INFORMATION RETRIEVALS TRIES TREE VS INVERTED FILE WORD METHOD FOR ARABIC LANGUAGE

[1]Dr. KHALAF KHATATNEH, [2]IMAN HUSSEIN

Prince Abdu Allah Bin Ghazi for IT, Al-Balqa Applied University Salt, Jordan

ABSTRACT

One of the key challenges of managing very huge volumes of data in scalable Information retrieval systems is providing fast access through keyword searches. The major data structure in the information retrieval system is an inverted file, which records the positions of each term in the documents. When the information set substantially grows, the number of terms and documents are significantly increased as well as the size of the inverted files.

In this research we implement to techniques of inverted file -posting list and Tries trees- on Arabic Language, then we will try to state the optimum technique to store the vocabulary according to the speed of retrieving queries from the collection of documents.

Keywords: *Inverted File(Word Method),Tries Tree (B+ tree). Information Retrieval, Posting List, Indexing, Index Terms, Depth-First Algorithm, Vocabulary, Stopword.*

1. INTRODUCTION

the information retrieval its Aims to retrieve information stored large amount of data access and contribute to it, as fast as possible to contain information on more than one form of the visual and textual or audio information retrieval system is a system to store information to be available for future use. Most actual information system, systems store and enable the retrieval of only textual. The user can to retrieve the information through the query, where the information retrieval system to retrieve all theinformation that "satisfy" the query, which is in contrast to database systems. Information retrieval not return a definite answer, but produce a ranking of documents that seem tocontain information relevant to the query given to the system, queries and documents in natural language based system, information retrieval on the division of each word. This process is called indexing and During indexing stops words ("the", "but", "and", etc.) are discarded, and suffixes are removed, where the words that the system remained divided into two terms to be queries and documents[7].

Several different types of index have been expressed. The most well-organized index structure for text query evaluation is the Inverted File:a collection of lists, one per term, recording the identifiers of the documents containing that term[20].

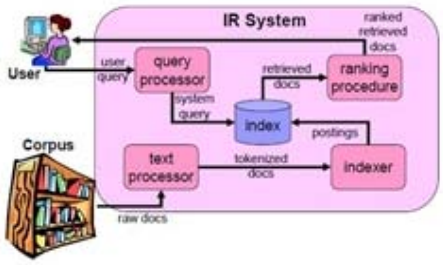
An elective structure for storing vocabularies is a B+ tree. The high branching factor typical of these trees means that the internal nodes are only a small percentage of the total vocabulary size. For example, suppose that in a B+-tree leaves contain pointers to inverted lists that the vocabulary of some database contains 1,000,000 distinct 12-byte terms, and that the disk being used operates with 8-kilobyte blocks and 4-byte pointers. Then at most 64 kilobytes is required for the internal nodes. Given this much memory, at most one disk access is required to fetch a vocabulary entry. Since the exact address of the inverted list is then known, a second access success to retrieve the corresponding inverted list [19].

When the dictionary is stored as a B-tree with a high branching factor, it takes one disk seek and one disk read to retrieve the position/length

pair for any given term . To retrieve an inverted list takes one more disk seek and one more disk read [19].a

2. INFORMATION RETRIEVAL SYSTEMS

General Scheme Figure 1 explain the essential structure of a classical information retrieval system. Through the first phase the preprocessing, the raw documents of the corpus are processed to tokenized documents and then indexed as a list of postings per terms. At the second phase the user gives a query to represent his "information need". The query is then transformed to a system query and its relevant documents are retrieved from the index. The retrieved documents are ranked according to their relevance to the query and returned to the user[5]. Many information retrieval systems include a mechanism which allows the user to provide a feedback as to the quality of the results. Using this feedback the system adapts and attempts to



produce better results for the query[5].

Figure 1: IR System Scheme

3. SEARCH ENGINES-GENERAL SCHEME

A search engine is an information retrieval system, however, It is not identical to the classical IR system previously described. The dierences between classical IR systems and search engines derive from the dierence in the source of the corpus, i.e.,a well dened closed archive such as a library as opposed to the World Wide Web. Since there is no direct access to the documents on the Web (as there is in a library corpus), a crawleris needed.This software component is responsible for fetching Web pages and storing them in a local repository. The crawling mechanism poses technological challenges related to the eciency of the process and to the relevance of the documents - as Web pages are dynamic, the crawler should keep the repository up to date. Crawling the documents out of the Web is not enough as the Web data includes many redundant information. The global analyzer is responsible to eliminatedispensable data such as duplicate Web pages and pages that include pornography. In addition, the global analyzer is responsible for global calculations used by the information retrievalsystem such as page ranking (the rank of a page is determinedmostly by pages that have links to it and pages it has links to)[5].

