A Boolean Model in Information Retrieval For Search Engines

ARASH HABIBI LASHKARI
FCSIT, University of Malaya (UM)
Kuala Lumpur, Malaysia
a_habibi_l@hotmail.com

FERESHTEH MAHDAVI
FCSIT, University of Malaya (UM)
Kuala Lumpur, Malaysia
fdotnet@Yahoo.com

VAHID GHOMI
Manufacturing Dep., Engineering Faculty,
University of Malaya (UM) ,KL, Malaysia
Vahid.ghomi@Gmail.com

***Abstract*— an information retrieval (IR) process begins when a user enters a query into the system. Queries are formal statements of information needs, for example search strings in web search engines. In IR a query does not uniquely identify a single object in the collection. Instead, several objects may match the query, perhaps with different degrees of relevancy.**

An object is an entity which keeps or stores information in a database. User queries are matched to objects stored in the database. Depending on the application the data objects may be, for example, text documents, images or videos. The documents themselves are not kept or stored directly in the IR system, but are instead represented in the system by document surrogates.

Most IR systems compute a numeric score on how well each object in the database match the query, and rank the objects according to this value. The top ranking objects are then shown to the user. The process may then be iterated if the user wishes to refine the query.

In this paper we try to explain IR methods and asses them from two view points and finally propose a simple method for ranking terms and documents on IR and implement the method and check the result.

Keywords*— *Information Retrieval, Data Mining, Search Engine, Ranking

I. INTRODUCTION

The meaning of the term information retrieval can be very broad. Just getting a credit card out of your wallet so that you can type in the card number is a form of information retrieval. However, as an academic field of study, Information retrieval (IR) is finding material (usually documents) of an unstructured nature (usually text) that satisfies an information need from within large collections (usually stored on computers).

As defined in this way, information retrieval used to be an activity that only a few people engaged in: reference librarians, paralegals, and similar professional searchers. Now the world has changed, and hundreds of millions of people engage in information retrieval every day when they use a web search engine or search their email.

Information retrieval is fast becoming the dominant form of information access, overtaking traditional database style searching (the sort that is going on when a clerk says to you: "I'm sorry, I can only look up your order if you can give me your Order ID").

IR can also cover other kinds of data and information problems beyond that specified in the core definition above. The term "unstructured data" refers to data which does not have clear, semantically overt, easy-for-a-computer structure. It is the opposite of structured data, the canonical example of which is a relational database, of the sort companies usually use to maintain product inventories and personnel records. In reality, almost no data are truly "unstructured".

II. Information Retrieval Models

Retrieval models form the theoretical basis for computing the answer to a query. They differ not only in the syntax and expressiveness of the query language, but also in the representation of the documents (For the information retrieval to be efficient, the documents are typically transformed into a suitable representation), there are several representations.

In a total view we have two categories of IR model that are Ad hoc and browsing, you can see the detail of these categories and relations in the picture "Fig. 1".

A. Ad hoc

In ad hoc information retrieval, a user often needs to interact with the retrieval system several times to obtain satisfactory results for one information need, which provides opportunities for the retrieval system to actively participate in this iterative retrieval process. Most traditional retrieval systems just passively respond to user queries and put the responsibility of refining/improving the search solely on the user. But there has been evidence showing that a retrieval system can play an active role in this process, e.g., obtaining user feedback explicitly or implicitly when the user browses these documents, and exploiting such

978-0-7695-3595-1/09 $25.00 © 2009 IEEE
DOI 10.1109/ICIME.2009.101

information to improve the performance in the next round of search

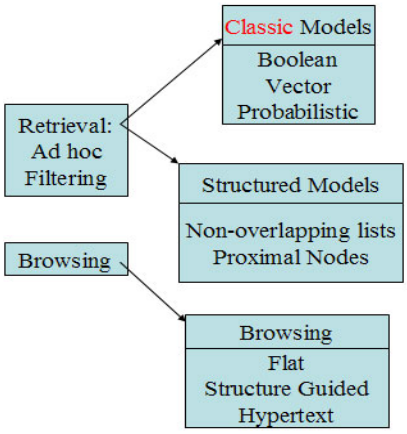

Figure 1. Query or Non-Query based categorize

B. *Browsing model*

A model of browsing-based conceptual information retrieval is proposed employing two different types of dictionaries, a global dictionary and a local dictionary. A global dictionary with the authorized terms is utilized to capture the commonly acknowledgeable conceptual relation between a query and a document by replacing their keywords with the dictionary terms.

