International Journal of Advanced Computer Technology (IJACT)

ISSN:2319-7900

Enhanced Method for Text Searching Using Compression in Information Retrieval

Nazia Tabassum, Student (M.Tech ,CS), Jamia Hamdard; Ehtiram Raza Khan,Asst Prof.,Deptt of CS,Jamia Hamdard.

Abstract

With the exponential rise in the volume of database there is a demand for faster and timely retrieval of data.Early data retrieval (IR) systems used straightforward tools of word matching, and the texts to be searched were small.But in the present situation, due to giant volume of knowledge sources ,which too is in numerous forms, there is a requirement of additional Economical retrieval techniques which might retrieve only that part of the data which has relevancy. This paper discusses the underlying Principles and challenges of IR, and provides relatives Complexities of tools and techniques for the 2 major jobs performed by any IR system, i.e., data illustration and knowledge Retrieval.

Introduction

Information Retrieval (IR) is the discipline that deals with retrieval of unstructured data, especially textual documents, in response to a query or topic statement, which may itself be unstructured, e.g., a sentence or even another document, or which may be structured, e.g., a boolean expression. The need for effective methods of automated IR has grown in importance because of the tremendous explosion in the amount of unstructured data, both internal, corporate document collections, and the immense and growing number of document sources on the Internet. An information retrieval process begins when a user enter a query into the system. Queries are formal statements of information needs, for example search strings in web search engines. In information retrieval a query does not uniquely identify a single object in the collection. Instead, several objects may match the query, havingh different degree of relevancy. An object is an entity that is represented by information in database. User queries are matched against the database information. Often the documents themselves are not kept or directly stored in the IR system, but are instead represented in the system by metadata. Most IR system compute a numeric score on how well each object in the database match the query , and rank the objects according to value. Text categorization, text routing and text filtering system are all concerned with long term information needs. Text categorization labels the text automatically based on a set of predefined categories. For example, computer science abstracts might be classified based on subject areas, like -Operating System, Data Structure, and Artificial Intelligent etc. Text filtering systems allow only certain texts to pass through the filter. The filter specifies topics which interest the user and only such topics are passed to the user [1]. The paper presents the challenges of IR for the text representation retrieval and proposes new directions for more realistic approaches.

Principles and challenges in IR

Query-based IR system must be able to accept a query about any topic and find texts that contain the specified information of query. Many texts (also called text database) are very large, and sometimes IR systems are required to operate in real-time, which demand the IR system to be fast and efficient. Also, the search is conducted on the natural language text, which inherently have all the ambiguities and imprecision. The following part discusses some of the new changes in trends of IR system must incorporate.

Information Retrieval System Procedure

The main problem in the input is to obtain a representation of each document and generate a query suitable for a computer to use.Most computer-based retrieval systems store only a representation of the document (or query) which means that the text of a document is lost once it has been processed for the purpose of generating its representation. The two factors likely to be affecting are

A. Document Representation

A document representative could, for example, be a list of extracted words considered to be significant. Rather than having the computer process the natural language, an alternative approach is to have an artificial language within which all queries and documents can be formulated. There is some evidence to show that this can be effective [6]. Of course it assumes that a user is willing to be taught to ex-

press his information need in the language. When the retrieval system is on-line, it is possible for the user to change his request during one search session in the light of sample retrieval, and improving the subsequent retrieval run. Such a procedure is commonly referred to as feedback. An example of a sophisticated on-line retrieval system is the Medline system [7].

B. Processor Part

Secondly, the processor, that part of the retrieval system which is concerned with the retrieval process. The process may involve structuring the information in some appropriate way, such as classifying it. It will also involve performing the actual retrieval function that is executing the search strategy in response to a query

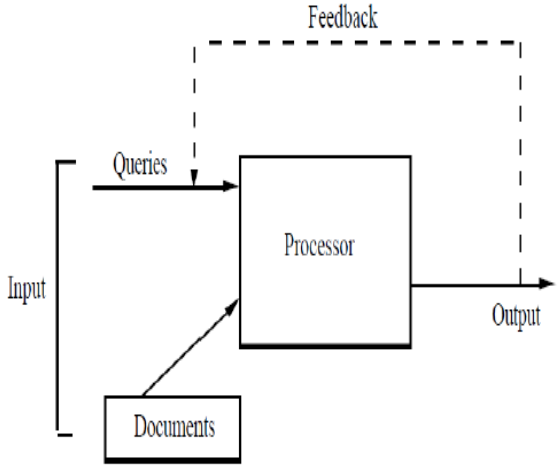

Figure. 1 Retrieval Process [15]

In figure. 1 the documents have been placed in a separate box to emphasize the fact that they are not just input but can be used during the retrieval process in such a way that their structure is more correctly seen as part of the retrieval process. Generally output would be usually a set of citations or document numbers. In operational system the flow ends here. However, in an experimental system it leaves the evaluation to be done.

