



Information Retrieval System Using Binary Search Tree and Indexing Approach

Aarti Sharma¹, Vishal Shrivatava²

^{1,2}Department of Computer Engineering & Information Technology,
Arya College of Engineering & I.T. (ACEIT), Kukas, Jaipur, Rajasthan, INDIA

Abstract: As the time grows forms of storage changes and information retrieval become more complex as medium of storage changes a lot. There is huge amount of information on computer hard disks or storage devices etc. So, now to retrieve data our key issue is achieving efficient information retrieval developed to search queries of people to ensure delivery of qualitative data in response of search. As we have various methods for effective retrieval of information from document provided on computer and on internet.

The System which provides access of quality information from any method of searching is called Retrieval System. The proposed system provides information with the help of index. Index is generated with the help of keyword (Words) from information documents and by searching Index we can search relevant information accurately and fast according to need. In this information retrieval approach we can use various algorithms to work effectively to give results of queries by people.

Index Terms: Index, Information Retrieval.

I. INTRODUCTION

As Internet is a vast source of information and to harness this information various information retrieval systems are used. These systems can range from a classical information system like library system to a web based system like search engines. The effective retrieval is developed by search mechanism that would examine precision according to need. Information retrieval community employ keyword based searching and retrieves data in response of query from document that contain user specific keywords. This online information retrieval is called Web mining.

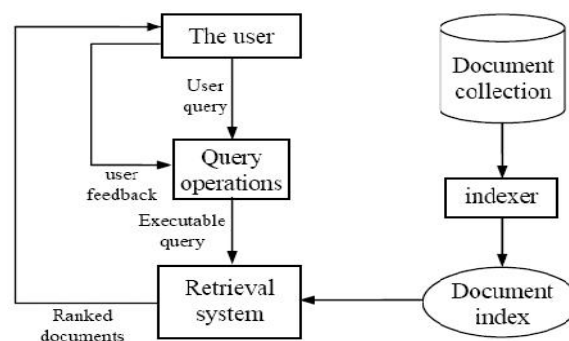


Figure 1: Information Retrieval System

II.

III. LITERATURE SURVEY

A. What is Information Retrieval:[9]

The process of actively seeking out information relevant to a topic of interest is called information retrieval. It refers to the automatic (rather than manual) retrieval of documents like Information Retrieval System (IRS). Whereas "Document" is the generic term for an information holder (book, chapter, article, webpage, etc).

Data retrieval involves the selection of a fixed set of data based on a well-defined query (e.g SQL, OQL, and many more). Whereas Information retrieval (IR) involves the retrieval of documents of natural language.

IR deals with the problem of finding and presenting documents of unstructured data that satisfy information need from within collections of documents. IR deals with the problem of finding and presenting documents of unstructured data that satisfy information need from within collections of documents.

A user information need, also referred to as query, must be 'translated' in order for an IR system to process it and retrieve information relevant to its topic.

B. The Retrieval Process[9]

The information retrieval system must have to do at least three different processes as illustrated in figure 2.

- Representing the content of documents.
- Representing a user's information need.
- Comparing or matching both representations.

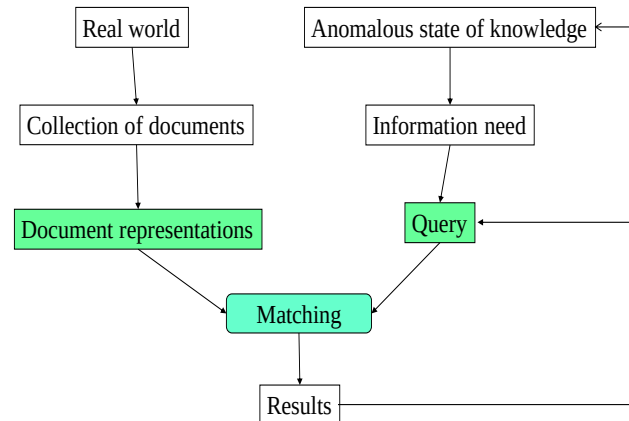


Figure 2: Information Retrieval process[9]

A retrieval model predicts and explains what a user will find relevant given the user query.

C. Types of Information Retrieval Process:

There are various types of information retrieval process:

- **Keyword based information retrieval:[2]**

A distributed information retrieval framework which is based on the Okapi probabilistic model. This is the enhanced probabilistic retrieval model based on binary independence model. Here used a simplified Okapi weighting function, in which w_1 was assigned to a given term t_1 in a document d and was computed according to the following formula:

$$\text{where } w_i = (k_1 + 1) \cdot \frac{tf_i}{K + tf_i}$$

$$K = k \cdot (1 - b) + b \cdot \frac{l}{avdl}$$

l is the document length,

$avdl$ is the average of document length.

b, k, k_1 are constants.

tf_i is the occurrence frequency of term t in document d .

