Asian Journal of Information Technology 15 (22): 4617-4625, 2016
ISSN: 1682-3915

A New Efficient Approach for Multi-Language Search Engines And Formation Retrieval Systems

Safaa I. Hajeer, Rasha M. Ismail, Nagwa L. Badr and M.F. Tolba
Faculty of Computer and Formation Sciences, Ain Shams University, Cairo, Egypt

Abstract: There are billions of web pages available on the Internet. Search egines always have a challenge to find the best ranked list to the user's query from those huge numbers of pages. A lot of search results that corresponds to a user's query are not relevant to the user's needs. Most of the page ranking algorithms use lnk-based ranking (web structure) or cntent-based ranking to calculate the relevancy of the information to the user's need, but those ranking algorithms might be not enough to provide a good ranked list. So in this study we proposed an Efficient Hybrid Usage-based Ranking Algorithm called EHURA. EHURA was applied to 1033 eglish crpus abnd aabic crpus to measure its performance. The result show improvements in the recall and precision for using EHURA over the cntent-based ranking algorithm representation in both lnguages.

Key words: Information Retrieval (IR), Tokenization, usage-based ranking, content-based ranking, link-based ranking, page rank, weighted pagerank, arabic language

INTRODUCTION

The amount of information in the world is increasing exponentially over years. New books, journal articles and conference proceedings have been coming out each year. Searching within this huge amount of information becomes a critical behavior of our life. Millions of users interact with search engines daily and as a result, it is time to create the technologies that can help us sift through all the available information which is most valuable to users. This technology could be services, different kind of languages which are currently spoken around the world and services for all user's needs.

Now a days, information retrieval systems play critical roles to obtain relevant information resources for searching engines; the basic idea of information retrieval system's is based on an indexing techniques technology (Yates and Neto, 1999; Singhal, 2001).

Most of existing web search engines often calculate the relevancy of web pages for a given query by counting the search keywords contained in the web pages, this approach is called the content-based ranking algorithms that uses words in each document to determine its ranking. This approach works well when users' queries are clear and specific, however in the real world, web search queries are often short (<3 words) and ambiguous (Jiang *et al*., 2005) also web pages contain a lot diverse and noisy information. These will very likely lead to the deteriorating of the performance of web search engines, due to the gap between query space and document space. Another approach, Link-based ranking algorithms assign scores to web pages based on the number and quality of hyperlinks between pages. Links that point to a particular page or endorse a page can help to improve link-based rankings; finally, Usage-based ranking algorithms score documents by how often they are viewed by internet users. For usage-based ranking, there is limited work to utilize the usage data in the web information retrieval systems, especially in the ranking algorithm. For some systems (Ding *et al*., 2002; (Rodriguez-Mula *et al*., 1998) that do use the usage data in ranking, they determine the relevance of a web page by its selection frequency. This measurement is not that accurate to indicate the real relevance. The time spent on reading the page, the operation of saving, printing the page or adding the page to the bookmark and the action of following the links in the page are all good indicators, perhaps better than the simple selection frequency. So it is worth further exploration on how to apply this kind of actual user's behavior to the ranking mechanism.

Also, the internet speaks surprisingly few of the world's languages. English is the most frequently used language around the world on the other hand billons of internet users prefer website's that use their own language. Arabic is one of the six official languages of the united nations and the mother tongue of <360 million people (Dilekh and Behloul, 2012). The number of Arab Internet users are increasing recursively over years because of the changing for the requirements of the life. Relatively fewer arabic search engines are currently

Corresponding Author: Safaa I. Hajeer, Faculty of Computer and Formation Sciences, Ain Shams University, Cairo, Egypt

available despite the enormous efforts to satisfy the needs of the growing number of arabic internet users. Moreover, Arabic is a highly inflected language and has a complex morphological structure. That's a problem both for internet users, who are needed the content of websites pages in an understood language for them and the websites, applications and services that are trying to reach new users in emerging markets around the world.

To overcome all of the pervious problems, this study provides a hybrid ranking algorithm to utilize the usage data called EHURA (Efficient Hybrid Usage-based Ranking Algorithm). The objective of this algorithm is to improve the ranked list provided from a multi-languages search engines. The improvement is important to study because it will affect the effectiveness and the performance of the information retrieval systems and web search engines.

Literature review: Ranking search results are a fundamental problem in information retrieval. Most common approaches primarily focus on the similarity of a query and a page, as well as the overall page quality. However, with increasing popularity of search engines, the capturing of user's behaviors appears on the surface more. Much information such as; links user's click, how long users spend on a page and the user's satisfaction degree from the relevance of the page could be estimated. It is actually a kind of implicit feedback (i.e., the actions users take when interacting with the search engines), such kind of usage data could be used to improve the rankings (Konstan *et al*.,1997; Sanderson *et al*., 2010; Taherizadeh and Moghadam, 2009; Weiler, 2005) .

