Information Retrieval Models, Techniques and Applications

Olalere A. Abass[1], Oluremi A. Arowolo[2]
[1, 2]Dept. of Computer Science, Tai Solarin College of Education, Omu-Ijebu, Ogun State, Nigeria

***Abstract*— An Information Retrieval (IR) system focuses on the processing of data collection by means of representation, storage, and searching for the purpose of knowledge discovery in response to user request via query. The tendency of the IRS to produce relevant documents with high precision and recall to meet user's need based on query input depends on the adoption of the appropriate techniques by the search engine. In this paper, we explain the concepts of IR and traditional models in which various IR techniques rely upon. We equally give detail description of IR techniques that have been successfully applied to store, manage and retrieve documents from huge amount of data available to users of IR systems. This shows that applications of these retrieval techniques in digital libraries, information filtering system, media search, search engine and domain-specific areas of IR are capable of increasing the throughput and minimize the access time of the user with respect to information needs.*

***Keywords*— IRS framework, models, IR techniques, recall, precision.*

I. Introduction

Information retrieval (IR), as subfield of computer science, deals with the representation, storage, and access of information and is concerned with the organization and retrieval of information from large database collections (Sagayam et al, 2012). In response to user request via query, IR focuses on the processing a collection of data by means of representation, storage, and searching for the purpose of knowledge discovery. This process involves various stages initiated with representing data and ending with returning relevant information to the user. Intermediate stages include filtering, searching, matching and ranking operations. The primary objective of information retrieval system (IRS) is to support users to access relevant information corresponding to their needs or a document that satisfies user information needs.

According to [1], there are two basic measures for assessing the quality of IRS as follows: (i) *Precision*- the percentage of retrieved documents that are in fact relevant to the query and (ii) *Recall* - the percentage of documents that are relevant to the query and were in fact retrieved. The tendency of the IRS to yield a list of relevant documents with high precision and recall to meet user's need as specified by the query depends on the use of the appropriate techniques by the search engine. This remains the focus of the paper as we attempt to fully explain different IR techniques so far used by various researchers and the developers of the IRS.

The structure of this paper is as follows. A brief review of IRS framework and IR models are presented in Section 2, followed by IR techniques in Section 3. Section 4 deals with different areas of application of IR techniques. Finally, the conclusion of the paper is drawn in Section 5.

II. Information Retrieval System

A. *Framework of IRS*

According to Sharma and Patel (2013), there are three basic processes an IRS has to support: (i) the representation of the content of the documents, (ii) the representation of the user's information need, and (iii) the comparison of the two representations. The processes are visualized in Figure 1 as opined by Sharma and Patel (2013). In the figure, squared boxes represent data and rounded boxes represent processes. Representing the documents is usually called the indexing process.

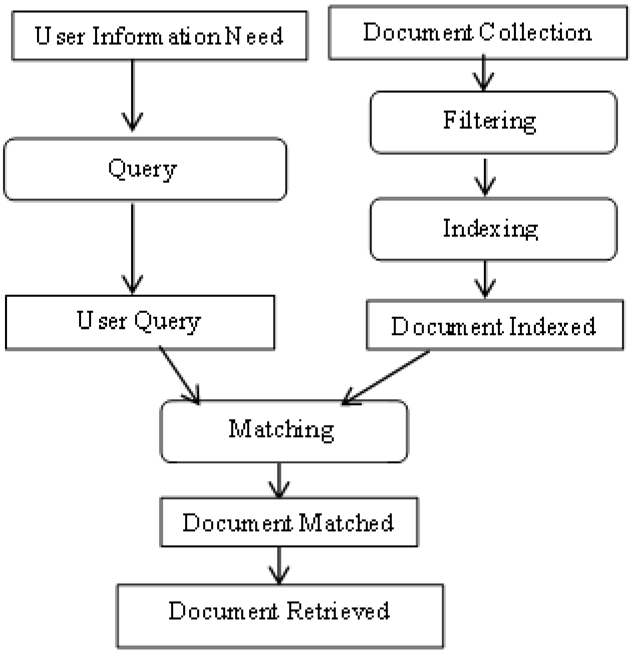


Fig. 1. A general framework of IRS.

The process takes place off-line, that is, the end user of the IRS is not directly involved. The indexing process results in a representation of the document, the process of representing user's information need is often referred to as the *query formulation process* and resulting representation is the query (Hiemstra, 2009). Comparing the two representations is known as the *matching process*. Retrieval of documents is the result of this process.