In the other hand we distinguish query-oriented systems from non-query-oriented or exploratory systems "Fig. 2". Query-oriented systems may be characterized as conventional or non-conventional. Conventional retrieval systems are exact-match systems. Online catalogs modeled after the card catalog or Boolean information retrieval systems, or both, are exact-match systems. Where conventional information retrieval systems employ an exact match retrieval strategy, non-conventional query-oriented systems employ a "closest" or "best match" strategy where degree of closeness or similarity of a candidate item's content description to the textual query is taken into account.

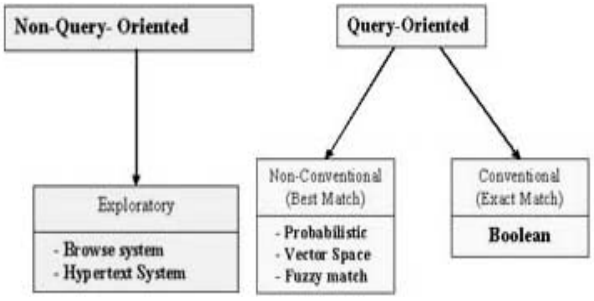

Figure 2. Query or Non-Query based categorize

Before we explain some of retrieval models, define consists of each retrieval model:

F: a modeling framework for D, Q and the relationships among them
D: representation for documents
Q: An Expression
R: representation for queries

R (q, d^i): a ranking or similarity function which orders the documents with respect to a query.

III. STANDARD BOOLEAN MODEL

The Standard Boolean model, one of the earliest and simplest retrieval methods, uses exact matching to match documents to a user "query" or information request by finding documents that are "relevant" in terms of matching the words in the query. The adjective "Boolean" refers to the use of Boolean algebra, whereby words are logically combined with the Boolean operators AND, OR, and NOT. for example, the Boolean AND of two logical statements x and y means that both x AND y must be satisfied, while the Boolean OR of these same two statements means that at least one of these statements must be satisfied. Any number of logical statements can be combined using the three Boolean operators.

Actually The Boolean Model is a simple retrieval model based on set theory and Boolean algebra that Documents are represented by the index terms assigned to the document. There is no indication on which terms are more important than others (weights are binary either 0 or 1).

F: Boolean algebra over sets of terms and sets of documents

D: set of words (indexing terms) present in a document that each term is either present (1) or absent (0).

Q: A Boolean expression that terms are index terms and operators are AND, OR, NOT

R (q, d^i): a document is predicted as relevant to a query expression if it satisfies the query expression

That each query terms specifies a set of documents containing the term:

And ($\wedge$): The intersection of two sets.
OR ($\vee$): The union of two sets
Not ($\neg$): Set inverse or really set difference

For example:
If we have 4 documents as:
Doc1: Information Retrieval has 2 models and Information.
Doc2: Boolean is a basic Information Retrieval classic model.
Doc3: Information is a data that processed, Information.
Doc4: When a Data Processed the result is Information, Data.
And D is: Information, Data, and Retrieval

Then we have:

R	Doc1	Doc2	Doc3	Doc4
Data	0	0	1	1
Retrieval	1	1	0	0
Information	1	1	1	1

If Q is: (Data $\wedge$ Information) $\vee$ ($\neg$ Retrieval)
You have:

Data: Doc3, Doc4
Retrieval: Doc1, Doc2
Information: Doc1, Doc2, Doc3, Doc4

Then:
Data: Doc3, Doc4
$\neg$Retrieval: Doc3, Doc4
Information: Doc1, Doc2, Doc3, Doc4

Then:

Data $\wedge$ Information: Doc3, Doc4
$\neg$Retrieval: Doc3, Doc4

Then Result is:

(Data $\wedge$ Information) $\vee$ ($\neg$Retrieval): Doc3, Doc4

Now we can say that the advantages of Boolean model are:

- Clean Formalism
- Easy to implement
- But the disadvantages of Boolean models are:
- Exact matching may retrieve too few or too many documents.
- Difficult to rank output.
- Difficult to control the number of documents retrieved that all matched documents will be returned.