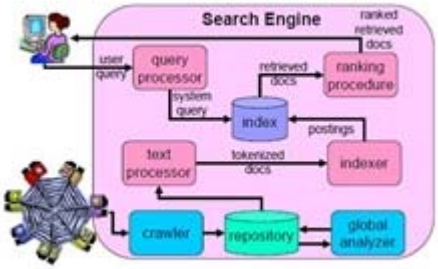


Figure 2 explain the basic structure of search engines.

4. CLASSICAL IR VS. WEB IR

The table in Figure 3 presents the main differences between classical information retrieval systems and Web information retrieval systems[5].

	Classical IR	Web IR
Volume	Large	Huge
Data quality	Clean, no dups	Noisy, dups
Data change rate	Infrequent	In flux
Data accessibility	Accessible	Partially accessible
Format diversity	Homogeneous	Widely diverse
Documents	Text	HTML
# of matches	Small	Large
IR techniques	Content-based	Link-based

Figure 3: Classical IR Vs. Web IR

5. INFORMATION RETRIEVAL VS. DATA RETRIEVAL

An information retrieval system is not a data retrieval system. Figure 4 presents some of the distinguishing properties of data and information retrieval systems[5].

	Information Retrieval	Data Retrieval
Data	Free text, unstructured	Database tables, structured
Queries	Keywords, Natural language	SQL, Relational algebras
Results	Approximate matches	Exact matches
Results	Ordered by relevance	Unordered
Accessibility	Non-expert humans	Knowledgeable users or automatic processes

Figure 4: Information Retrieval Vs. Data Retrieval

6. INFORMATIO RETRIEVAL ON ARABIC LANGUAGE

Researching in English and other languages are rich in the field of Information retrieval but those written in Arabic are very poor. Here lies the importance of our study, to take part in the development of the Arabic Language to be a main participant in this age of information. The Arabic language is our identity and our path to the genuine understanding of this age. It Is also the way to achieve an Arabic scientific development that takes part in building the modern civilization.

There are many researchers discuss field information retrieval system.in this section is show some existing potential work in IR on Arabic language .

Mohammed Aljlayl and Ophir Frieder [3] have compared MT (Machine Translation) and MRD (Machine-Readable Dictionary) to Arabic-English CLIR (Cross-Language Information Retrieval).The most important technique in CLIP for query translation from one language retrieve relevant documents in other languages is MT and MRD. They present three methods (Every-Match (EM) method, First Match (FM) method, Two Phase (TP) method) of query translation using a Bilingual dictionary for Arabic-English CLIR .they achieved roughly half of the performance of the monolingual retrieval. They display that good retrieval success can be achieved without complex resources using a new method called Two-Phase method for Arabic-English CLIR.

Mohammed Aljlayl and Ophir Frieder [4] have using two stemming algorithms for Arabic information retrieval system (The root algorithm which is measured as an aggressive stemmer and surface-based (no stemming) approach). They planned and experimented with a novel stemming algorithm called light stemming (LS) for resolve the ambiguity associated with the root algorithm. LS is measured a non-aggressive stemmer. This approach is primarily based on suffix and prefix removal and normalization. The LS algorithm significantly outperforms the root algorithm. They found 87.4% and 24.1% increase in average precision over the Arabic surface form and root algorithm.

Leah S. Larkey and Margaret E. Connell [18] have put three monolingual runs and one cross-language run. They first explain the models, techniques, and resources they used, then they explain each run in detail. They run performed moderately well, in the second tier (3rd or 4th place). Since give these results, they have increased in quality normalization and stemming, increased in quality dictionary construction, widened Arabic queries, made better estimation and smoothing in language models.