A lot of research has been done on the implicit measures of user's preferences in the fields of IR (i.e. implicit feedback in IR) one of the earliest evaluations of time aspects was presented by Morita. Their experiments show a positive correlation between user interests and the reading time of articles. In addition, they found a low correlation between reading time and the length and readability of an article (Hofgesang, 2006).

The usage-based ranking algorithm was presented by Ding *et al*. (2002) for web information retrieval systems that applies time spent on a page against standard selection- frequency-based ranking, i.e., the basic idea of rank score is calculated on the time users spend on reading the page and browsing the connected pages, the high-ranked pages may have a negative adjustment value if their positions couldn't match their actual usage and the low-ranked pages may have a positive adjustment value if uses tend to dig them out from low positions (Ding *et al*., 2002).

According to the study of (Kellar *et al*., 2004) that focused on the relation between web search tasks and the time spent on reading results, their results support the correlation and show that it is even stronger as the complexity of a given task increases.

Agichten studied user's behavior data to improve ordering results in a real web search setting. Their report involved over 3000 queries and 12 million user interactions with a popular web search engine, the results of this study show the accuracy of entering a user feedback term and was improved by comparing with the original one.

Kritikopoulos *et al*. studied as a method in for evaluating the quality of ranking algorithms. Success Index takes into account a user's click-through data; the results show their method is better than explicit judgment.

A comparison study was appeared on (Liu *et al*., 2010) between three methods of ranking in usage fields. Those methods are pagerank, weighted pagerank and HITS. All of those methods are focus on the structure of the page. The result of this comparison is HITS is the best.

Jain and Purohit (2011) this research was presented a method based on a combination of the click-through of pages by the users (event) and the summarization of documents. They used the advantage of implicit modeling effectively improving the user model without any extra effort of user. As a result, implicit feedback information improves the user modeling process.

Another study was presented by Rekha. This study was provided a new model to find a user's preferences from click-through behaviors and using the exposed preferences to adapt the search engine's ranking function for improving search services. In this proposed model, the combination of viewed and stored document summaries is used. The results show that this combining improved the reliability of a ranked list than ever before.

Mukherjee *et al*. (2012), presented a method to discover web knowledge for presenting web users with more personalized web content. Their method was collected usage data from different users and then finds the similarities between all pairs of users. Experimental results generate correct suggestions that retrieve relevant documents to the user (Mukherjee *et al*., 2012).

Tuteja (2003)'s was based on user behaviors in order to enhance the weighted pagerank algorithm by considering the term Visits Of Links (VOL) completed by the end of 2013. This research idea was presented as modifying the standard weighted pagerank algorithm by incorporating visits of links. The result shows that adding

the number of Visits Of Links (VOL) to calculate the values of page rank proves that the relevant results are retrieved first. In this way, it may help users to get the relevant information much quicker.

Iyakutti (2011), this research was presented a new approach and is introduced to re-order the search results based on the contents and user interests rather than keyword and page ranking that's provided by search engines and based on the user's query, search engine results are retrieved. When the user visits the web page out of this reordered list, the query, url and the contents extracted from the web page are stored in the server log. when the next time the user enters a query, the scores are awarded to each result link based on the data in the server log which indirectly incurs the user's interest.

A few researches considered in usage-based ranking based on pages' selection frequency. This might be an incorrect indicator; the reasons might be the inadvertent human mistakes, misleading titles of web pages or the returned summaries not representing the real content. As a conclusion, ranking algorithms still have some drawbacks to a ranked list provided from some search engines. So, we decide to develop a hybrid ranking algorithm to utilize the usage data, this hybrid ranking algorithm bases on content-based ranking which is the more accurate indicator instead of link-based ranking, we thought the content of the page is rather more important than how much it holds incoming links and out-coming links to a page in addition to other usage factors which are:

- Frequency of visits that determine the relevance of a web page by its selection frequency
- Time spent that shows how long users spend on a page after removing the download time of the page

METERIALS AND METHODS

The system architecture: This study discusses the hybrid approach, the basic idea of this approach is based on a new Hybrid Ranking Algorithm called EHURA algorithm.

EHURA's algorithm holds two parts: Content-based ranking which is an accurate indicator and the usage-based ranking. This hybrid approach was applied to the following system architecture as shown in Fig. 1. According to Fig. 1, the system consists of several modules, explains into three phases:

Phase 1: Document pre-processing module consists of the following Modules:

Module 1: Tokenization this stage is for breaking a stream of text into words and keeping the words in a list called a word's list.