The following formula shows the weight given to the term t_i in document d .

$$qw_i = \frac{qtf_i}{k_3 + qtf_i df_i} \cdot \log \left[\frac{n - df_i}{df_i} \right]$$

- **Attention based information retrieval:[4]**

In Attention based IR, here an eye tracker is used as the feedback observer. It observes about the user that which part of the document is read or skimmed by the user. Information generated by eye tracker is stored in an attention annotated documents (ex Passages which are commented, or highlighted) with the help of these annotated documents, estimations on user's current thematic context can be implicitly or explicitly generated. Challenges with this system are to know the reaction of the user about the information provided. This can be found out by taking explicit feedback from the users. By taking into account such a feedback relevancy of the document increases. This requires high efforts from users side thus can be avoided.

Thus there is a focus on implicit feedback. Here eye tracker can observe the movement of eyes very precisely and thus provides a feedback through user's eye movement as it can directly reflects the user's attention.

- **Ontology based information retrieval:[3][5]**

In ontology there developed package with three different approaches to document retrieval:

- Vector representation approach.
- Latent semantic indexing method (LSI)
- Ontology-based approach.

A system is designed for gathering the information for such tasks and the system developed is web based Webocrat system. Webocrat can interact at the knowledge level with the help of language. This language has been provided to the system by ontology with syntax and semantic rules.

Use of ontology enables to define concepts and relations representing knowledge about a particular document in domain specific terms. So, to express the content of document explicitly, it is necessary to create link between document and relevant part of domain model. Such approach support techniques where search engine can utilize domain model to find relevant data related to user specification.

- **Index Based information retrieval[1]**

According to the relevancy whole or part of the document is provided. This comparison can be done in the form of matrix multiplication. For this purpose all the documents are given a id and similarly ids are given to the queries and matrix of both are generated separately, After this matrix multiplication can be done and the necessary result has been generated. Then with the help of this result it can be found out which document is more relevant to the query.

Mathematically it can be shown as:

Consider there are 2 documents (i and j) represented as:

Doc (i) = (Term (i1), Term(i2).....Term(ik)
 Doc (j) = (Term (j1), Term(j2).....Term(jl))

Where k and l are no. of terms in respective documents.

So, all terms for all documents together can be represented as =

[Term (i1), Term (i2).....Term (ik) U (Term (j1), Term (j2).....Term (jl)]
 - [Term (i1), Term (i2).....Term (ik) ∩ (Term (j1), Term (j2).....Term (jl)]

= [(Term (1), Term (2).....Term (n)]

i.e. Term(1)....Term(n)= all distinct terms of both documents i and j.

By using this searching time is also reduces effectively.

IV. PROPOSED WORK

A. Context Based Indexing:

The architecture of the proposed work based on context based indexing in IR system is shown in the following figure 3.

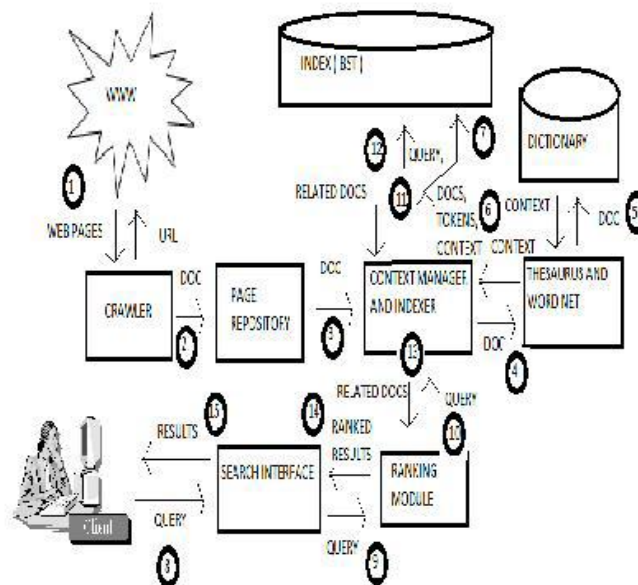


Figure 3: Architecture of context based indexing

Use context based Indexing in IR System to preprocess the documents from web pages and separate out tokens. According to context based indexing architecture the Web Crawler is used to copy the web pages

accessed and later provided for fast search. These pages are passed through context manager and indexer where the pre-processing of the documents are done. At the next level these documents are ranked according to the frequency of the words contained by the document. Higher the frequency of words in the document higher will be the rank.

B. Binary Search Tree:

It is a finite set of nodes that is either empty or consists of a root and two disjoint subsets called left and right sub-trees. Binary trees are used to implement binary search algorithm for the faster data access and most relevant data access.

Indexed structure for binary search tree

Structure used to store the documents is shown in the figure 4.