Aitao Chen and F Gey[6] have performed one Arabic monolingual run (BKYMON)and three English-Arabic cross-language retrieval runs(BKYCL1, BKYCL2, and BKYCL3), all being automatic. They took the approach of translating queries into document language using two machine translation systems. Best cross-language retrieval run obtain 87.94% of the monolingual retrieval performance. Advanced one MT-based Arabic stemmer and one light Arabic stemmer. The Berkeley light stemmer process better than the automatically created MT-based stemmer. The exploratory results give query expansion largely increased in quality the retrieval performance.

Leah S. Larkey et al.[11] have advanced several lightstemmers based on heuristics and a statistical stemmer based on co-occurrence for Arabic retrieval system .They compare the retrieval efficiency of stemmers and of a morphological analyzer on the TREC-2001(For the 2001 Text Retrieval Conference)data. The best light stemmer was more efficient for cross-language retrieval than a morphological stemmer which tries to get the

root for each word. But still poorer to good light stemming or morphological analysis.

Leah S. Larkey et al. [12] have explain problem of Proper names for cross language information retrieval. Standard bilingual dictionaries classically have poor treatment of proper names. Then again, IR mission involving news corpora, like TDT and TREC (Text Retrieval Conference) cross language IR, have Proper names at their core. In this paper, they prove the importance of proper names in one such mission, the TREC 2002 (Arabic-English) cross language track, by viewing that performance degrades a large amount (50%) when the bilingual lexicons do not have proper names. They then check several different sources of proper name translations from English to Arabic, both static and generative (transliteration) and explore their effectiveness in the context of the TREC 2002 cross language IR task. They provide for a conclusion that a combination of static translation resources add transliteration provides a successful solution.

Young-Suk Lee et al. [14] they come near Arabic's rich morphology using a model that a word consists of a sequence of morphemes in the pattern prefix*-stem-suffix* (* denotes zero or more occurrences of a morpheme). This method is sow seeds using a small manually segmented Arabic corpus and uses it to bootstrap an unsupervised algorithm to build the Arabic word segmenter from a large unsegmented Arabic corpus and uses a trigram language model to decide the most probable morpheme sequence for a given input. For improve the segmentation accuracy by use an unsupervised algorithm for automatically obtaining new stems from a 155 million word unsegmented corpus, and re-estimate the model parameters with the widened vocabulary and training corpus. The resultant Arabic word segmentation system achieves around 97% exact match accuracy on a test corpus containing 28,449 word tokens.

Nasreen AbdulJaleel and Leah S. Larkey [2] they explain problem of Out of vocabulary (OOV) words for cross language information retrieval. In the paper, they show a simple statistical technique to train an English to Arabic transliteration model from pairs of names. They call this aselected n-gram model because a two-stage training procedure first learns which n-gram segments should be added to the unigram inventory for the source language, and then a second stage learns the translation model over this inventory. This method require no heuristics or linguistic knowledge of either language. They calculate the statistically-trained model and a simpler hand-crafted model on a test set of named entities from the Arabic AFP corpus and prove that they perform better than two online translation sources.

Bassam Hammo et al. [10] they explain the design and implementation of a question answering (QA) system called QARAB. It is a system that takes natural language questions described in the Arabic language and attempts to supply short answers. The method using techniques from IR and NLP to process a collection of Arabic text documents as its primary source of knowledge.

Fredric C. Gey and Douglas W. Oard [15] In this paper, have provided an general survey of that work in a way that help readers distinguish similarity and difference in the approaches taken by the participating teams. have also wanted to discover the utility of the test collection itself, providing aggregate information about topic complexity that individual teams may find useful when interpreting their results, dentifying a potential concern regarding the completeness of the pools of documents that were judged for relevance, and illustrating a surprising insensitivity of
retrieval effectiveness to query length.

Kareem Darwish [8] has show a rapid technique of developing a shallow Arabic morphological analyzer. It will only be concerned with generating the possible roots of any given Arabic word. The analyzer is dependent upon automatically derived rules and statistics.

Hele-Mai Haav and Tanel-Lauri Lubi [16]have show for solve the problem of information overload on the web recent information retrieval apparatus need to be increased in quality. Much more "intelligence" must be inserted to search tools to manage effectively search, retrieval, filtering and presenting relevant information. This can be done by concept-based (or ontology driven) information retrieval, which is measured as one of the high-impact technologies for the next years. Anyway, most of commercial products of search and retrieval class do not report about concept-based

search features. The paper provide an overview of concept based information retrieval techniques and software tools currently available as prototypes or commercial products. apparatus are evaluated using feature classification, which incorporates general characteristics of tools and their information retrieval features.