Module 2: Data cleaning removes useless words from the word's list, these useless words are stored in a stop words database as appears in the figure. The database has 311 English stop words with a size of 3 kb; on the other hand, the database of arabic has 1459 stop words with a size 10 kb.

.

Module 3: Stemming: In this stage, we applied a hybrid affix removal algorithm (Arafat and Saad, 2008) for Arabic language and the Enhanced Porter Stemmer Algorithm (EPSA) (Hajeer *et al*., 2014) for English, they are explained in Stemming section.

Module 4: Indexing indexing is a process for describing or classifying a document by index terms; index terms are the keywords that have a meaning of their own, (i.e., which usually has the semantics of the noun). This index terms are grouped in an indexer and the stemmer is service at this stage by improving the group of these keywords in the indexer.

Phase 2: Log files analysis this phase for removing irrelevant records from log files which contain lots of it. In order to enhance the efficiency of the usage-Based retrieval algorithm by using relevant records only. This Phase consists of a series of processes like data cleaning, user identification, session identification as appears in Fig. 1. "Log Files Analysis" section for the details of this processes.

Phase 3: Ranking module (EHURA Algorithm) consists of the following modules:

Module 1: Content-basedranking; the user's query is matched with the index terms to get the relevant documents to the query. Documents are then ranked using ranking algorithms according to the most relevant to the user's query.

Module 2: Usage based parameters this stage is for calculating several parameters which is the co-operation to service and the usage based re-ranking algorithm.

Module 3: Usage-based re-ranking is the combination of the previous modules to provide a new weight called usage-based weight for pages, then ranking those pages according to their new weight.

Stemming

Arabic stemmer: Many stemmers have been developed for arabic language, Although, arabic language has a very

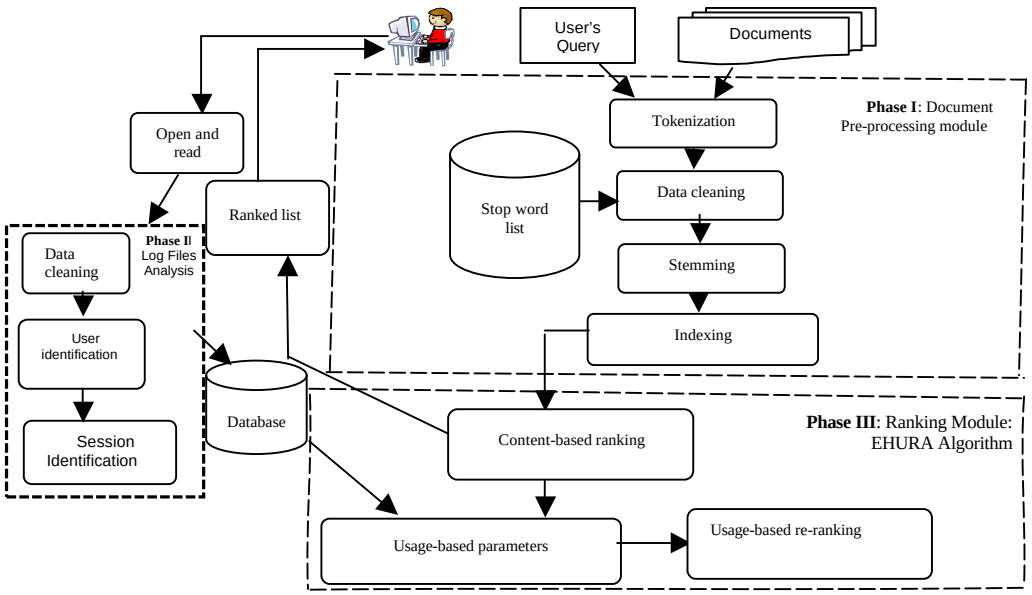

Fig. 1: The system architecture

difficult structure than other languages because of it is a rich language with complex morphology. So arabic stemmers still have many weakness and problems. The best one through our study for several kinds of them is hybrid affix removal algorithm stemmer. The hybrid stemmer proposed by Arafat and Saad (2008) is used to support the hybrid affix removal algorithm stemmer between the root-based and light stemmers. It removes the suffixes and prefixes of arabic words and in addition it returns some words to their basic roots. For example, it would change a word in the plural form to its singular form. They start by the input text documents and normalize them, after this tokenizing into words, each word enters a second stage of normalization by passing this step which removes all diacritics except in the case of the diacritic "shaddah" in which case, the stemmer will remove the "shaddah" and duplicate the letter to compensate for the shaddahs removal. Also, the stemmer removes the "alf tanween" when found at the end of a word.