B. *Information Retrieval Models*

Mathematical models are used in many scientific areas with the objective to understand and reason about some behaviour or phenomenon in the real world. A model of IR predicts and explains what a user will find relevant to a given query. The correctness of the model's predictions can be tested in a controlled experiment. Hence, a model of IR serves as a blueprint which is used to implement an actual IRS (Hiemstra, 2009).

C. *The Traditional or Classical Models*

The three most used models in IR research are the vector space, the probabilistic model, and the inference network models (Singhal, 2001). These three models are regarded as the traditional retrieval models.

i. Boolean model (BM) - A measure of exact match

This model provides exact matching, i.e. documents are either retrieved or not, but the retrieved documents are not ranked. The retrieval function in this model treats a document as either relevant or irrelevant (Alhenshiri, 2003). That is, in BM, the retrieved documents are adjudged as either "relevant" or "not relevant".

ii. Vector space model (VSM) - A measure of document similar to query by ranking

The VSM can best be characterized by its attempt to rank documents by the similarity between the query and each document Salton and McGil, 1986). In the VSM, documents and query are represented as a vector and the angle between the two vectors is computed using the similarity cosine function. Similarity Cosine function can be defined as (Sharma and Patel, 2013):

$$sim(d_j.q) = \frac{d_j.q}{||d_j||\,||q||} = \frac{\sum_{i=1}^{N} w_{i,j} w_{i,q}}{\sqrt{\sum_{i=1}^{N} w_{i,j}^2}\ \sqrt{\sum_{i=1}^{N} w_{i,q}^2}} \qquad (1)$$

Documents and queries are represented as vectors.

$$d_j = (w_{1,j}, w_{2,j}, \dots, w_{t,j}) \qquad (2)$$

$$q = (w_{1,q}, w_{2,q}, \dots, w_{t,q}) \qquad (3)$$

VSM introduced term weight scheme known as *if-idf* weighting. These weights have a term frequency (*tf*) factor measuring the frequency of occurrence of the terms in the document or query texts and an inverse document frequency (*idf*) factor measuring the inverse of the number of documents that contain a query or document term (Samar et al, 2016). The VSM of IR is a very successful statistical method proposed by Salton and Buckley, 1988).

A major achievement of the researchers that developed the VSM is the introduction of the family of *tf.idf* term weights. These weights have a term frequency (tf) factor measuring the frequency of occurrence of the terms in the document or query texts and an inverse document frequency (idf) factor measuring the inverse of the number of documents that contain a query or document term.

iii. The probabilistic model – A measure of probability of relevance

This family of IR models is based on the general principle that documents in a collection should be ranked by decreasing probability of their relevance to a query. This is often called *the probabilistic ranking principle* –PRP (Robertson, 1990). The most important characteristic of the probabilistic model is its attempt to rank documents by their probability of relevance given a query (Robertson and Jones, 1976). Documents and queries are represented by binary vectors ~d and ~q, each vector element indicating whether a document attribute or term occurs in the document or query, or not. Instead of probabilities, the probabilistic model uses odds O(R), where O(R) = P(R)/1 − P(R), R means "document is relevant" and $\bar{R}$ means "document is not relevant" (Hiemstra et al, 2000).

III. Information Retrieval Techniques

A. *Term Weighting*

Weighting methods developed under the probabilistic models rely heavily upon better estimation of various probabilities (Singhal, 2001). Term weighting is a technique of obtaining the most critical piece of information needed for document ranking in all IR models. Various methods for weighting terms have been developed in the field. Weighting methods developed under the probabilistic models rely heavily upon better estimation of various probabilities (Robertson and Jones, 1976)]. Methods developed under the VSM are often based on researchers' experience with systems and large scale experimentation Salton and Buckley, 1988). In both models, three main factors come into play in the final term weight formulation (Singhal, 2001):

i. Term Frequency (or *tf*)

Words that repeat multiple times in a document are considered salient. Term weights based on *tf* have been used in the VSM since the 1960s. TF addresses how relevant is a particular document *d* to the given particular term *t*. One way of measuring *TF(d,t)*, the relevance of a document *d* to term *t*, is:

$$TF(d,t) = \log(1 + \frac{n(d,t)}{n(d)} \qquad (4)$$

where n(d) denotes the number of terms in the document and n(d,t) denotes the number of occurrences of term *t* in the document *d*.