Abduelbaset Goweder et al. [9] have developed several techniques for BP (Broken plurals) detection, and evaluated them using an unseen test set. They exist as a corporationthe BP detection component into a new light-stemming algorithm that mixes both regular and broken plurals with their singular forms. They also evaluated the new light-stemming algorithm within the context of information retrieval, comparing its performance with other stemming algorithms.

Leah S. Larkey et al. [13] have found, though, that afull solution to this problem is not necessary for helpful information retrieval. Light stemming allows extremely good information retrieval without given that correct morphological analyses. They developed more than a few light stemmers for Arabic, and valued at their effectiveness for information retrieval using standard TREC data. They have also examined in order to find similarities and differences light stemming with several stemmers based on morphologi-
cal analysis.

7. INVERTED FILE

Inverted files are the data structures employed by most modern retrieval systems to associate index terms (words, stems, phrases, big rams, etc) with document occurrences. Indexes are organized into posting lists containing several pointers which carry the correspondence information [17]. An inverted file index has two main parts: a search structure or vocabulary, containing all of the distinct values being indexed; and for each distinct value an inverted list, storing the indenters of the records containing the value[20].

Queries are evaluated by fetching the inverted lists for thequery terms, and then intersecting them for conjunctive queries and merging them for disjunctive queries. To minimize buyer space requirements, inverted lists should be fetchedin order of increasing length; thus, in a conjunctive query, the initial set of candidate answers are the records in the shortest inverted list, and processing of subsequent lists only reduces the size of this set. Once the inverted lists have been processed, the record indenters must be mapped to physical record addresses. This is achieved with an address table, which can be stored in memory or on disk [17].

The inverted file basically follows the concept of the indices used in books. Information retrieval systems usually con-struct indices for the contents of the documents to simplify the querying operations. When the users perform queries, user queries can be satisfied by returning the document pointers whose documents contain requested terms[23].

Figure 1 shows the data structure of an inverted file. The terms of the index are sorted in alphabetical order. For each term we maintain a data structure called posting, which is a list of pointers pointing to documents containingthe term, and the positions of the term appeared in those documents.

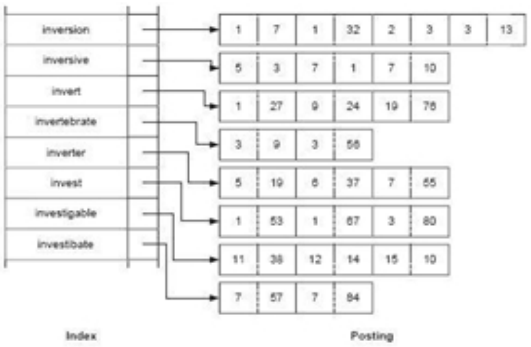


Figure 5: structure of posting with document list

8. TRIES-TREE

A tries (from retrieval),is a multi-way tree structure useful for storing strings over an alphabet. It has been used to store large dictionaries of English (say) words in spelling checking programs and in natural-language "understanding" programs. Given the data: an, ant, all, allot, alloy, aloe, are, ate, be [1]. the corresponding tries would be in figure 6 :

each moment where there is an extra something new was added today on web sites.

On the contrary, technique that using Inverted File algorithm, where information is semi-fixed and take time to cataloguing and processing and then displayed them for use to satisfied the information need(Query).

9. OUR APPROACH

9.1Text Operation

Text Operation: Processes that we will do it on text for the collection of documents: Document Pre-Processing is the process of incorporating anew document into an information retrieval system. The goal form it is

• Represent the document efficiently in terms of both space(for storing the document)and time(for processing retrieval requests)requirements.

• Maintain good retrieval performance(precision and recall).