IV. Vector Space Model

In the vector space model, we represent documents as vectors. The success or failure of the vector space method is based on term weighting. In the vector space IR model, a vector is used to represent each item or document in a collection. Each component of the vector reflects a particular concept, key word, or term associated with the given document. The value assigned to that component reflects the importance of the term in representing the semantics of the document. Typically, the value is a function of the frequency with which the term occurs in the document or in the document collection as a whole. Suppose a document is described for indexing purposes by the three terms applied, linear, and algebra. It can then be represented by a vector in the three corresponding dimensions. The word algebra is the most significant term in the document, with linear of secondary importance and applied of even less importance.

A database containing a total of d documents described by t terms is represented as a t_d term-by-document matrix A. The d vectors representing the d documents form the columns of the matrix. Thus, the matrix element a^{ij} is the weighted frequency at which term i occurs in document j. In the parlance of the vector space model, the columns of A are the document vectors, and the rows of A are the term vectors. The semantic content of the database is wholly contained in the column space of A, meaning that the document vectors span that content. Not every vector represented in the column space of A has a specific interpretation in terms of the document collection itself (i.e., a linear combination of vectors corresponding to two document titles may not translate directly into a meaningful document title). What is important from an IR perspective, however, is that we can exploit geometric relationships between document vectors to model similarities and differences in content.

We can also compare term vectors geometrically in order to identify similarities and differences in term usage.

A variety of schemes are available for weighting the matrix elements. In this case, the factory is a global weight that reflects the overall value of term i as an indexing term for the entire collection. As one example, consider a very common term like computer within a collection of articles on personal computers.

For example:
We have 3 documents:

Doc1: A student and a police….
Doc2: A student and his friend
Doc3: A student and his lecturer….
Then we need to dividing terms and found the number of frequently of terms:

R	Doc1	Doc2	Doc3
A	2	1	1
Student	1	1	0
And	1	1	1
His	0	1	1
Police	1	0	0
Friend	0	1	0
lecturer	0	0	1

Then we must sort the table by terms:

R	Doc1	Doc2	Doc3
A	2	1	1
And	1	1	1
Friend	0	1	0
His	0	1	1
Lecturer	0	0	1
Police	1	0	0
Student	1	1	0

Then you can make vectors for documents:
Doc1's Vector: 2,1,0,0,0,1,1
Doc2's Vector: 1,1,1,1,0,0,1
Doc3's Vector: 1,1,0,1,1,0,0

Now you can make vector for each term because each doc is a vector here:
Doc1=2T1+T2+T6+T7
Doc2=T1+T2+T3+T4+T7

Doc3=T1+T2+T4+T5

And for each term generate a vector:

A, Doc1=2,0,0,0,0,0,0

And, Doc1=0,1,0,0,0,0,0

…

Now you can calculate the weight of each document on each term for each Q "Fig. 3":

For example our Q is: "A student friend"

D=3, Dfi=number of non zero columns of all documents, IDFi=Log (D/Dfi)

		Counts						Weights			
Terms	Q	Doc1	Doc2	Doc3	Dfi	D/DFi	IDFi	Q	Doc1	Doc2	Doc3
A	1	1	1	1	3	3/3=1	0	0	0	0	0
And	0	1	1	1	3	3/3=1	0	0	0	0	0
Friend	1	0	1	0	1	3/1=3	0.47	0.47	0	0.47	0
His	0	0	1	1	2	3/2=1.5	0.17	0	0	0.17	0.17
Lecturer	0	0	0	1	1	3/1=3	0.47	0	0	0	0.47
Police	0	1	0	0	1	3/1=3	0.47	0	0.47	0	0
Student	1	1	0	0	1	3/1=3	0.47	0.47	0.47	0	0

Figure 3. Total Weight of each document on each term

In this model you can make a dictionary from all terms in each document and the frequency of each term in that document that it is a local dictionary and in the future you can see that the best information retrieval model in these days use the local dictionary and Global Dictionary that we will explain in this assignment.