The third stage unifies the Arabic letters i.e., unifies the different forms of a single letter. The arabic letters that need to be unified are:" alef"," yeh" and the marboota". For example, the different forms of the Arabic letter "alef" which are "aa, ae, a" are all replaced with "a"

The final stage in the stemming process is to remove the prefixes and suffixes attached to words. A list of Arabic prefix's and another for suffixes are kept. Each word is compared to these lists and prefixes and suffixes attached to the word are removed or sometimes replaced with other letters according to certain rules depending on the word's length.

In order to clear the word normalization, normalization process is the process that corpus and queries pass to some conditions for normalized, these conditions:

- Convert to windows arabic encoding (CP1256)
- Remove punctuation
- Remove diacritics (primarily weak vowels). Some dictionary entries contained weak vowels. Removal makes everything consistent
- Remove non letters
- Replace "alef madda, aa" and "ae" with a
- Replace final "alef maqsooura" to "yeh"
- Replace final "ta marboota" with "ha"

English stemmer: Most of the stemming experiments done so far are for english, Thus, english language is an easy language that does not have the complexities of the article. The most famous English stemmer is porter stemmer. Really, many researches tried to improve the structure of porter algorithm; however, they still have several drawbacks. Some of them concentrated on plural and singular words only, others concentrated on the semantic of some words without being careful about singular and plural. Also, the previous work didn't discuss the past tense of words ending by "ed" and the verbs ending by "en". We choose to our system the Enhanced Porter Stemming Algorithm (EPSA) to overcome these problems. ESPA stemming includes the original porter rules and our new rules that are proposed

Table 1: The EPSA rules (Hajeer *et al.*, 2014)

If the word:	Rule 1	Rule 2
Ends with "e", function must keep e at the end of the word	ends with "ize" - m=2, keep it - m>1, "ize" removed	ends with "er" after it constant then delete "r"
Ends with "ches" or with "shes"… remove "es" only	ends with "ive" - m=1, keep it - m>1, "ive" removed	If end "es", Remove "s", keep "e"
Ends with "is", don't delete	ends by "iral" , m=2, start with vowelkeep it	If end "en", Keep "e"
Ends with "ying"¬i and "yed"¬y	ends "al", m=2, delete "al" and add "e"	If the word end by "y", Replace it with "I"
m=2, consonant, vowel, consonant, vowel, then remove "al"	ends –knives, -knives¬ -knife	ends "ed" or "ing", keeping "e" while removing "ed" or "ing"
Ends by "ative" and m=2, ative¬"ate"	ends "ic" ,m=2, delete "ic"	ends¬staves, -staves¬-staff
Ends with "ness", m=1, Consonant, vowel & consonant, "ness"¬"ness"	ends "icate", delete "ate"	ends –xes, -xis_ -x
m=2, ends with "ness", ness¬	m=1, ends "ical" , "ical"¬ "ic"	ends–trixes, -trixes¬-trix
Ends "ousness", m=1, Consonant, vowel and consonant, ousess¬ ous	m> 0, Ends with "ator", Remove "ator" and replace it with "ate"	ends –ei, -ei¬-eus
m> 0, Ends with "less", "less"¬"less"	ends with "ceed", m>0, remove "ceed" and replace it with "cess"	ends –pi, -pi¬-pus
m> 0, Ends with "lessly", "lessly"¬"less"	ends –wives, -wives¬ -wife	ends ¬ses, -sis¬-s
m> 0, Ends with "fully", delete "ly"	ends –feet, -feet¬ -foot	ends with "ence", m=2, delete "ence"
Ends with "ous", m>1,Delete "ous"	ending in –men, -men¬ -man	ends with "ment", m=2, Keep it
Ends with "ous", m=2, Keep "ous"	ends –ci, ci¬-cus	ends with "ment", m>2, remove "ment"
Ends with "eer", m=2 Then remover "er"	end with "eed" -m=0, then Keep "eed" -m>0, remove "d"	ends with "tion", m=2, replaced with "e"
Ends with "ible", m=2, Starts with a consonant, not ending with series of consonant vowel consonant vowel Then keep it as it is	m> 0, Ends with "ator", Remove ator" and replace it with "ate""	ends with "ional" then delete "al"
Ends with "nate" or "ate" -m=2, keep it as it is m>2, "ate" or "nate" removed m=1, then "at" is kept - m=0, then left it as it is	m=2, ends "able", delete "able" m>2, remove "ible"	ends with "ance", m=2, Consists of series consonant,vowel, consonant, vowel , Replaced with "e", Else, removed "ance"

to solve the errors suffered from the original porter (Hajeer *et al.*, 2014). Also, we collected the critical rules from other researches (Karaa, 2013; Kara and Gribaa, 2013; Megala *et al.*, 2013; Porter, 1980) and added them to the EPSA stemming. Table 1 shows the details of these rules.