Proposed Techniques

A. Automatic Classification

Document clustering on large document collections can be both effective and efficient. This means more research is needed to devise ways of speeding up clustering algorithms without sacrificing too much structure in the data. It may be possible to design probabilistic algorithms for clustering procedures which will compute a classification on the average in less time than it may require for the worst case. For example, it may be possible to cut down the 0(n2) computation time to expected 0(nlogn), although for some cases it would still require 0(n2). A big question that has not yet received much attention, concerns the extent to which retrieval effectiveness is limited by the type of document description used.

The use of keywords to describe documents has affected the way in which the design of an automatic classification system has been approached. It is possible that in the future documents will be represented inside a computer entirely differently. Document classification is a special case of a more general process which would also attempt to exploit relationships between documents. It so happens that dissimilarity coefficients have been used to express a distance-like relationship. Quantifying the relationship in this way has in part been dictated by the nature of the language in which the documents are described. However, were it the case that documents were represented not by keywords but in some other way, perhaps in a more complex language, then relationships between documents would probably best be measured differently as well. Consequently, the structure to represent the relationships might not be a simple hierarchy, except perhaps as a special case.

B. File Structures

On the file structure chosen and the way it is used depends the efficiency of an information retrieval system. Inverted files have been rather popular in IR systems. Certainly, in systems based on un-weighted keywords especially where queries are formulated in Boolean expressions, an inverted file can give very fast response. Unfortunately, it is not possible to achieve an efficient adaptation of an inverted file to deal with the matching of more elaborate document and query descriptions such as weighted keywords. The only way of getting at this may be to start with a document classification and investigate file structures appropriate for it. Along this line it might well prove fruitful to investigate the relationship between document clustering and relational data bases which organize their data according to n-ary relations.

C. Search Strategies

One approach would involve having a number of cluster representatives each derived from the data according to different principles. Probabilistic search strategies have not been investigated much either, although such strategies have been tried with some effect in the fields of pattern recognition and automatic medical diagnosis. Of course, in these fields the object descriptions are more detailed than are the document descriptions in IR mentioned that bottom-up search strategies are apparently more successful than the more traditional top-down searches. This leads me to speculate than it may well be that a spanning tree on the documents could be an effective structure for guiding a search for

ISSN:2319-7900

relevant documents. A search strategy based on a spanning tree for the documents may well be able to use the dependence information derived from the spanning tree for the index terms. An interesting research problem would be to see if by allowing some kind of interaction between the two spanning trees one could improve retrieval effectiveness.

D. Simulation : The three areas of research discussed so far could fruitfully be explored through a simulation model. We now have sufficiently detailed knowledge to enable us to specify a reasonable simulation model of an IR system. For example, the shape of the distributions of keywords throughout a document collection is known to influence retrieval effectiveness. It is possible to devise more efficient file structures by going through the performance of various file structures while simulating different keyword distributions. One major open problem is the simulation of relevance is no one has been able to simulate the characteristics of relevant documents successfully. Once this problem has been cracked it opens the way to studying such hypotheses as the Cluster and Association hypothesis by simulation.

New Trends in IR

A. Text Summarization

The most widely used method for summarizing / extracting data is shown in fig. 2. During the analysis phase of retrieval each sentence of the source text is evaluated in terms of the following features given below

- Location of the sentence in the text.
- Appearance of the cue phrase in the sections like - conclusion, abstract, major results, significant features, etc.
- Statistical salience which statistically discriminates one text from others in a collection

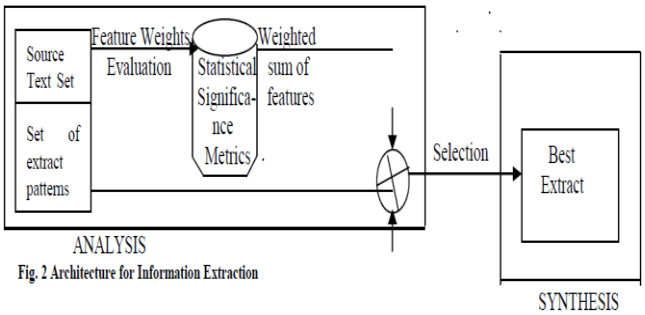

Fig. 2 Architecture for Information Extraction

If necessary, more features can be considered. A weighted sum of these features is compared with the weights of the phrasal extract patterns associated with the class of the text under analysis. For each sentence, the best possible phrasal extract having largest matching is selected. Finally during summarization the selected phrasal extracts are arranged in an appropriate order to represent the meaningful and coherent extract (i.e., summary).