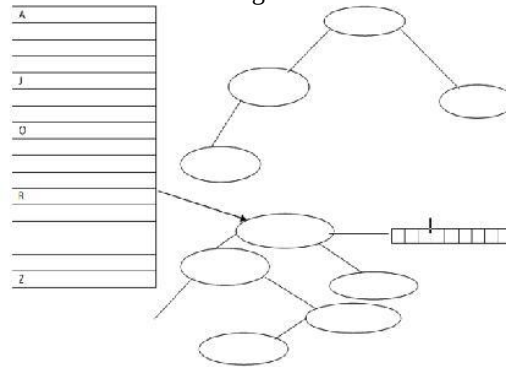


Figure 4. Data Structure used to store document keywords.

BST will finally store the stem word and generate list of files as preorder in form of nodes of BST, with the no. of frequency of occurrence of the words which having similar meaning including the stem word. According to this frequency we decide the file is more relevant to the query searched. Thus, the list of most relevant files according to the root node is provided to the user according to query searched.

C. Porter Algorithm: [6]

It is a process for removing the common morphological endings from words in English. Its main use is to normalize the process of information retrieval process. This algorithm separate out the stem word from the similar meaning words automatically.

D. Hits Algorithm: [7]

HITS algorithm identifies two different forms of Web pages called hubs and authorities.

In HITS algorithm, ranking of the web page is decided by analyzing their textual contents against a given query. After collection of the web pages, the HITS algorithm concentrates on the structure of the web only, neglecting their textual contents.

In this HITS algorithm, the hub and authority are calculated using the following algorithm.

1. Initialize all weights to 1
2. Repeat until the weights converge:
3. For every hub $p \in H$
4. $H_p = \sum A_q$ where $q \in I_p$
5. For every authority $p \in A$
6. $A_p = \sum H_q$ where $q \in B_p$
7. Normalize

This concept of hits algorithm to find frequency of related words according to stem word generated present inside the file to find the most relevant files present in the corpus.

E. Proposed algorithm:

The algorithm shows the various steps to construct the context based searching of keywords.

Step1: The preprocessing of the documents which includes stemming and removing stop words is performed by indexer.

Step 2: Parse the document using parser and thus generates the list of tokens (words).

Step 3: Create a BST

Step 4: Store Tokens in the BST.

Step 5: Convert BST into pre order BST.

Step 6: Query entered by user, from these query keywords are extracted.

Step 7: Context are made available to the user.

Step 8: Search begins for the query as according to the context has been chosen.

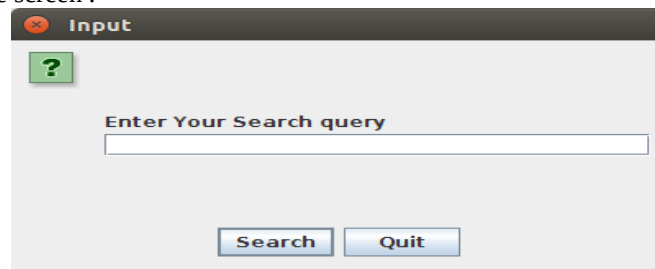
Step 9: Result has been generated in the form of document names.

Step 10: Frequency of the keyword, found in the documents is also given along with document names.

V. IMLEMENTATION

The chronological order of the working of the proposed work:

- **Query:** The user will enter a query, over here I have used swing to develop its User Interface.
- **Tokenizer:** The query entered by the user will get converted into tokens using Tokenizer.
- **Stemmer:** Here, converted token will be go through stemmer algorithm used over here which will find the stem word from the token. Token represents the theme of the document. Different word with the same root doesn't present any extra information about theme. For example the words 'play', 'playing', 'plays' and 'played', all are with same root 'play' and all of them present the same theme 'to play'. Play will be the stem word here extracted as root node.
- **File Search:** the stem words that we get from the porter algorithm will be used to searched through the files using parsing and Hits Algorithm.
- **Binary Search Tree:** the result that we will ,will be get stored in the Binary Search Tree and then get displayed on the screen .

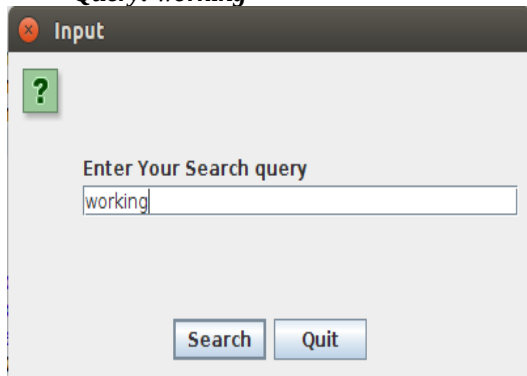


VI. RESULT AND ANALYSIS

The results has been compared for precision value for the list generated without using BST in IR system and with using BST in IR system including indexing approach . Precision is the fraction of the documents retrieved that are relevant to the user's information need.

