International Journal of Computer Sciences and Engineering Open Access
Review Paper Volume-2, Issue-11 E-ISSN: 2347-2693

A Review on Document Retrieval from Unstructured Text

Sneha Lohbare[1]* and Ashwini Meshram[2]
[1,2]*Department of Computer science & Engineering, G.H.R.A.E.T, Nagpur University, India.*

www.ijcaonline.org

Received: 24Oct 2014 Revised: 08 Nov 2014 Accepted: 22 Nov 2014 Published: 30 Nov 2014

Abstract— A simple search over a document can be considered as a traditional method of searching from a single document in database. A keyword or string is considered as core element while searching where string may be strings of words, characters for any phrase. Many problems in such keyword or phrase-based searching arise when a keyword or phrase is intended to be searched in multiple documents. For the same, a solution suggested is a repetitive procedure of searching for every document. It can be helpful for limited number of copies of document. But this solution can never be considered efficient and effective in case of large number of documents in database which is supposed to be increasing continuously. Also, searching for the pattern based or the regular expression based content from the document is one of the demanding topics of research. Processing such queries requires a lot of processing time and complete indexing of data is bit difficult process.

Keywords— Document retrieval, Indexing, Unstructured Text

I. INTRODUCTION

The basic function of communication is transferring the data from one corner to another corner of the world. The data is basically stored in the form of documents, files and these files are arranged under folder or subfolder. The random creation and storage makes them unstructured in nature which results in inefficient data retrieval and modification as well as updation. E-commerce and corporate intranets has led to the growth of organizational repositories containing large and unstructured document collection. So, efficient storage and transmission of documents as well as archiving and information retrieval for document databases have become important research issues. Structured documents must maintain the structure where in addition to pure textual information, the meaning of different sections likes author, title, abstract, heading of section or subsection, paragraph, etc. are also stored within same document whereas unstructured document does not have a predefined manner. Unstructured information is basically text-heavy, also contain data such as dates, numbers, facts [1].

As the size of unstructured data in our world continues to increase, text mining tools that allow to sift through this information with ease will become more and more valuable. Text mining tools are beginning to be readily applied in the biomedical field, where the volume of information on a particular topic makes it impossible for a researcher to cover all the material, much less explore related texts[2]. Text mining methods can also be used by the government's intelligence and security agencies to try to piece together terrorist warnings and other security threats before they occur. Another area that is already benefiting from text mining tools is education. Students and educators can find more information relating to their topics at faster speeds than they can use traditional adhoc searching. The new developments in text mining technology that go beyond simple searching methods are the key to information discovery and have a promising outlook for application in all areas of work[2][3].

In current research, Information Retrieval (IR) methods, such as text indexing (TI), have been developed to handle the unstructured documents. But these IR methods become ineffective for continuously increasing large amount of text data. On the other hand, small fraction of this large text data is only relevant to user in order to analyze and extract useful information from data. Thus, Text Mining has become more and more popular and essential topic in DM. Text Mining, also known as knowledge discovery (KD) from text, and document information mining (IM), refers to the procedure of extracting fascinating information from very large text quantity for the purposes of determining knowledge[4]. It is an interdisciplinary field involving IR, understanding text, extraction of information, clustering, classification, linkage of concept, visualization, database knowledge, machine learning (ML), and DM. Search engine is the most well known Information Retrieval tool. Application of Text Mining techniques to Information Retrieval can improve the precision of retrieval systems by filtering relevant documents for the given search query[4].

II. DOCUMENT RETRIEVAL AND TEXT MINING

Document Retrieval is defined as the matching of some user stated query against a set of free-text records. The main goal of information retrieval system (IRS) is to "finding relevant information or a document that satisfies user information needs [5].

A. Document Retrieval

There are two basic document retrieval processes. First is browsing or navigation system and another is classical IR system. In former system, User skims document collection by jumping from one document to the other via hypertext or

Corresponding Author: *Sneha Lohbare, snehalohbare@gmail.com*

hypermedia links until relevant document found. In later system, also called as question answering system, query-answer format is implemented to retrieve document or specific information. A query i.e. question in natural language is placed and answer is directly extracted from text of document collection [6].

Document Retrieval system consists of a database of documents, a classification algorithm to build a full text index, and a user interface to access the database. The system finds information to given criteria by matching text records (documents) against user queries, as opposed to expert systems that answer questions by inferring over a logical knowledge database. There are two main classes of indexing schemata for document retrieval systems are form based (or word based),and content based indexing. The document classification scheme (or indexing algorithm) in use determines the nature of the document retrieval system. Form based document retrieval addresses the exact syntactic properties of a text, comparable to substring matching in string searches. A suffix tree algorithm is an example for form based indexing. The content based approach exploits semantic connections between documents and parts thereof, and semantic connections between queries and documents. Most content based document retrieval systems use an inverted index algorithm [6].