B. Text Compression

Compressing both the index and the complete text cuts the total space in half. The time required to build the index and answer a query is far less than if the index and the text had not been compressed. Therefore, there is no space-time trade-off. For searching, the search key is compressed rather than decompressing the entire text. Consequently, using the compressed text makes the search faster because the fewer bytes are to be scanned [2].

Number of compression algorithms exists, but not all support the searching in the compressed text. Most compression methods use inverted tree data structure similar to the case of uncompressed text. One requirement in compressed text is random accessibility of words and phrases for implementation of text searching.

One of the efficient techniques for text compression is Huffman Compression. This assigns smaller symbols (codes) to the words having higher frequency of their occurrence and longer symbols to the words having lower frequency. The other advantages of Huffman methods are that it lets you search the compressed text exactly as you would do the uncompressed text. When you submit a query; the text is in compressed form, and the pattern is in uncompressed form. For searching the text the pattern is compressed rather than uncompressing the text. The compression occurs in two passes over full text. In the first pass frequency of each word is obtained and in the second pass actual compression is performed. The reference [2] discusses the various techniques for IR for the compressed source text.

Algorithm

Ultimately we present here an algorithm to develop a text processing system which by means of computable methods with the minimum of human intervention will generate from the input text (full text, abstract, or title) a document representative adequate for use in an automatic retrieval system. A document will be indexed by a name if one of its *significant* words occurs as a member of that class. Such a system will usually consist of three parts

- Removal of high frequency words.
- Suffix stripping.
- Detecting equivalent stems.

The removal of high frequency words, 'stop' words or 'fluff' words is one way of implementing Luhn's upper cut-off. This is normally done by comparing the input text with a 'stop list' of words which are to be removed. The advantages

ISSN:2319-7900

of the process are not only that non-significant words are removed and will therefore not interfere during retrieval, but also that the size of the total document file can be reduced by between 30 and 50 per cent. The second stage, suffix stripping, is more complicated. A standard approach is to have a complete list of suffixes and to remove the longest possible one. For example, we may well want UAL removed from FACTUAL but not from EQUAL. To avoid erroneously removing suffixes, context rules are devised so that a suffix will be removed only if the context is right. _Right' may mean a number of things:

- The length of remaining stem exceeds a given number; the default is usually 2.
- The stem-ending satisfies a certain condition, e.g. does not end with Q.

Many words, which are equivalent in the above sense, map to one morphological form by removing their suffixes. Others, unluckily, though they are equivalent, do not. It is this latter category which requires special treatment. Probably the simplest method of dealing with it is to construct a list of equivalent stem-endings. For two stems to be equivalent they must match except for their endings, which themselves must appear in the list as equivalent. For example, stems such as ABSORB- and ABSORPT- are conflated because there is an entry in the list defining B and PT as equivalent stem-endings if the preceding characters match.

The assumption (in the context of IR) is that if two words have the same underlying stem then they refer to the same concept and should be indexed as such. This is obviously an oversimplification since words with the same stem, such as NEUTRON AND NEUTRALISE, sometimes need to be distinguished. Even words which are essentially equivalent may mean different things in different contexts. Since there is no cheap way of making these fine distinctions we put up with a certain proportion of errors and assume (correctly) that they will not degrade retrieval effectiveness too much.

It is inevitable that a processing system such as this will produce errors. Fortunately experiments have shown that the error rate tends to be of the order of 5 per cent surprisingly, this kind of algorithm is not core limited but limited instead by its processing time. The final output from a conflation algorithm is a set of classes, one for each stem detected. A class name is assigned to a document if and only if one of its members occurs as a significant. A document representative then becomes a list of class names. These are often referred to as the documents *index terms* or *keywords*. Queries are of course treated in the same way. They can be processed at the same time as the documents. In an operational situation, the text processing system needs to be applied to the query at the time that it is submitted to the retrieval system.

Future Aspects

Information retrieval systems are likely to play an important part in or day to day life. They are likely to be on-line and interactive. One major recent development is that computers and data-bases are becoming linked into networks. It is foreseeable that individuals will have access to these networks through their private telephones and use normal television sets as output devices.

The main impact of this for IR systems will be that they will have to be simple to communicate with, which means they will have to use ordinary language. It is likely that an IR system will be expected to provide not just a citation, but a display of the text, or part of it, and perhaps answer simple questions about the retrieved documents. Even specialists may well desire of an information retrieval system that it do more than just retrieve citations. To bring all this will have to be integrated with data retrieval systems, to give access to facts related to those in the documents.

Another hardware development likely to influence the development of IR systems is the marketing of cheap microprocessors. Because these cost so little now, many people have been thinking of designing 'intelligent' terminals to IR systems, that is, ones which are able to do some of the processing instead of leaving it all the main computer. One effect of this may well be that some of the so-called more expensive operations can now be carried out at the terminal. It is an unfortunate fact that so much modern technology is established before we can actually assess whether we want it or not. In the case of information retrieval systems, there is much time to predict and investigate their impact. If we think that IR systems will make an important contribution, we ought to be clear about what it is we are going to provide and why it will be an improvement on the traditional methods of retrieving information.

