Concepts and Techniques*", Academic Press, A Harcourt Science and Technology Company http://www.academicpress.com .

[4] K.Kaiser and S.Miksch,"Information Extraction".Vienna University of Technology.

[5] J. Lovins, "Development of a Stemming Algorithm".*Mechanical Translation and Computational Linguistics", (*1-2), 11-31, 1968

[6 E. lizabeth Castro, "*HTML For The World Wide Web*", Visual Quickstart Guide (Fifth Edition), Copyright © 2003 by Elizabeth Castro.

[7] M. Porter," An algorithm for suffix stripping. *Program ", 14*(3)-130-137, July 1980
F

[8]C. Platzer and S. Dustdar, "A Vector Space Search Engine for Web Services" Proceedings of the Third European Conference on Web Services (ECOWS'05) 0-7695-2484-2/05 © 2005 IEEE

[9] Paice, C., Husk, G.," Another Stemmer", *ACM SIGIR orum 24(3):566:1990*