ii. Document Frequency

Words that appear in many documents are considered common and are not very indicative of document content. A weighting method based on this, called *inverse document frequency* (or *idf*) weighting, was proposed by Sparck-Jones early 1970s. In a query which may contain multiple keywords, the relevance of a document to such a query with two or more keywords is estimated by combining the relevance measures of the document to each word. A simple way to combine the measure is to add them up. However, not all terms used as keywords are equal. To fix this problem, weights are assigned to terms using the *inverse document frequency (IDF) defined as:*

$$IDF(t) = \frac{1}{n(t)} \qquad (5)$$

where *n(t)* denotes the number of documents (among those indexed by the system) that contain the term *t*.

The relevance of a document *d* to a set of terms *Q* is the defined as:

$$r(d,Q) = \sum_{t \in Q} TF(d,t) * IDF(t) \qquad (6)$$

The weight of an index term is proportional to its frequency in a document (*term frequency* or *tf factor*), and inversely proportional to its frequency among all documents in the system (*inverse document frequency* or *idf factor*). This measure can be further refined if the user is permitted to specify weights *w(t)* for terms in the query, in which case the user-specified weights are also taken into account by multiplying *TF(t)* by *w(t)* in the above formula. This approach of using *term frequency* and *inverse document frequency* is called *TF-IDF* approach.

It is important that the assignment of weights to every index term (called "term weighting") is automatic. The so-called TF-IDF method is mainly used for knowing the weight of a term; TF is the frequency of occurrence of a term in a document and IDF varies inversely with the number of document to which the term is assigned Ropero et al, 2012).

iii. Document Length

This is the third factor in term weighting. When collections have documents of varying lengths, longer documents tend to score higher since they contain more words due to word repetitions. This effect is usually compensated by normalizing for document lengths in the term weighting method. Before TREC (Text Retrieval Conference), both the VSM and the probabilistic models developed term weighting schemes which were shown to be effective on the small test collections available then. Inception of TREC provided IR researchers with very large and varied test collections allowing rapid development of effective weighting schemes. The state-of-the-art scoring technique that combines the above three factors is called *Okapi weighting based document score* (Robertson et al, 1999) as shown in the Eq. 7 below.

$$\sum_{t\in Q,D} \ln\frac{N-df+0.5}{df+0.5}.\frac{(k_1+1)tf}{\left(k_1(1-b)+b\frac{dl}{avdl}\right)+tf}.\frac{(k_3+1)qtf}{k_3+qtf} \qquad (7)$$

tf is the term's frequency in document
qtf is the term's frequency in query
N is the total number of documents in the collection
df is the number of documents that contain the term
dl is the document length (in bytes), and
avdl is the average document lenght
k_1 (between 1.0-2.0), *b* (usually 0.75) and k_3 (between 0-1000) are constants

According to Singhal (2001), the pivoted normalization weighting based document score is

$$\sum_{t\in Q,D} \frac{1+\ln\left(1+\ln(tf)\right)}{(1-s)+s\frac{dl}{avdl}}.qtf\,.\ln\frac{N+1}{df} \qquad (8)$$

where s is a constant (usually 0.20).

B. Query Modification using Synonyms

In the early years of IR, researchers realized that it was quite hard for users to formulate effective search requests. It was thought that adding *synonyms* of query words to the query should improve search effectiveness. Early research in IR relied on a thesaurus to find synonyms (Singhal, 2001).

However, it is quite expensive to obtain a good general-purpose thesaurus. Researchers then developed techniques to automatically generate thesauri for use in query modification. Most of the automatic methods are based on analyzing word co-occurrence in the documents (which often produces a list of strongly related words). Most query augmentation techniques based on automatically generated thesaurii had very limited success in improving search effectiveness. The main reason behind this is the lack of query context in the augmentation process. Not all words related to a query word are meaningful in context of the query.

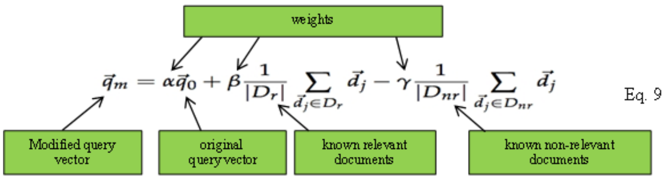
C Relevance Feedback for Query Modification

In IRS, the indexing step pre-processes documents and queries in order to obtain keywords (relevant words, also named terms) to be used in the query. At this point, it is important to consider the use of stemming and stopword (removal of words or terms that carry little or no semantically important information during searching and indexing processes) lists in order to reduce related words to their stem, base or root form.