Document Pre-Processing is complex process that leads to

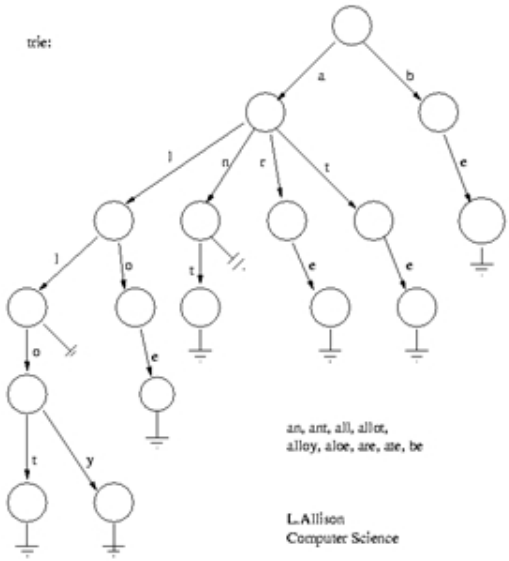


Figure 6: structure of Tries-Tree [1]

The idea is that all strings sharing a common stem or prefix hang off a common node. When the strings are words over a..z,a node has at most 27 children one for each letter plus a terminator[1]. The elements in a string can be recovered in a scan from the root to the leaf that ends a string. All strings in the trie can be recovered by a depth-first scan of the tree.

The challenges presented by text search have led to thedevelopment of a wide range of algorithms and data structures. These include representations for text indexes, index Construction techniques and algorithms for evaluation of text queries.

In this paper, we explain how to implement Tries Tree algorithm for text indexing for Arabic language, this structure useful for storing strings over an alphabet. It is providing fast access through keyword Searches.

Indexes based on these techniques are crucial to the rapid response and immediate processing provided by the major Web search engines to each something new to happen(from attach new: Web pages, newspaper articles, academic publications, company reports, research grant applications, manual pages, encyclopaedias, parliamentary proceedings, bibliographies, historical records, electronic mail and court transcripts etc.)

Tries Tree is technique to store the vocabulary (Stemming algorithms :no Root for word) according to the speed of retrieving queries from the collection of documents.

We use search technique using the Tries Tree algorithm when the information is current, in other words, variable in the representation of each document by a select set of index terms. However, some Web search engines are giving up on much of this process and index all(or virtually)the word in a document.

Document Pre-Processing includes operations:

1. Lexical analysis of the text: Identify (determine) the words in the text (document),in other words,Lexical analysis separates the input alphabet into

• Word characters (e.g., the letters a-z)

• Word separators (e.g., space, new line)

2. Elimination of stop words: Filtering out words which are too frequent among the docs in the collection are not good discriminators. A word occurring in 80% of the docs in the collection is useless for purposes of retrieval. Filtering out stop words achieves a compression of 40% size of the indexing structure.

3. Selecting of indexing terms(indexing):Increase efficiency by extracting from the resulting document a select set of terms to be used for indexing the document.

9.2 Description Of Tries Tree Algorithm In Our Approach

1. Work Text Operation.

2. Storage keywords with a tree and is represented it in two dimensional matrix, in other words, indexing of keywords with a tree (The result of the Tries Tree is a list of terms (key word) that represent documents in the collection. In order to facilitate searching these terms in an efficient way, an index is created).

3. When the user makes a query, represented a keyword for the query by tree, it is match with the keywords stored in the matrix.

This algorithm is stored on the server search engines and when used as we have already said previously dealt with the query immediately for(indexing and other operations).for return all relevant documents (at modern).

9.3 Description of our Approach programming

Our approach to compare this two method (Tries Tree and Inverted File) was the following steps for both methods:

1. Insert all stop word in STOPWORD table.
Stop words are those words that are repeated in every document, so they are considered as weak to be distinguished, we can not distinguish the content of a text depending on them [22]. There are other benefits from eliminating stopword as "shortening indexing structure" [21] and are useful in making the process faster and doesn't have information Retrieval and the degree of the efficiency of recalling system.[21] It doesn't also burden the system with unnecessary information
(Swaine',1994)

It is not clear which words can be considered stopwords and which cannot. Traditional methods consider that words that are repeated many times are stop words, but there are some words that are repeated in a certain document and considered as important words "indexing terms". But when the subjects are more specialized, as to say a subject specialized in database. Then the use of repeated words, as "index terms" such as the word "computer" are useless to be "index terms"[23].