Traditional document storage systems allow users to identify documents using metadata such as author, title, keywords. The basic idea is to index every individual word in the document collection. Effectively, documents are represented as a "bag of words"—that is, the set of words that they contain, along with a count of how often each one appears in the document. Many practical systems discard common words or "stop words", primarily for efficiency reasons. A query is expressed as a set, or perhaps a Boolean combination, of words and phrases, and the index is consulted for each word in the query. A well-developed technology of relevance ranking allows the salience of each term to be assessed relative to the document collection as a whole, and also relative to each document that contains it. These measures are combined to give an overall ranking of the relevance of each document to the query, and documents are presented in relevance order [7].

B. *Indexing*

Information Retrieval systems must cope with at least three different processes

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

The process of representing the content of documents is also called indexing. This task involves deriving which parts and keywords are extracted from the documents in the collection. Other approaches are stemming and stopwords removal [8] [9][13]. Stemming refers to the conflation of words to their lemmatized base form, where the morphological variants of words are stripped to one single suffix entry. Stopwords removal refers to eliminating very frequent terms, like the, of, a, from the indexing process, as they are likely a good indicator of poor content [13]. Ruling out these words from the index can result in significant space savings, while not hurting retrieval performance.

C. *Indexing*

The set of documents relevant to a query be denoted as {Relevant}, and the set of documents retrieved be denoted as {Retrieved}. The set of documents that are both relevant and retrieved is denoted as {Relevant} ∩ {Retrieved}.There are two basic measures for assessing the quality of text retrieval.

Precision: This is the percentage of retrieved documents that are in fact relevant to the query. It is formally defined as [10][11].

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

Recall: This is the percentage of documents that are relevant to the query and were, in fact, retrieved. It is formally defined as

$$\text{Recall} = | \{\text{Relevant}\} \cap \{\text{Retrieved}\} | / | \{\text{Relevant}\} |$$

An information retrieval system often needs to tradeoff recall for precision or vice versa. One commonly used tradeoff is the F-score, which is defined as the harmonic mean of recall and precision

$$\text{F-score} = \text{recall} \times \text{precision} / (\text{recall} + \text{precision}) / 2$$

The harmonic mean discourages a system that sacrifices one measure for another too drastically precision, recall, and F-score are the basic measures of a retrieved set of documents. These three measures are not directly useful for comparing two ranked lists of documents because they are not sensitive to the internal ranking of the documents in a retrieved set [10].

In the beginning, Information Retrieval (IR) provided many term-based methods to solve this challenge, such as Rocchio and probabilistic models, rough set models, BM25 and support vector machine (SVM) based filtering models. The advantages of term based methods include efficient computational performance as well as mature theories for term weighting, which have emerged over the last couple of decades from the IR and machine learning communities. However, term based methods suffer from the problems of polysemy and synonymy, where polysemy means a word has multiple meanings, and synonymy is multiple words having the same meaning. The semantic meaning of many discovered terms is uncertain for answering what users want [12].

III. RELATED WORK

The following sections explain the survey of various papers. Different methods are used for extracting text information from the given set of data by many researchers.

K.Sreerama Murthy, Dr G. Samuel Varaprasad Raju, Dr. C. Sunil Kumar discussed some problems in conventional IR methods which are mentioned with proposed solutions. Conventional IR methods become insufficient to handle large Text Databases containing high quantity of text documents. To explore relevant documents from the large document collection, inverted files are used which are then read from the disk. The major cost of searching process are the space requirement in memory to hold inverted file entries, and the time spend to process huge size inverted files maintaining record of each document of the quantity as they are potential solutions. To speed up the procedure of text document retrieval and effective use of the memory space, K.Sreerama Murthy et al. proposed an algorithm which is based on inverted index file. By using the range partition feature of oracle, the space requirement of memory is condensed considerably as the inverted index file is stored on secondary storage and only the required portion of the inverted index file is maintained in the main memory [4].

B. Ganga, in the process of document retrieval, focused on work to combine the advantages of two methodologies: suffix tree and Boyer-Moore. As a result the Phrase Based Document retrieval Algorithm represent each document as suffix trees, where phrases of the documents are stored as suffixes of tree in hash table for efficient storing and retrieval and Boyer Moore algorithm is used to check the presence of pattern i.e. the input phrase in order and without order. General description of the algorithm is, it takes input as set of documents in the form of Portable Document Format, MS Format files and Text files and input phrase and implements suffix tree algorithm which represent document as suffix and then Boyer-Moore algorithm is applied and output is obtained as the set of documents which contains the input phrase in order as well as without order [6].