Matching, as a process, involves computation of the similarity between documents and queries by weighting terms. The TF-IDF and BM25 (best match) algorithms are the frequently applied algorithms for term weighting. Base on the use of these algorithms, most IRS return a list of ranked document in response to a query where the documents more similar to the query considered by the system are first on the list. Once the first answer set is obtained, different query expansion techniques can be applied. For example, the most relevant keywords of the top documents previously retrieved can be added to the query in order to re-rank the documents. This process is called "relevance feedback" (RF). The retrieval can be further enhanced by modifying the words of the queries using other keywords more representative of the document content - e.g., including MeSH Headings (Rivas et al, 2014). In 1965, Rocchio proposed using RF for query modification (Singhal, 2001). RF is motivated by the fact that it is easy for users to judge some documents as relevant or non-relevant for their query. Using such relevance judgments, a system can then automatically generate a better query by adding related new terms for further searching. In general, the user is asked to judge the relevance of the top few documents retrieved by IRS.

Based on these judgments, the system modifies the query and issues the new query for finding more relevant documents from the collection. RF has been shown to work quite effectively across test collections.

Rocchio algorithm was the RF mechanism introduced and popularized by Salton's SMART system. In a real IR query context, there exists a user query and partial knowledge of known relevant and non-relevant documents.


weights
$\vec{q}_m = \alpha\vec{q}_0 + \beta\frac{1}{|D_r|}\sum_{\vec{d}_j\in D_r}\vec{d}_j - \gamma\frac{1}{|D_{nr}|}\sum_{\vec{d}_j\in D_{nr}}\vec{d}_j$
Eq. 9
Modified query vector
original query vector
known relevant documents
known non-relevant documents

The algorithm proposes using the modified query in Eq. 9 where q_0 is the original query vector, D_r and D_{nr} are the set of known relevant and non-relevant documents respectively, and *a*, *b*, and *g* are weights attached to each term. These control the balance between trusting the judged documents set versus the query: if there is a lot of judged documents, a higher β and γ are obtained.

Starting from q_0, the new query moves the user some distance toward the centroid of the relevant documents and some distance away from the centroid of the non-relevant documents. This new query can be used for retrieval in the standard VSM. We can easily leave the positive quadrant of the vector space by subtracting off a non-relevant document's vector.

In the Rocchio algorithm, negative term weights are ignored. That is, the term weight is set to 0. RF can improve both recall and precision. But, in practice, it has been shown to be most useful for increasing recall in situations where recall is important. This is partly because the technique expands the query, but it is also partly an effect of the use case: when they want high recall, users can be expected to take time to review results and to iterate on the search. Positive feedback also turns out to be much more valuable than negative feedback, and so most IRS set $\gamma < \beta$. Reasonable values might be $\alpha = 1$, $\beta = 0.75$, and $\gamma = 0.15$. In fact, many IRS allow only positive feedback, which is equivalent to setting $\gamma = 0$. Another alternative is to use only the marked non-relevant documents which received the highest ranking from the IR system as negative feedback.

New techniques to do meaningful QE in absence of any user feedback were developed early 1990s. Most notable of these is pseudo-feedback, a variant of relevance feedback (Buckley et al, 1995). Given that the top few documents retrieved by an IRS are often on the general query topic, selecting related terms from these documents should yield useful new terms irrespective of document relevance. In pseudo-feedback, the IRS assumes that the top few documents retrieved based on the initial user query are "relevant", and does RF to generate a new query. This expanded new query is then used to rank documents for presentation to the user. Pseudo feedback has been shown to be a very effective technique, especially for short user queries.

D. Document Clustering

Many other techniques have been developed over the years with varying degree of success. This is a process of grouping similar documents together to perform the task of IR fast and efficiently. It is just one of several ways of organizing documents to facilitate retrieval from large databases. Clustering hypothesis states that documents that cluster together (are very similar to each other) will have a similar relevance profile for a given query (Griffiths and Steyyers, 2004). Document clustering techniques were (and still are) an active area of research. Though the usefulness of document clustering for improved search effectiveness (or efficiency) has been very limited, document clustering has allowed several developments in IR, e.g., for browsing and search interfaces.