The following table shown of STOPWORD in the figures below :

المنصرم	الذين	إنكما	إذ	أي	أن	أخر
انعم	الرغم	إننا	إذا	أيا	أنا	أبدا
انس	السابق	إنه	إطلاقا	أية	أنا	أحد
بأقل	الضرورة	إنها	إلا	أيضا	أنت	أحيانا
بإمكان	الضروري	إنهم	إلى	أين	أنتما	أخرى
بإمكانكم	الغير	إنهما	إلى	أينك	أنفسهم	أخيرا
بإمكانما	القادم	إني	إلك	أينكم	أنفسهن	أشباه
بإمكانه	القول	احتمال	إلىكم	أينكما	أنك	ألا
بإمكانهم	اللاتي	احتمالا	إلىكما	أينكن	أنكم	أما
بإمكانهن	اللاحق	احتمالات	إلىكن	أيها	أنكما	أمام
بإمكاني	اللتان	اقل	إليما	أيه	أنكن	أمامك
بالإضافة	اللتين	اكثر	إليه	أيها	أنما	أمامكم
بالإمكان	اللذان	الأن	إليها	أيهم	أنني	أمامكما
بالتأكيد	اللذين	الأمام	إليهم	أيهما	أنه	أمامكن
بالرغم	اللواتي	الأمر	إليهما	أينهن	أنها	أمامنا
بالضبط	الماضي	الاحتمال	إليهن	أيني	أنهم	أمامه
بالمسة	المحتمل	الاحتمالات	إما	أيها	أنهما	أمامها
بالمسة	المدة	الذي	إن	أيهم	أني	أمامهم
برغم	المزيد	الجاري	إنا	أيهما	أو	أمامهما
بسبب	المقبل	الحالي	إنك	أينهن	أواخر	أمامهن
بسرعة	المكان	الذي	إنكم	إحدى	أولئك	أمامي

Figure 7: STOPWORD table

لديدا	كلانا	قبل	فيها	فجاة	صنكما	عليدا
لديه	كلاهما	كأن	فيهم	فجاة	صنه	عليه
لديها	كلتا	كأنكم	فيهما	فقط	صنها	عليها
لديهم	كلكم	كأننا	فيهن	فوق	صنهم	عليهم
لديهما	كلتا	كأنه	قال	فوقك	صنهما	عليهما
لديهما	كله	كأنها	قالا	فوقكم	صنهن	عليهن
لديهن	كلها	كأنهما	قالت	فوقكما	صني	عن
لذا	كلهم	كأنهما	قالوا	فوقكن	صير	عنا
لذلك	كلهن	كأنهن	قد	فوقنا	صيرك	عند
لست	كليدا	كأني	قديما	فوقه	صيركم	عندئذ
لستم	كليهما	كافيا	قريبا	فوقها	صيركما	عندك
لستما	كي	كان	قلت	فوقهم	صيركن	عندكم
لستن	كيف	كانا	قول	فوقهما	صيرنا	عندكما
لسن	كيلا	كانت	قولا	فوقهن	صيره	عندنا
لعل	لئلا	كانتا	قولك	في	صيرها	عنده
لك	لا	كانوا	قولكم	فيك	صيرهم	عندها
لكل	لدى	كثير	قولكما	فيكم	صيرهما	عندهم
لكلا	لدي	كثيرا	قولنا	فيكن	صيرهن	عندهما
لكلتا	لديك	كذلك	قوله	فيما	صيري	عندهن
لكم	لديكم	كل	قولها	فينا	فاحتسال	عنك
لكما	لديكما	كلا	قولهما	فيه	فالضرورة	عنكم