Ning Zhong, et al. put forward the somewhat negative part of phrase-based approach of searching. Although phrases are less ambiguous and more discriminative than individual terms, the likely reasons for the discouraging performance include:

- Phrases have inferior statistical properties to terms,
- They have low frequency of occurrence, and
- There are large numbers of redundant and noisy phrases among them [11].

R. Sagayam, S.Srinivasan, S. Roshni mention two indexing techniques which are inverted indices and signature files. An inverted index is an index structure that maintains two hash indexed or B+-tree indexed tables: document table and term table, where document table consists of a set of document records, whereas term table consists of a set of term records. With such organization, it is easy to answer queries like "Find all of the documents associated with a given set of terms," or "Find all of the terms associated with a given set of documents". For example, to find all of the documents associated with a set of terms, we can first find a list of document identifiers in term table for each term, and then intersect them to obtain the set of relevant documents. Inverted indices are widely used in industry. They are easy to implement. A signature file is a file that stores a signature record for each document in the database. Each signature has a fixed size of b bits representing terms. A simple encoding scheme goes as illustrated. Each bit of a document signature is initialized to 0. A bit is set to 1 if the term it represents appears in the document. A signature S1 matches another signature S2 if each bit that is set in signature S2 is also set in S1. Since there are usually more terms than available bits, multiple terms may be mapped into the same bit. Such multiple-to-one mappings make the search expensive because a document that matches the signature of a query does not necessarily contain the set of keywords of the query. The document has to be retrieved, parsed, stemmed, and checked. Improvements can be made by first performing frequency analysis, stemming, and by filtering stopwords, and then using a hashing technique and superimposed coding technique to encode the list of terms into bit representation [10]. Nevertheless, the problem of multiple-to-one mappings still exists, which is the major disadvantage of this approach.

Bhushan Inje, Ujawla Patil have investigated the existing data mining methods with respect to the alternating approach for finding relevant pattern in large documents collection; some research works have been used phrases rather than individual words. However, the effectiveness of the text mining systems was not improved very much. The likely reason is that, a phrase-based method has "lower consistency of assignment and lower document frequency for terms". Hence, in this paper, they presented a concept for mining text documents for sequential patterns. Instead of using single words, they used pattern-based taxonomy (is-a) relation to represent documents. The semantic meaning of many discovered patterns is uncertain for answering what users want [14].

Ziqi Wang, et al. discussed the well-known dictionary matching algorithm called as Aho-Corasick algorithm. The AC tree is a trie with "failure links", on which the Aho-Corasick string matching algorithm can be executed. The Aho-Corasick algorithm is a well-known dictionary matching algorithm which can quickly locate the elements of a finite set of strings within an input string. The time complexity of the algorithm is of linear order in the length of input string plus the number of matched entries [15].

Saima Hasib, et al. has put light on Aho-Corasick algorithm. Aho-Corasick algorithm is best suited for multiple pattern matching and it can be used in many application areas, but it has been observed that as the size of automata increases drastically the performance of algorithm degrades in terms of

time and space both. The complexity of the algorithm is linear in the length of the patterns plus the length of the searched text plus the number of output matches. It is found to be attractive in large numbers of keywords, since all keywords can be simultaneously matched in one pass [16].

IV. SUMMERIZED TABLE

Sr. No.	Issue discussed	Advantages	Disadvantages
1.	Inverted Files are used for document retrieval[4]	Inverted files are easy to implement	Space requirement in memory to hold inverted file entries, and the time spend to process huge size inverted files maintaining record of each document.
2.	Suffix trees and Boyer-Moore[6]	Suffixes trees are used for efficient storing and retrieval and Boyer Moore algorithm used to check the presence of pattern	More space is wasted in trie(a multiway tree structure). Also difficult when input phrase is large.Retrieval time grows proportionally with size of documents.
3.	Phrase based approach[11]	phrases are less ambiguous and more discriminative.	Phrase-based approach have low frequency of occurrence,and there are large numbers of redundant and noisy phrases among them.
4.	Inverted indices and signature files[10]	Inverted indices are easy to implement	In signature files,multiple-to-one mappings make the search expensive.
5.	Pattern based method[14]	Phrase based method have lower consistency of assignment, lower document frequency for terms which is overcome by pattern based methodology.	The semantic meaning of many discovered patterns is uncertain for answering what users want.
6.	Aho-Corasick tree(used for string matching)[15]	Quickly locate the elements of a finite set of strings.	The time complexity of the algorithm is of linear order in the length of input string plus the number of matched entries.
7.	Aho-Corasick tree(used for string matching)[16]	Best suited for multiple pattern matching.	The performance of algorithm degrades in terms of time and space both.