Now we can say that the advantages of Vector space model are:

- Its term weighting scheme can improve retrieval performance
- Allows partial matching
- Retrieved documents are sorted according to their degree of similarity
- But the disadvantages of Boolean models are:
- Terms are assumed to be mutually independent. In some cases this might hurt performance.

V. Inference Network (probabilistic)

Probabilistic models attempt to estimate the probability that the user will find a particular document relevant. Retrieved documents are ranked by their odds of relevance -- the ratio of the probability that the document is relevant to the probability that the document is not relevant to the query. This model operates recursively and requires that the underlying algorithm guess at initial parameters, then iteratively try to improve this initial guess to obtain a final ranking of relevancy probabilities.

Improvements to the basic probabilistic model of information retrieval are made with Bayesian inference techniques (BAEZA-YATES & RIBEIRO-NETO, 1999).

Unfortunately, probabilistic models can be very hard to build and program. Their complexity grows quickly, deterring many researchers and limiting the scalability of the engine based on the model. Probabilistic models also require several unrealistic simplifying assumptions, such as independence between terms as well as between documents. For instance, in this document the most likely word to follow information is the word retrieval, so it is not reasonable to assume that these terms are independent.

The retrieval performance of an IR system usually increases when it uses the relationships among terms contained in given document collection, so it is becoming a hotspot in IR research of how to obtain these relationships efficiently, and how to use them to retrieval document given a user's query.

There are two kinds of means to characterize term-to-term relationships:

One is synonym, which can be interchanged in information retrieval system, just like "worker" and "workman".

Another is which have different means but there are some relationships between two Related-words, like "computer" and "network".

A synonym of one word can be obtained in a thesaurus, and the related word usually can be determined by co-occurrences frequency method or co-occurrences analysis method.

VI. Fuzzy Model

Classical retrieval approaches are mainly guided by efficiency rather than expressiveness. This yield to Information Retrieval (IR) systems which retrieve documents very efficiently but their internal representations of documents and queries is simplistic. This is especially true for web retrieval engines, which deal with huge amounts of data and their response time is critical. Nevertheless, it is well known that users have often a vague idea of what they are looking for and, hence, the query language should supply adequate means to express her/his information need.

VII. Web Search Engine

Search is a compelling human activity that has extended from the Library at Alexandria to the World Wide Web. The Web has introduced millions of people to search. The information retrieval (IR) community stands ready suggest helpful strategies for finding information on the Web. One classic IR strategy - indexing Web pages with topical metadata - has already been tried, but the results are disappointing.

If the Web is a big, distributed document database and Web pages are composed in HTML (i.e., 'the document in my browser goes from <html> down to </html>') ,it makes technological sense for Web authors to add topical metadata to Web pages, just as an indexer might add descriptors to a document in a database. An affirmative answer validates the topical metadata debate. If, however, the Web is not a big document database, but is instead a network of rapidly changing presentations, HTML is primarily a presentation technology, and most Web pages are transitory and volatile presentations governed by the whims of viewer taste and the contingencies of viewer technology. A negative answer signals that debating the value of topical metadata is premature until it can be shown that they are technologically appropriate additions to Web pages.

VIII. Our solution

Finally we try to find a simple way to calculate the weight of terms in a document.