Figure 8: STOPWORD table

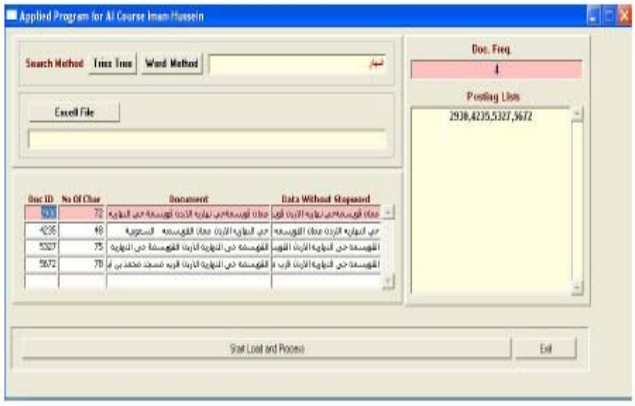
صدهما	ذي	خلفكما	حولنا	نقولان	بهما	بسهولة
صدهن	ذيك	خلفكن	حولهم	نقولوا	بهن	بشأن
صدي	رسا	خلفنا	حولهما	نقولون	بواسطة	بصعوبة
صدين	رعم	خلفه	حولهن	تكون	بي	بضع
ضرورة	رعما	خلفها	حولي	تكون	بين	بضعة
ضروري	سواء	خلفهم	حيث	تكونان	بيك	بعد
ضروريا	سواءا	خلفهما	حينما	تكونوا	بيكم	بعدئذ
طالما	سوف	خلفهن	حين	تكونون	بينكما	بك
طويلا	شيء	خلفي	حينئذ	تلك	بينكن	بكل
عامة	شيئا	دائما	حينا	تيك	بينما	بكم
عدا	شيئان	داخلا	حينذاك	تيك	بينما	بكما
عدة	شيئين	دون	حينما	ثم	بيه	بكن
عدم	ضد	دونما	حينه	جدا	بينها	بل
على الإطلاق	ضدك	دونه	حينها	جميعا	بينهم	بلى
على السواء	ضدكما	دونها	خارجا	حاشا	بينهما	بما
عليّ	ضدكن	دونهما	خاصة	حالما	بينهن	بمفرده
عليك	ضدنا	دونهن	خلال	حول	تحت	بنسبة
عليكم	ضده	ذا	خلف	حولك	تقريبا	به
عليكما	ضدها	ذاك	خلفك	حولكم	تلقاء	بها
عليكن	ضدهم	ذلك	خلفكم	حولكن	تلقاء	بهم

Figure 9: STOPWORD table

لكن	ما	معكن	مسكن	نفسهم	وحدكما	يكونوا
لكي	ماذا	معنا	مسكنا	نفسهما	وحدكن	يكونون
للضرورة	متى	معه	من	هؤلاء	وحدنا	يمكن
لم	مثل	معها	ما	هاتان	وحده	يمكنك
لما	متلا	معهم	مذ	هاتين	وحدها	يمكنكم
لمدة	مثلما	معهما	مائة	هتان	وحدهم	يمكنكما
لمن	محتمل	معهن	منكم	هتين	وحدهما	يمكنكن
لنا	محتملات	معي	منكما	هذا	وحدهن	يمكننا
له	محتملان	مكان	منكن	هذان	وحدي	يمكنني
لها	محتملة	مكانك	منه	هذه	وقتئذ	يمكنه
لهذا	محتملين	مكانكم	منها	هذين	يا	يمكنها
لهم	مدة	مكانكما	منهم	هل	يبدو	يمكنهم
لهما	مزيد	مكانكن	منهما	هم	يجب	يمكنهما
لهن	مزيدا	مكاننا	منهن	هما	يقل	يمكنهن
لو	مطلقا	مكانه	مني	هن	يقول	يومئذ
لولا	مع	مكانها	نتيجة	هنا	يقولا	
لي	مع أن	مكانهم	نحن	هناك	يقولان	
ليس	معا	مكانهما	نحو	هو	يقولوا	
ليست	معك	مكانهن	نفس	هي	يقولون	
ليسوا	معكم	مكاني	نفسه	وحدك	يكون	
مؤكدا	معكما	مما	نفسها	وحدكم	يكونان	

Figure 10: STOPWORD table

2. Insert the entire document in the DOC table (12000) document.
3. Make trim for all documents to remove unlike spaces.
4. Calculate document character length.
5. Remove numeric values from the text.
6. Remove stop word using stopword table and _ll WORDS table for the inverted _le method and CHARACTERS table for tries tree method.



9.4 Results Of Our Approach Programming

Then make small application to test the speed of information retrieval speed for documents and the result was as the following: -

1. The Collection Documents 12000 The Tries Tree Was Faster
Three Time Than Normal Inverted Le Word Method.
2. The collection documents 1500 both of them approximately speed time are equal.
3. The collection list less than 1000 document the inverted le was faster than the tries tree method.