Log files analysis: The log file consists of lots of irrelevant entries which need to be removed. To enhance the efficiency of usage-based retrieval, noise must be removed before retrieving usage data. Log file analysis consists of a series of process like; data cleaning, user identification, session identification as appears in Fig. 1

Data cleaning is the process of removing unnecessary records like graphics, video and formatted information like css. In addition, this process also removes records of failed HTTP status codes.

- User identification is the process of identifying users and user agent fields of log entries, its considered on
  - Different IP addresses refer to different users
  - The same IP with different operating systems or different browsers and should be considered as a different user
  - While the IP operating system and browsers are all the same, new users can be determined by whether the requesting page can be reached by accessed pages according to the topology of the site
  - A user session is considered to be all pages accessed that occur during a single visit to a web site. In session identification is the process for defining users that may access the site more than once

Ranking module: ehura algorithm: As explained in the previous section, EHRA is a hybrid ranking algorithm that holds four modules; we will explain their works and equations on this section.

Content-based ranking: This part focuses on the ranking algorithm based on the content of documents and

query's. It simply tries to find the similarity between the content of documents and query's. We applied here the cosine similarity measure, this selection is based on studies represented in (Hajeer, 2012a, b) which proves that the cosine measure is the most efficient one in comparison to other statistical measures.

The cosine measure calculates the angel between two documents (between document and user's query which is treated as a document) representation vectors. Thus a cosine value of zero means that the query and document vector were orthogonal to each other and that means that there is no match or the term simply does not exist in the document being considered. To know cosine relation between two documents (document D and query Q) (Eq. 1):

$$\text{Cosine (D, Q)} = \frac{|D \cap Q|}{\sqrt{|D| * |Q|}} \quad (1)$$

Where:

Cosine (D, Q) = The Cosine similarity relationship between document D and user's query Q

D = Refers to the document in the collection

Q = Refers to the user's query

After calculating the similarity measure, the ranked list appears to the user as an answer of his/her query. This list is arranged from the highest value of cosine measures to the lowest one as a weight as a ranked list.

Usage-based parameters: In this stage the system calculates two usage-based parameters as the following: Frequency of visits that determine the relevance of a web page by its selection frequency in order to find the frequency weight which is the admittance frequency of a page which is the number of times the page is visited and the page rank which appears in the ranked list from the previous stage. The frequency weight equation is:

$$FW = \frac{\text{Number of visit on a page}(u)}{\text{Total number of visit on all page}} \times PR(u) \quad (2)$$

Where:

FW = Frequency weight

PR (u) = The page rank of a page u

Time spent that shows how long users spend on a page after removing the download time of the page because a user generally spends more time on a more useful page and does not waste time on screening the page and rapidly skipping to another page. So, it's an important parameter to indicate usefulness of pages, this parameter is considered to calculate the real time spent on a page by taking the value of time taken (time spent on the page) from the log file subtracting from the download time in order to find a time spent weight as the following:

$$TW = \frac{\text{TimeSpent on a page}(u) - \text{Download time}(u)}{\max\left(\begin{matrix}\text{Time spent on a page}(u) - \\ \text{Download time}(u)\end{matrix}\right)} \quad (3)$$

where, TW is time spent weight.

$$\text{Download time}(u) = \frac{\text{size of a page}(u)}{\text{Transfer rate for page}(u)} \quad (4)$$

Usage-based re-ranking: This is the final stage in our EHRA algorithm, it's basically used the two parameters that were calculated in the previous stage to find the usage-Based weight which is equal the new weight for each page, this weight is used to re-rank the pages and the effective results reflect on the previous rank list to get a new rank list, as a result a new search engine result appears to the user.

In order to see if these results are making search engines more efficient, the system tested using IR performance measures that will be explained.

Performance studies: In order to study the performance of the system, we used different evaluation measures.