When we find terms of a document, and then try to calculate the frequency of them:

T = Token

M = number or Tokens that selected

N= all Tokens of Document

Count= number of repeated a token in document

Then use this formula for calculate the weight of a term. "Fig. 4":

$$\text{Weight}(T_1 + T_2 + \ldots + T_M) = \frac{\sum_{i=1}^{m} count(i)}{m * \sum_{i=1}^{N} Count(i)}$$

Figure 4. Eight Colculated Formula

In the implementation we define these steps:

1. Create a text file and save it in hard drive.
2. Run insert procedure for saving the name and contents of the file in an oracle table (table name: Files).
3. Parsing the "files" table for eliminate the punctuations symbols such as ".", ",", "?", "!"…
4. Retrieve terms from "files" table and save them in a token table with the file name and frequency of each term.
5. Eliminates the duplicate terms from table "token"
6. Sort the "token" table
7. Calculate the weight of entry terms by using the "token" table data and our formula.

For use the formula and issued of applicable of it, we implemented an Oracle Base program by VB and run it for 20 sample text file, the program's results were so similar to the human result.

Conclusion

In this paper at first we define the meaning of Information Retrieval and explain Query based or Non-Query based algorithm, and Ad-Hoc filtering or browsing method. Secondly, in each group we explained the methods and try to discuss about two major methods by examples. Finally discuss about IR in search engine and propose a simple method in terms and documents ranking. Then try to implement it by Visual Basic on Oracle database and analyze the result. At the last step, we found the method was true in small and medium data size system base on terms frequency.

In these days we try to apply ontology and semantic attributes to our method and hope in the next paper we will propose a more complex method for huge data systems that support the semantic attributes.

Acknowledgment

We would like to express our appreciation to our parents and all the teachers and lecturers who help us to understand the importance of knowledge and show us the best way to gain it.

References

[1] JAMES ALLAN, "Information Retrieval Overview"

[2] ANDY MACFARLANE, "Overview of Open Source and Information Retrieval"

[3] Monika Henzinger , "Tutorial Web Information Retrieval"

[4] Robertson , Buckley and Singhal ,"Flavors of tf*idf weighting including BM 25, pivoted weight"

[5] Dragos and Anton Manolescu, "Feature Extraction—A Pattern for Information Retrieval"

[6] C.J. "Keith" van Rijsbergen, "Introduction to Information Retrieval"

[7] Prabhakar Raghavan, "Webbar 2007 Information Retrieval"

[8] David D.Lewis, "Natural Language Processing for information retrieval"

[9] Zhiwei Shao, "Introduction to Information Retrieval Systems"

[10] G. Bordogna, P. Carrara, G. Pasi , "EXTESDING BOOLEAN ISFOILlATIOS RETRIEVAL A FUZZY MODEL BASED ON LINGUISTIC VARIABLES"

[11] J. H. Wang, "IR Models"

[12] Jamie Callan , "Retrieval Models: Multiple Sources of Evidence"

[13] Michel Beigbeder and Annabelle Mercier, "An Information Retrieval Model Using the Fuzzy Proximity Degree of Term Occurrences"

[14] Thomas Hofmann , "Information Retrieval and Retrieval Models I"

[15] Campbell Wilson and Bala Srinivasan and Maria Indrawan, "A GENERAL INFERENCE NETWORK BASED ARCHITECTURE FOR MULTIMEDIA INFORMATION RETRIEVAL"

[16] Morten Hertzum , "A Comparison of Three Data Models for Text Storage and Retrieval Systems"

[17] Google search engine fundamental, Copyright © 2003 Google Inc. Used with permission.

[18] Internet web sites:

http://www.oracle.com/technology/pub/articles/cook_dotnet.html

http://download.oracle.com/docs/cd/B19306_01/win.102/b14307/featConnecting.htm#sthref87

http://www.oracle.com/technology/sample_code/tech/windows/odpnet/howto/connect/index.html

http://download.oracle.com/docs/cd/B19306_01/server.102/b14231/toc.htm

http://www.quest-pipelines.com/newsletter-v3/0702_C.htm

http://www.lorentzcenter.nl/awcourse/oracle/appdev.920/a96620/xdb03usg.htm