During the IR and ranking process, two classes of similarity measures must be considered: (i) the similarity of a document and a query; and (ii) the similarity of two documents in a database. The similarity of two documents is important for identifying groups of documents in a database that can be retrieved and processed together for a user input query. Serizawa and Kobayashi (2013) opine that several important points should be considered in the development and implementation of algorithms for clustering documents in very large databases. These include identifying relevant attributes of documents and determining appropriate weights for each attribute; selecting an appropriate clustering method and similarity measure; estimating limitations on computational and memory resources; evaluating the reliability and speed of the retrieved results; facilitating changes or updates in the database, taking into account the rate and extent of the changes; and selecting an appropriate search algorithm for retrieval and ranking. This final point is of particularly great concern for Web-based searches. Serizawa and Kobayashi (2013) further stress further that there are two main categories of clustering: *hierarchical* and *non-hierarchical*. Hierarchical methods show greater promise for enhancing Internet search and retrieval systems. Although details of clustering algorithms used by major search engines are not publicly available, some general approaches are known. For instance, Digital Equipment Corporation's Web search engine, AltaVista, is based on clustering. Anick (2003) explore how to combine results from latent semantic indexing and analysis of phrases for context-based information retrieval on the Web.

E Natural Language Processing (NLP)

NLP has also been proposed as a tool to enhance retrieval effectiveness but with very limited success (Strzalkowski et al, 1997). Despite that document ranking is a critical application for IR, it is definitely not the only application. The field has developed techniques to attack many different problems like information filtering (Belkin and Croft (1992), topic detection and tracking (or TDT) (Allan et al, 2000), speech retrieval ((Sparck et al, 2000), cross-language retrieval (Grefenstette, 1998), question answering (Pasca and Harabagiu, 2001), and many more.

F. Indexing

The term "indexing" is used in the same spirit in the context of retrieval and ranking has a specific meaning. Some definitions proposed by experts are "a collection of terms with pointers to places where information about documents can be found" (Manber. 1999). Indexing is building a data structure that will allow quick searching of the text (Baeza-Yates and Ribeiro-Neto, 1999) or the act of assigning index terms to documents, which are the objects to be retrieved (Korfhage, 1997). Serizawa and Kobayashi (2013) identified four approaches to indexing documents on the Web which are (1) human or manual indexing; (2) automatic indexing; (3) intelligent or agent-based indexing; and (4) metadata, resource description framework (RDF), and annotation-based indexing. The first two appear in many classical texts, while the latter two are relatively new and promising areas of study. However, the development of effective indexing tools to aid in filtering is another major class of problems associated with Web-based search and retrieval.

There are several popular IR indexing techniques, including inverted indices and signature files Sharma and Patel, 2013).

i. Signature file

In signature file method, each document yields a bit string (signature) using hashing on its words and superimposed coding. The resulting document signatures are stored sequentially in a separate file called *signature file*, which is much smaller than the original file, and can be searched much faster (Foloutsos and Oard, 1995).

ii. Inversion indices

Each document can be represented by a list of keywords which describe the contents of the document for retrieval purposes (Foloutsos and Oard, 1995). Fast retrieval can be achieved if we invert on those keywords. The keywords are stored, e.g. alphabetically; in the index file for each keyword we maintain a list of pointers to the qualifying documents in the postings file. This method is followed by almost all the commercial systems Salton and McGil (1986).

IV. Areas of Application of Information Retrieval Techniques

IR systems were firstly developed to help manage the huge amount of information. Many universities, corporate, and public libraries now use IR systems to provide access to books, journals, and other documents. Information retrieval is used today in many applications (Sharma and Patel, 2013). Hence, the application IR techniques cannot be overemphasized in the current dispensation with respect to Information technology (IT). The general and domain-specific applications of IR techniques are as highlighted below.

A. General Areas of Application IR Techniques

(i) *Digital Libraries*: A special library with collection of digital objects that can include text, visual material, audio material, video material, stored as electronic media formats (as opposed to print, microform, or other media), along with means for organizing, storing, and retrieving the files and media contained in the library collection. The digital content may be stored locally, or accessed remotely via computer networks. An electronic library is a type of IR.