Evaluation of the proposed system: This study is to evaluate the performance of our IR system and compare its results with the result gained when using the exact match (without stemming), using stemmers and using our hybrid algorithm (EHUR). Performance is measured by the recall and precision measurements and include other measures which are represented in the following Equation:

$$\text{Percision} = \frac{\left|\begin{matrix}\{\text{relevent docoments}\} \cap \\ \{\text{retrived docoments}\}\end{matrix}\right|}{\left|\{\text{retrived docoments}\}\right|} \quad (5)$$

$$\text{Recall} = \frac{\left|\begin{matrix}\{\text{relevent docoments}\} \cap \\ \{\text{retrived docoments}\}\end{matrix}\right|}{\left|\{\text{relevant docoments}\}\right|} \quad (6)$$

Fall-out is the proportion of non-relevant documents that are retrieved out of all non-relevant documents available.

$$\text{Fall out} = \frac{\left|\{\text{non relevent docoments}\} \cap \{\text{retrived docoments}\}\right|}{\left|\{\text{non} - \text{relevant docoments}\}\right|} \quad (7)$$

$$F\_\text{measure} = \frac{\left|2^{*}\text{Precision, Recall}\right|}{\left|\text{Precision, Recall}\right|} \quad (8)$$

where, F-measure is the weighted harmonic means of precision and recall:

$$\text{AveP} = \frac{\sum_{i=1}^{N_0} P_i(r)}{N_q}$$

Where:

AveP = Average precision at recall level r

$P_i(r)$ = The precision at the recall level r for the ith query

N_q = The number of queries used

RESULTS AND DISCUSSION

For testing our system, it was applied on the Ain Shams University Arabic/English corpus. The Arabic corpus belongs to the modern standard rabic type; it contains 242 documents with different sizes and we tested the system with 20 queries in order to evaluate the IR system's performance. The english corpus contains 1033 documents with different sizes and the system is tested by 30 queries in order to evaluate the IR system performance.

The log files stored 622 MB of data and we have got 323 MB of data after pre-processing. by analyzing those log files using one of the analyzer tools called deep log analyzer. Deep log analyzer did the series of processes that are explained, i.e., data cleaning, user identification and session identification with several statistics about usage data like the numbers of hits, numbers of successful hits, numbers of repeated visitors and where most visitors come from and which country by percentage and value… etc.

Our system was tested using IR evaluation measures which is mentioned in the evaluation section. For arabic corpus, Fig. 2 shows the precision and recall results for each query of the content-based algorithm in comparison with the Hybrid Algorithm (EHURA). It's clear that EHURA reached a better result than the content-based one. The average precision of our new approach

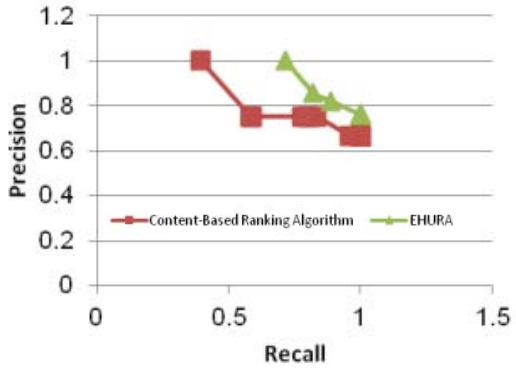

Fig. 2: Precision and r ecall for ranking against the Arabic ain Sham's corpus 20 queries

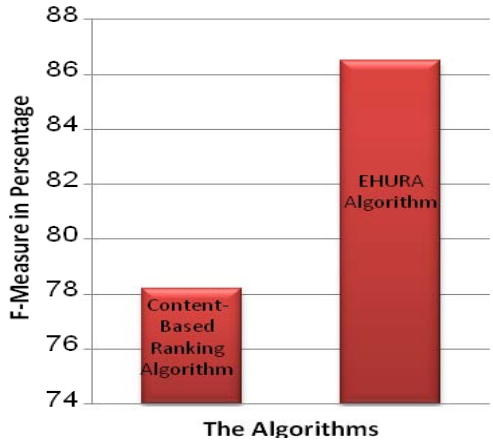

Fig. 3: F-Measure for ranking against the Arabic Ain sham's corpus

(EHURA) reached 97% while the precision of the content-based ranking algorithm is 86%, the results are shown in Table 2. So, the proposed EHURA algorithm improves the precision over the Content-Based ranking algorithm by about 10% while it also improves the recall percentage by 7%.

The proportion of non-relevant documents retrieved (Fall-out) from the system using the content-based algorithm (Hajeer, 2012a, b) reached 29% while the proposed EHURA algorithm reached 22%.

Figure 3 shows the F-measure using the content based algorithm (Hajeer, 2012a, b) and the EHURA and it's clear from the figures that the EHURA algorithm improved the F-Measure over the content-based algorithm (Hajeer, 2012a, b) by 9%.