The Understanding seems to us the key issue getting the machine to understand or simulate understanding of what a user is asking and helping users to understand what the retrieval system has done or is offering as a response to a query. And then, we must sooner or later get to the point where we can query graphics and sound recording

A. Context

An information retrieval system should be able to base word association or document matching techniques on a selected context. This, of course, would require a great many specialized dictionaries and thesauri, and possibly the searcher telling the system what that context is. We have made significant improvements in recent years, but far more could be done.

B. Syntax

We have made relatively little progress in interpreting the syntax of a natural language query. When one word modifies another syntactically, it changes its meaning and that should, in turn, change what the IR system is searching for. We still cannot handle nor logic in natural language, as it is even difficult to do so in English. Finally, there is the anaphora problem –recognizing the intended meaning of words that have no inherent meaning, but refer to other words – such as *it* or *that*

C. User Training

It has been metioned that the end users often lack the skills needed to compose queries and validate the results. The searching for information is a topic that should be taught in schools, beginning at the primary school level.The concept of information searching like in asking direction on the road, helping a customer in a shop, or searching for information in a library or database.It is just as important for potential doctors, lawyers, librarians, nurses, or salespeople.

D. Searching Graphics and Sound Records

Today, we cannot search a file of graphic images by drawing a picture of what we want to retrieve and having the system find the matches.. In general, graphic images has to be converted to, or indexed as, text code, or numbers. We match fingerprint images, for example, not by directly comparing images, but by comparing versions of the images converted to codes and numbers. We would want full graphic search capability in the future, as well as sound recordings, both without having had to first represent their content as text.

Conclusion

Information retrieval is now so widely used by scientists, managers, and people in many other professions that both its scientific and commercial importance has been recognized.. This paper presents a brief discription of emerging techniques for information representation and its retrieval to meet the challenges due to information explosion in this internet era. The choice of a technique is highly problem specific. Often a hybrid approach, combining different techniques, provides improved results in the form of faster speed, increased compression. It is realized that information representation and retrieval techniques, based on AI, Bayesian Probabilistic approach, relevant feedback-based interactive retrieval, text summarization and comprehension will continue to remain in focus, and will receive attention of a large number of Information System research workers.

Acknowledgments

The authors are thankful to IJACT Journal for the support to develop this document.

References

[1] Tucker Jr. A.B., ―The Computer Science and Engi neering Hand book, CRC /ACM, 1997.

[2] Ziviani Netal, ―Compression A Key for Next-Generation Text Retrieval Systems‖, In IEEE Com puter, Vol. 33, No.11, pp. 37- 44, Nov. 2000

[3] Zadozny Wetal., "Natural language for Personalized Interactions", Communications of the ACM, pp. 116-120, Vol.43, No.8, Aug. 2000.

[4] Hahn U. and Mani I., ―The Challenges of Automatic Summarization‖, IEEE Computer, Vol. 33, No.11, pp. 29-36, Nov. 2000.

[5] Subasic P., Huettner A., "Affect Analysis of Text Us ing Fuzzy Sematic *Typing*" In IEEE transactions on Fuzzy Systems, pp. 483-496, Vol. 9, No. 4, Aug. 2000

[6] Barber A.S., Barraclough E.D. and Gray W.A., ―On-line information retrieval as a scientist 'stool', Informa tion Storage and Retrieval‖, 9, 429-44- (1973).

[7] Klavans J., Kan M., ―Role of Verbs in Document Analysis in Proceedings of the 17th international Con ference on Computational linguistics" , pp.680-686, (COLING-ACL '98) Montreal, Canada: Aug. 1998.

[8] Baeza-Yates R. and Ribeiro B., ―Modern Information Retrieval", Addi-son Wesley Longman, Reading, Mass., 1999

[9] Witten I., Moffat A. and Bell T., "Managing Giga bytes, 2nd ed"., Morgan Kaufmann, San Francisco, 1999.

[10] Mccarn D.B. and Leiter J., ―On-line services in medi cine and beyond Science", 181, 318-324 (1973).

[11] Christopher D. Manning, Prabhakar Raghavan and Hinrich Schütze, ―Introduction to Information Re trieval ", Cambridge University Press. 2008

[12] B. Croft, D. Metzler, T. Strohman, ―Information Retrieval in Practice" Pearson Education, 2009

[13] Ayşe Göker, John Davies, ―Information Retrieval: Searching in the 21st Century‖ 2010.

[14] Paolo Ferragina, Giovanni Manzini, "On Compress ing the Textual Web" University di Pisa, 2010.

[15] C. J. van Rijsbergen, "Information Retrieval" ,University of Glasgow 2011.