(ii) *Information filtering system:* This system uses automated or computerized methods to remove redundant or unwanted information from an information stream before presenting it to human user. This is to enhance the management of the information overload and increment of the semantic signal-to-noise ratio by comparing user's profile to some reference characteristics. These characteristics may originate from the information item (the content-based approach) or the user's social environment (the collaborative filtering approach). This is applicable in the field of email spam filters necessitated by online information explosion. Recommender systems and content discovery platforms are examples of active information filtering systems that attempt to present to the user information items (e.g. film, television, music, books, news, web pages) the user is interested in.

iii. *Media Search*: This involves computer systems for blog and news searches as well as image, 3D, music, speech and video retrieval processes. An image retrieval system is used to browse, search and retrieve images from a large database of digital images. To search for images, a user may provide query terms such as keyword, image file/link, or click on some image, and the system will return images "similar" to the query. The similarity used for search criteria could be meta tags, colour distribution in images, region/shape attributes, etc. Music information retrieval is the interdisciplinary science of retrieving information from music.

iv. *Search Engine*: IR system users through search engine to obtain their information needs. Search engine includes site, desktop, enterprise, mobile, social and web searches. *Desktop search* tools search within a user's own computer files as opposed to searching the Internet including web browser history, e-mail archives, text documents, sound files, images, and video with the advantage of searching results displayed quickly due to the use of proper indexes. *Social Search* engine mainly searches user-generated content such as news, videos and images related search queries on social media like Facebook, LinkedIn, Twitter, Instagram and Flickr. It is an enhanced version of web search that combines traditional algorithms. *Mobile search* is an evolving branch of information retrieval services that is centered on the convergence of mobile platforms and mobile phones, or that it can be used to tell information about something and other mobile devices. Web search engine ability in a mobile form allows users to find mobile content on websites which are available to mobile devices on mobile networks. As this happens, mobile content shows a media shift toward mobile multimedia. *Enterprise search* is the information search software within an enterprise (though the search function and its results may still be public). In contrary to web search, enterprise search applies search technology to documents on the open web and desktop search that applies search technology to the content on a single computer.

B. Domain-Specific Application IR Techniques

Domain-specific areas of application of IR techniques include geographic information retrieval, information retrieval for chemical structures, information retrieval in software engineering, legal information retrieval and vertical search.

V. Conclusion

IR is an art of searching and retrieving the relevant knowledge-based information from document collections with the help of query. Based on different IR models, diverse techniques were developed by researchers and IR developers. It is obvious that these IR techniques focus on yielding mostly relevant document(s) from the corpus to satisfy user’s information needs. This paper has clearly shown that these techniques have been the success factors to effective storage, management and retrieval from huge amount of data available to users of IR systems. The applications of these retrieval techniques are capable of increasing the throughput and minimize the access time of information needs.

REFERENCES

[1] R. Sagayam, S. Srinivasan, and S. Roshni, “A Survey of Text Mining: Retrieval, Extraction and Indexing Techniques,” *International Journal of Computational Engineering Research*, vol. 2, issue. 5, pp. 1443-1444, 2012.

[2] M. Sharma and R. Patel, “A survey on information retrieval models, techniques and applications,” *International Journal of Emerging Technology and Advanced Engineering*, vol. 3, issue 11, pp. 542-545, 2013

[3] D. Hiemstra, *Information Retrieval Models*, published in Goker, A., and Davies, *J. Information Retrieval*: Searching in the 21st Century. John Wiley and Sons, Ltd., ISBN-13: 978-0470027622, 2009.

[4] A. Singhal, “Modern information retrieval: A brief overview,” *Bulletin of the IEEE Computer Society Technical Committee on Data Engineering*, 2001

[5] A. A. Alhenshiri, “Web information retrieval and search engines techniques,” *Al-Satil Journal*, pp. 55-92, 2013

[6] G. Salton and M. J. McGill, *Introduction to Modern Information Retrieval*, McGraw-Hill, Inc., New York, NY, USA, 1986.

[7] T. Samar, A. Bellogin, and A. de Vries, “The strange case of reproducibility vs. representativeness in contextual suggestion test collections,” *Information Retrieval Journal*, vol. 19, issue 3, pp. 230-255, 2016.

[8] G. Salton and C. Buckley, “Term-weighting approaches in automatic text retrieval,” *Information Processing and Management*, vol. 24, issue 5, pp. 513-523, 1988.