For the English corpus, Fig. 4, shows the precision and recall results for each query with the content-based ranking algorithm and our Hybrid Usage-Based Ranking

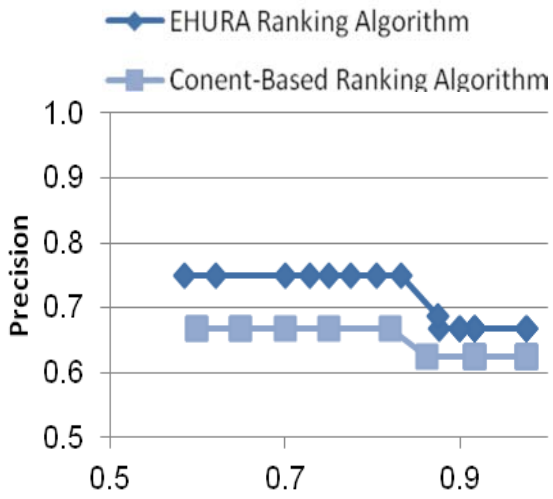

Fig. 4: Precision and recall for ranking against the english ain Sham's corpus 30 queries

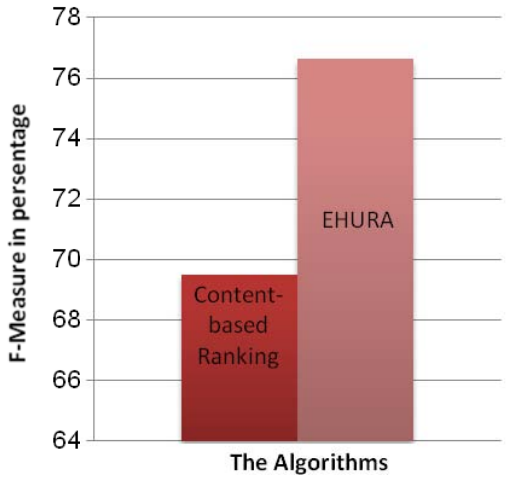

Fig. 5: F-Measure for ranking against the English Ain Sham's corpus

Table 2:Evaluation measures for the proposed system for arabic language

Variables	Precision	Recall	Fall-out	F-measure
Content-based ranking	0.8662	0.7125	0.2875	0.7819
EHURA	0.9711	0.7800	0.22	0.8651

Table 3: Evaluation measures for our system for english language

Variables	Precision	Recall	Fall-out	F-measure
Content-based Ranking algorithm	0.672	0.7199	0.2801	0.6951
EHURA algorithm	0.8198	0.7199	0.2801	0.7665

Algorithm (EHURA). The average precision for the proposed IR system using EHURA is 82% while the same system using only the content-based algorithm is nearly 67%. these results are shown in Table 3. From these results, the EHURA algorithm improves the precision over the content-based ranking by about 15% while realizing approximately the same recall percentages.

The proportion of non-relevant documents that are retrieved from the system reaches 28% on the other hand, the F-measure value is nearly 76% with the EHURA Algorithm while the content-based Algorithm realized only 69% the EHURA algorithm improves the measure over the content-based algorithm by 7% the results are shown in Fig. 5 and Table 3.

CONCLUSION

Searching becomes a normal part of our life and millions of users interact with search engines daily. Many of the existing information retrieval systems still rely on various approaches of ranking algorithms, like content-based ranking algorithms, link-based ranking, or a few of them are based on utilizing the user's behavior via usage-based ranking algorithm. Unfortunately, those ranking algorithms still have some drawbacks compared to a ranked list provided from some search engines. Thus in this study we proposed an efficient multi-language information retrieval system using a new hybrid usage-based ranking algorithm called EHURA. The objective of the EHURA algorithm is to overcome the drawbacks of ranking algorithms and improve the efficiency of web searching.

The system was applied to ain shams arabic/english corpuses for testing. The results show that the EHURA algorithm improves the performance of the information retrieval system in respect to the recall and precision measures. For arabic, the improvement of precision over the content-based ranking algorithm is about 10% while improving the recall percentage by 7%. For English, it improves the precision over the content based algorithm by about 15% while realizing approximately the same recall percentage. As a result, EHURA improves the precision over the content based algorithm for multi-language search engines by a good percentage.

REFERENCES

Arafat, S. and S. Saad, 2008. An affix removal stemming algorithm for Arabic language. Int. J. Intell. Comput. Inf. Syst., 8: 141-153.

Dilekh, T. and A. Behloul, 2012. Implementation of a new hybrid method for stemming of Arabic text. Int. J. Comput. Appl., 46: 14-19.

Ding, C., C.H. Chi and T. Luo, 2002. An improved usage-based ranking. Proceedings of the International Conference on Web-Age Information Management, August 11-13, 2002, Springer Berlin Heidelberg, Berlin, Germany, ISBN: 978-3-540-44045-1, pp: 346-353.