V. CONCLUSION

In the last decade, many data mining techniques have been proposed for fulfilling various document retrieval tasks. These techniques include association rule mining, frequent itemset mining, sequential pattern mining, maximum pattern mining and closed pattern mining. However phrases in the field of text mining are difficult and ineffective. The reason is that a long phrase with high specificity lacks in support. Hence, adequate use of keywords or term based approach can extract the documents which are of user interest which can leads to the effective performance.

ACKNOWLEDGMENT

The authors would like to thank fellows of IJCSE for their reviews on this paper. I am grateful to my guide Prof. Ashwini Meshram for her valuable suggestions and encouragement. Special thanks to the authors of the reference papers which help me to understand the different techniques.

REFERENCES

[1] Debnath Bhattacharyya, Poulami Das," Unstructured Document Categorization: A Study", International Journal of Signal Processing, Image Processing and Pattern Recognition, pp. 55-62,Jan **2008**.

[2] Weiguo Fan, "Tapping into the Power of Text Mining", article accepted for publication at the

Communications of ACM, pp. 02-15, February 16, **2005**.

[3] V.V.Jaya Rama Krishnaiah, D.V.Chandra Sekhar, Dr. K. Ramchand H Rao, Dr. R Satya Prasad," Predicting the Diabetes using Duo Mining Approach", International Journal of Advanced Research in Computer and Communication Engineering ISSN : 2278 – 1021,Vol. 1, Issue 6, pp. 423-431, August **2012**.

[4] K.Sreerama Murthy, Dr G. Samuel Varaprasad Raju, Dr C. Sunil Kumar," Text Mining For Retrieving The Vital Information", International Journal of Research in Computer and Communication Technology, Vol. 3, Issue 1, pp.99-103,Jan **2014.**

[5] Manish Sharma, Rahul Patel," A Survey on Information Retrieval Models, Techniques and Applications", International Journal of Emerging Technology and Advanced Engineering, ISSN 2250-2459**,**pp.542-545, November **2013**.

[6] B.Ganga," Phrase Based Document Retrieving by Combining Suffix Tree index data structure and Boyer-Moore faster string searching algorithm", International Journal of Advancements in Research & Technology, ISSN 2278-7763,Vol. 3, Issue 3, pp. 147-153,March **2014**.

[7] Ian H. Witten," Text mining", Computer Science, University of Waikato, Hamilton, New Zealand, pp 01-23,**2004.**

[8] Roi Blanco González," Index Compression for Information Retrieval Systems", Ph.D. Thesis, University of A Coruña, **2008.**

[9] Deepak Agnihotri, Kesari Verma, Priyanka Tripathi," Pattern and Cluster Mining on Text Data", Fourth International Conference on Communication Systems and Network Technologies, IEEE Computer Society, pp. 428-432, **2014**.

[10] R. Sagayam, S.Srinivasan, S. Roshni," A Survey of Text Mining: Retrieval, Extraction and Indexing Techniques", International Journal Of Computational Engineering Research, ISSN 2250-3005, Vol. 2 Issue. 5, pp. 1443-1446, September **2012**.

[11] Sonali Vijay Gaikwad, Prof. Archana Chaugule, Swapnil Kulkarni, " Performance Comparison for Text Mining Methods: Review", International Journal of Advanced Engineering Research and Studies, E-ISSN 2249–8974, pp. 01-04, Oct.-Dec, **2014.**

[12] Ning Zhong, Yuefeng Li, and Sheng-Tang Wu," Effective Pattern Discovery for Text Mining", IEEE Transactions On Knowledge And Data Engineering, Vol. 24, No. 1,pp. 30-44, Jan. **2012**.

[13] S.S. Patil,V.M. Gaikwad, " Developing New Software Metric Pattern Discovery for Text Mining", International Journal of Computer Sciences and Engineering, Vol. 2, Issue-4,pp. 119-125, April **2014**.

[14] Bhushan Inje, Ujawla Patil," Operational Pattern Revealing Technique in Text Mining", IEEE Students' Conference on Electrical, Electronics and Computer Science,**2014.**

[15] Ziqi Wang, Gu Xu, Hang Li, and Ming Zhang," A Probabilistic Approach to String Transformation",published in IEEE Transactions On Knowledge And Data Engineering, Vol. 26, No. 5,pp. 1063-1075, May **2014.**

[16] Saima Hasib, Mahak Motwani, Amit Saxena," Importance of Aho-Corasick String Matching Algorithm in Real World Applications" published in International Journal of Computer Science and Information Technologies, ISSN: 0975-9646, Vol. 4 (3) , pp. 467-469,**2013**.