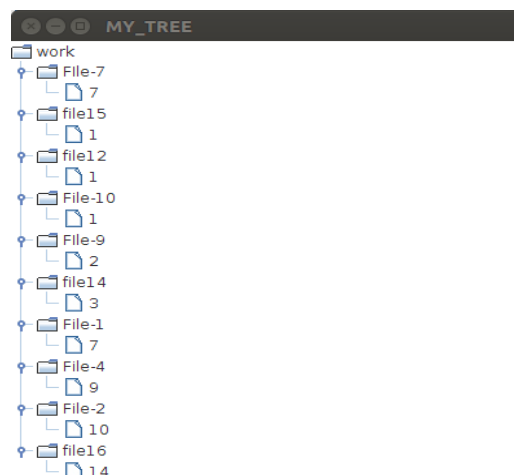
Snapshots of some different query results:

A. Query: working



```
MainHandler [Java Application] /usr/lib/jvm/java-7-oracle/bin/java :
Enter your Query :
working
File-7 : 7
File-9 : 2
File-1 : 7
File-10 : 1
File-2 : 10
File-4 : 9
file12 : 1
file14 : 3
file15 : 1
file16 : 14
Result After BST application:
File-7.txt : 7
File-1.txt : 7
File-2.txt : 10
File-4.txt : 9
file16.txt : 14
entered into traverse
7
File-7
1 file15
1 file12
1 File-10
2 File-9
3 file14
7 File-1
9 File-4
10 File-2
14 file16
```

Binary Tree:



Result Analysis: We use the following formula for result analysis:

$$\text{Precision} = (|\{\text{Relevant Documents}\} \cap \{\text{Retrieved Documents}\}|) / |\{\text{Retrieved Documents}\}|$$

Without BST implementation:

Relevant Documents = 5

Retrieved Documents = 10

$$(|\{\text{Relevant Documents}\} \cap \{\text{Retrieved Documents}\}|) = 5$$

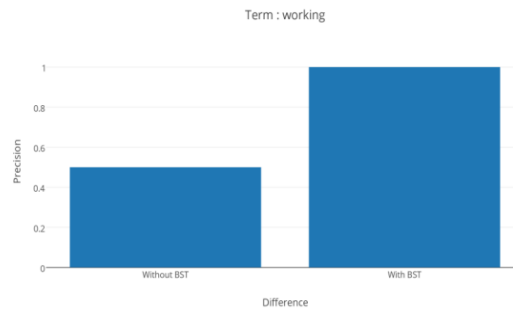
$$\text{Precision} = 5/10=0.5$$

With BST implementation:

Relevant Documents = 5

Retrieved Documents = 5

$$\text{Precision} = 5/5=1.0$$



Graphical representation of result comparison

VII. CONCLUSION

In this paper keywords are indexed along with their context which is extracted from the documents. Here Binary Search Tree is used to store the keywords for improving the performance of the retrieval system in terms of accuracy and efficiency for retrieving more, relevant documents as per the requirements of the user as context of the keywords are also provided. Thus using index for searching purposes provides fast access to document context structure along with an optional searching. This implementation shows that with using BST along with indexing approach in context based indexing gives use more relevant results faster than tradition approach.

REFERENCES

- [1] Ambesh Negi¹, Mayurbhirud, Dr. Suresh Jain, Mr.Amit Mittal, "index based information retrieval system", international journal of modern engineering research (ijmer),www.ijmer.com vol.2, issue.3, issn:2249-6645 , may-june 2012.
- [2]. Yves Rasolofo and Jacques Savoy, "term proximity scoring for keyword-based retrieval systems", published in lecture notes in computer science 2633, 1611-3349, 2003.
- [3]. Jan Paralac, Jvan Kostial , "ontology-based information retrieval". webocracy project, "knowledge technologies for information acquisition and retrieval" of scientific grant agency of ministry of education of the slovak republic, april 2011.
- [4]. Georg Buscher, "attention-based information retrieval", sigir conference doctoral consortium , 2007.
- [5]. Sofia Stamou, "retrieval effectiveness of an ontology based model for conceptual indexing", 26500 greece, 2005.
- [6] Dilip kumar Sharma , "a comparative analysis of web page ranking algorithms" , (ijcse) international journal on computer science and engineering, vol. 02, no. 08,2010, 2670-2676.
- [7] <http://www.bui.hawhamburg.de/pers/ursula.schulz/astep/porter.pdf>
- [8] Laxmi Choudhary and Bhawani Shankar Burdak,"role of ranking algorithms for information retrieval", published in international journal of artificial intelligence & applications (ijaia), vol.3, no.4, july 2012.
- [9] Hiemstra. *Using language models for information retrieval*. Ph.D. Thesis, Centre for Telematics and Information Technology, 2001.