[9] S. E. Robertson, “On term selection for query expansion,” *Journal of Documentation*, vol. 46, no. 4, pp. 359–364, 1990.

[10] S. E. Robertson and K. Sparck Jones, “Relevance weighting of search terms,” *J. Amer. Soc. Info. Science*, vol. 27, issue 3, pp. 129–146, 1976.

[11] D. Hiemstra and A. P. de Vries. “Relating the new language models of information retrieval to the traditional retrieval models,” published as CTIT technical report TR-CTIT-00-09, 2000.

[12] G. Salton and C. Buckley “Term-weighting approaches in automatic text retrieval,” *Information Processing & Management*, vol. 24, issue 5, pp. 513-523, 1988

[13] J. Ropero, A. Gómez, A. Carrasco, and C. León, “A fuzzy logic intelligent agent for information extraction: introducing a new fuzzy logic-based term weighting scheme,” *Expert Systems with Applications*, vol. 39, issue 4, pp. 4567–4581, 2012.

[14] S. E. Robertson, S. Walker, and M. Beaulieu, “Okapi at TREC–7: automatic ad hoc, filtering, VLC and filtering tracks,” in *Proceedings of the Seventh Text REtrieval Conference (TREC-7)*, pp. 253–264, NIST Special Publication 500-242, 1999.

[15] A. R. Rivas, E. L. Iglesias, and L. Borrajo, “Study of query expansion techniques and their application in the biomedical information retrieval,” *The Scientific World Journal*, Vol. 2014, Article ID 132158 pp. 1-10, 2014.

[16] C. Buckley, G. Salton, G. Allan, and A. Singhal, “Automatic query expansion using smart: TREC3,” in *Proceedings of the 3rd Text REtrieval Conference (TREC-3)*, NIST Special Publication 500–226, 1995.

[17] T. L. Griffiths and M. Steyvers “Finding scientific topics,” *Proceedings of the National Academy of Sciences*, vol. 101, suppl. 1, pp. 5228–5235, 2004.

[18] M. Serizawa and I. Kobayashi, “A study on query expansion based on topic distributions of retrieved documents,” In A. Gelbukh, Editor, *Computational Linguistics and Intelligent Text Processing*, vol. 7817 of *Lecture Notes in Computer Science*, pp. 369–379. Springer Berlin Heidelberg., 2013.

[19] P. Anick, “Using terminological feedback for web search refinement: a log-based study,” in *Proceedings of the 26th Annual ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 88-95, 2003.

[20] T. Strzalkowski, L. Guthrie, J. Karlgren, J. Leistensnider, F. Lin, J. Perez-Carballo, T. Straszheim, J. Wang, and J. Wilding, “Natural language information retrieval: TREC-5 report,” in *Proceedings of the Fifth Text REtrieval Conference (TREC-5*), 1997.

[21] N. J. Belkin and W. B. Croft, “Information filtering and information retrieval: Two sides of the same coin?,” *Communications of the ACM*, vol. 35, issue 12, pp. 29-38, 1992.

[22] J. Allan, M. E. Connell, W. B. Croft, F. Feng, D. Fisher, and X. Li, “NQUERY and TREC-9,” in *Proceedings of the 9th Text REtrieval Conference*, 2000.

[23] K. Sparck Jones, S. Walker, and S. E. Robertson, “A probabilistic model of information retrieval: development and comparative experiments,” *Inf. Process. Manage*., vol. 36, issue 6, pp. 779-808, 2000.

[24] G. Grefenstette, *Cross-Language Information Retrieval*, Kluwer Academic Publishers, 1998.

[25] M. A. Pasca and S. M. Harabagiu, “High performance question/answering,” in *Proceedings of the 24th International Conference on Research and Development in Information Retrieval*, pp. 366-374, 2001.

[26] Manber, U. Foreword. In Modern Information Retrieval, R. Baeza-Yates and B. Ribeiro-Neto. Addison-Wesley, Reading, MA, 5-8, 1999.

[27] R. A. Baeza-Yates and B. Ribeiro-Neto, *Modern Information Retrieval*, Addison-Wesley Longman, 1999.

[28] R. R. Korfhage, *Information Storage and Retrieval*, John Wiley and Sons, Inc., New York, NY, 1997.

[29] C. Faloutsos and D. W. Oard, “A survey of information retrieval and filtering methods,” *CS-TR-3514*, 1995.