Hajeer, S.I., 2012a. Comparison on the effectiveness of different statistical similarity measures. Int. J. Comput. Appl., 53: 14-19.

Hajeer, S., 2012b. Vector space model: Comparison between euclidean distance and cosine measure on Arabic documents. Int. J. Eng. Res. Appl., 2: 2085-2090.

Hajeer, S.I., R.M. Ismail, N.L. Badr and M.F. Tolba, 2014. An Adaptive Information Retrieval System for Efficient Web Searching. In: Advanced Machine Learning Technologies and Applications, Aboul, E.H. F.T. Mohamed and T.A. Ahmad (Eds.). Springer International Publishing, Berlin, Germany, ISBN: 978-3-319-13460-4, pp: 472-482.

Hofgesang, P.I., 2006. Relevance of time spent on web pages. Proceedings of the KDD Workshop on Web Mining and Web Usage Analysis in Conjunction with the 12th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, August 20, 2006, ACM, Philadelphia, Pennsylvania, pp: 1-18.

Iyakutti, K., 2011. Improving the information retrieval system through effective evaluation of web page in client side analysis. Int. J. Comput. Appl., 15: 35-39.

Jain, R. and D.G. Purohit, 2011. Page ranking algorithms for web mining. Int. J. Comput. Appl., 13: 0975-8887.

Jiang, X.M., W.G. Song and H.J. Zeng, 2005. Applying associative relationship on the click through data to improve web search. Proceedings of the European Conference on Information Retrieval, March 21-23, 2005, Springer Berlin Heidelberg, Berlin, Germany, ISBN: 978-3-540-25295-5, pp: 475-486.

Karaa, W.B.A. and N. Gribaa, 2013. Information Retrieval with Porter Stemmer: A New Version for English. In: Advances in Computational Science Engineering and Information Technology, Dhinaharan, N., K. Ashok and A. Annamalai (Eds.). Springer International Publishing, Berlin, Germany, ISBN: 978-3-319-00950-6, pp: 243-254.

Karaa, W.B.A., 2013. A new stemmer to improve information retrieval. Int. J. Network Secur. Appl., 5: 143-154.

Kellar, M., C. Watters, J. Duffy and M. Shepherd, 2004. Effect of task on time spent reading as an implicit measure of interest. Proc. Am. Soc. Inf. Sci. Technol., 41: 168-175.

Konstan, J.A., B.N. Miller, D. Maltz, J.L. Herlocker and L.R. Gordon *et al*., 1997. Group Lens: Applying collaborative filtering to Usenet news. Commun. ACM., 40: 77-87.

Liu, Y., T.Y. Liu, B. Gao, Z. Ma and H. Li, 2010. A framework to compute page importance based on user behaviors. Inf. Retrieval, 13: 22-45.

Megala, S., A. Kavitha and A. Marimuthu, 2013. Improvised stemming algorithm-TWIG. Int. J. Adv. Res. Comput. Sci. Software Eng., 3: 168-171.

Mukherjee, I., V. Bhattacharya, S. Banerjee, P.K. Gupta and P.K. Mahanti, 2012. Efficient web information retrieval based on usage mining. Int. J. Sci. Res. (IJSR.), 3: 60-66.

Porter, M.F., 1980. An algorithm for suffix stripping. Program Electron. Lib. Inform. Syst., 14: 130-137.

Rodriguez-Mula, G., H. Garcia-Molina and A. Paepcke, 1998. Collaborative value filtering on the web. Comput. Networks, ISDN Syst., 30: 736-738.

Sanderson, M., M.L. Paramita, P. Clough and E. Kanoulas, 2010. Do user preferences and evaluation measures line up?. Proceedings of the 33rd International ACM SIGIR Conference on Research and Development in Information Retrieval, July 19-23, 2010, ACM, Geneva, Switzerland, ISBN: 978-1-4503-0153-4, pp: 555-562.

Singhal, A., 2001. Modern information retrieval: A brief overview. Bull. IEEE Comput. Soc. Tech. Committee Data Eng., 24: 35-43.

Taherizadeh, S. and N. Moghadam, 2009. Integrating web content mining into web usage mining for finding patterns and predicting users' behaviors. Int. J. Inf. Sci. Manage., 7: 51-65.

Tuteja, S., 2013. Enhancement in weighted page rank algorithm using VOL. J. Comput. Eng. (IOSR-JCE.), 14: 135-141.

Weiler, A., 2005. Information-seeking behavior in generation Y students: Motivation critical thinking and learning theory. J. Acad. Librarianship, 31: 46-53.

Yates, B. and R. Neto, 1999. Modern Information Retrieval. 1st Edn., Addison-Wesley, USA.