4. Tries tree takes double time more than inverted le for initiate CHARACTERS table and inverted le word method take only one unit of time to initiate WORDS table.

10. CONCLUSIONS

After reading and study each method and implement them we recommend to use inverted le word method if the collection set is less than or equal 1000 document, and we recommend to use the tries tree method if the collection set is larger

than 1500 document, if the collection set is Arabic collection document.

11. FUTURE WORK

Many techniques are used to compress the index.Stopping and stemming techniques reduce the number of terms used in the index, and thus reduce the index size [20].

"Stemming algorithms are used in information retrieval to reduce di_erent variants of the same word with di_erent endings to a common stem .Stemmers can help information retrieval systems by unifying vocabulary, reducing term variants, reducing storage space, and increasing the likelihood of matching documents"[20] .

We recommend to application stemming techniques in Information Retrievals Tries Tree for Arabic language, one of the main reasons behind using such a method is reduce the index size.

We expect the results be as follows:

_ More relevant and retrieval document.

_ Need more processing .

_ Less memory.

REFERENCES:

[1] http://www.csse.monash.edu.au/lloyd/tildealgds/tree/trie/

[2] Nasreen AbdulJaleel and Leah S. Larkey. Statistical transliteration for english-arabic cross language information retrieval. international conference on Information, 2003.

[3] Mohammed Aljlayl and Ophir Frieder. E ective arabicenglish cross-language information retrieval via machinereadable dictionaries and machine translation. tenth international conference on Information ...., 2001.

[4] Mohammed Aljlayl and Ophir Frieder. On arabic search:
Improving the retrieval effectiveness via a light stemming approach. international conference on Information, 2002.

[5] Ziv Bar-Yossef. Information retrieval,lecture 2,algorithms for large data sets. Spring 2005.

[6] Aitao Chen and F Gey. Building an arabic stemmer for information retrieval. 2003.

[7] F. Crestani. Application of spreading activation techniques in information retrieval. Articial Intelligence Review, 1997.

[8] Kareem Darwish. Building a shallow arabic morphological analyzer in one day. 2003.

[9] Abduelbaset Goweder et al. Identifying broken pluralsin unvowelised arabic text. 2001.

[10] Bassam Hammo et al. Qarab: A question answeringsystem to support the arabic language. 2002.

[11] Leah S. Larkey et al. Improving stemming for arabicinformation retrieval: Light stemming and co-occurrenceanalysis. 2002.

[12] Leah S. Larkey et al. Whats in a name?: Proper names in arabic cross language information retrieval. 2003.

[13] Leah S. Larkey et al. Light stemming for arabic information retrieval. 2007.

[10] Bassam Hammo et al. Qarab: A question answering system to support the arabic language. 2002.

[11] Leah S. Larkey et al. Improving stemming for Arabic information retrieval: Light stemming and co-occurrence analysis. 2002.

[12] Leah S. Larkey et al. Whats in a name?: Proper names in arabic cross language information retrieval. 2003.

[13] Leah S. Larkey et al. Light stemming for arabic information retrieval. 2007.

[14] Young-Suk Lee et al. Language model based arabic word segmentation. Proceedings of the 41st Annual Meeting of the Association for Computational Linguistics, July 2003.

[15] Fredric C. Gey and Douglas W. Oard. The trec-2001cross-language information retrieval track: Searching arabic using english, french or arabic queries. 2001.

[16] Hele-Mai Haav and Tanel-Lauri Lubi. A survey of concept-based information retrieval tools on the web. 5th East-European Conference, 2001.

[17] Chung-Hung Lai and Tien-Fu Chen. Compressing inverted files in scalable information systems by binary decision diagram encoding. ACM/IEEE 2001 Conference, 2001.

[18] Leah S. Larkey and Margaret E. Connell. Arabic information retrieval at umass in trec-10. 2006.

[19] D Lucarella. A document retrieval system based on nearest neighbour searching. Journal of Information Science, 1988.

[20] Abdusalam F Ahmed Nwesri. Effective

retrieval techniques for arabic text. thesis submitted for the degree of doctor of philosophy, School of Computer Science and Information TechnologyScience, Engineering, and Technology Portfolio, RMIT University, Melbourne, Victoria, Australia., May, 2008.