International Journal on Advanced Science Engineering Information Technology

Vol.12 (2022) No. 6
ISSN: 2088-5334

Evaluation of Average Term Occurrences Weighting Technique for Arabic Textual Information Retrieval

Belal Mustafa Abuata [a,*], Lama Ali Al Omari [b]

[a] *Information Technology, Yarmouk University, Irbid, 21163, Jordan*
[b] *Yarmouk University, Irbid, 21163, Jordan*
*Corresponding author: *belalabuata@yu.edu.jo*

Abstract—**Information retrieval of documents is an important process in the current time, and the vector space retrieval model uses a term weighting scheme as a basic method for matching queries with documents. Term frequency-Inverse document frequency is a widely used and famous term weighting scheme, and many studies proved its effectiveness in information retrieval. However, this term weighting scheme has some drawbacks like retrieving irrelevant documents, which sometimes reduces effectiveness. From this point, a new term weighting scheme called Term Frequency with Average Term Occurrence was proposed and experienced in the English language to minimize retrieving unnecessary documents. In this paper, an information retrieval system is built for the Arabic language, and Open-Source Arabic Corpora was used to complete experiments. Calculations were made using two schemes which are traditional Term frequency-inverse Document Frequency and proposed Term Frequency with Average Term Occurrence. After that, comparisons of results were made using evaluation measures. With all obtained queries, four case studies with two approaches (stop word removal and stemming) are implemented. In English experiments, stop word removal was applied with another discriminative approach, which calculates the centroid of documents. After the analysis of the results, it was found that the proposed scheme is applicable on Arabic text and applied approaches enhance IR effectiveness if they are both implemented. Furthermore, it was found that stop word removal has a favorable effect on both schemes which was also proved in English experiments.**

Keywords— **Term Weighting Scheme (TWS); Term Frequency-Inverse Document Frequency (TF-IDF); Okabi BM 25 model; Term Frequency-Average Term Occurrences (TF-ATO).**

Manuscript received 14 Sep. 2020; revised 28 May 2021; accepted 31 Oct. 2022. Date of publication 31 Dec. 2022.

I. Introduction

Nowadays, Internet has been developed with a large amount of information that exists. Every user needs to retrieve his/her demands of information. Information retrieval (IR) is the solution to retrieve related documents and information from the massive volume and databases of information on the Internet. IR relates to the representation, storage, organization, and access to information items [1]. It should be mentioned that text representation can be done in two ways: the first one is indexing which involves allocating index terms for documents. The second one will be allocating weights for the terms based on their significance in documents [2].

Every system that is constructed on the foundation of texts demands representation of the involved documents, which must be adequate depending on the applied function done. In more detail, it is the most dominant factor to accomplish an efficient classification of documents [2]. Text mining is much more complicated than data mining since it involves addressing the semi-structured or unstructured documents and not only the well-structured ones that are the only type of data processed by data mining. Sometimes texts are being converted into numeric vectors; as a method of representing the text. This brings the importance of mentioning Text Categorization (TC), which is essential in text mining. There are two methods for the categorization of texts: the first one is the traditional indexing of Latent Semantic Indexing (LSI), and the other one is the Term Frequency - Inverse Document Frequency (TF-IDF) [3].

With the massive growth of E-documents with messages and web pages that all have text content, the need for mining and retrieving text has been more serious and substantial. For this objective, text categorization or automatic classification is performed [4]. However, classified text documents are being demonstrated in a vector space model (VSM) by using supervised machine learning, where they are allocated to classes that have been previously predefined [2].

Term Weighting (TW) procedure is to allocate different weights to terms for achieving higher fulfillment and effectiveness for text mining and sentiment analysis [5]. TW can be defined as the difficulty of allocating terms-specific values (weights). The significance of terms in documents is determined by their allocated weights and their assistance and importance in sentiment analysis and text clustering and classification in IR. TW can achieve higher performance for many IR functions[6].

Nowadays, several weighting mechanisms have been raised based on the different attributes of text terms. Such as inverse document frequency (IDF) supposition of a term's significance in a single document related to the frequency of times that term exists in the whole other documents. IDF assumes that documents containing words that frequently occur in many documents do not have the same significance as those that occur in a less repeated manner [7]. On the other hand, residual inverse documents frequency supposes that term's significance is determined by computing the variation of the actual frequency of the term in the documents with the frequency that is predicted for it using the random Poisson distribution [8].

Text classification comprises both IR and categorization of text, and the term indexing also comprises both attributes: semantic and statistical qualities [9]. Semantic quality concerns the meaning of the term, like if the term index can characterize the components of the text and a measure of that ability. Statistical quality concerns about the index term characterized power to recognize the document group that contains that term [10]. Every IR system consists of three basic procedures: the representation of documents and user requests, then the matching between them. Those procedures are expressed in natural language [10].

Collection of documents where documents are being stored and then represented based on their contents of data and information. The indexer module creates the representation of each document, which is done by eliciting the features of documents and can be defined as terms (a single or group of keywords). Information needs of the users where it is the set of queries entered by the users to get their demands of information. Again, the process of eliciting features consistent with the documents' features is done hereafter converting queries to their information combination. Techniques of matching are a measure of similarity for how documents fulfill the demands of information in the collection.

This study proposed using a new TWS approach called Term Frequency with Average Term Occurrence (TF-ATO) with Arabic text retrieval and a discriminative approach to remove less significant weights from the documents. We conduct a study to compare the performance of TF-ATO to the widely used TF-IDF approach using various types of document collections. Also, to compare the effects of the TF-ATO on both English and Arabic systems. 2. Using various document collections, we study the impact of our discriminative approach and the stop-words removal process on both IR systems' effectiveness and performance when using the proposed TF-ATO and the well-known TF-IDF. We find that these two processes positively affect both TWSs for improving the IR performance and effectiveness. Applying an IR system requires a set of main general processes that are common in most IR systems. Fig. 1 shows the main components and processes of an IR model [11].

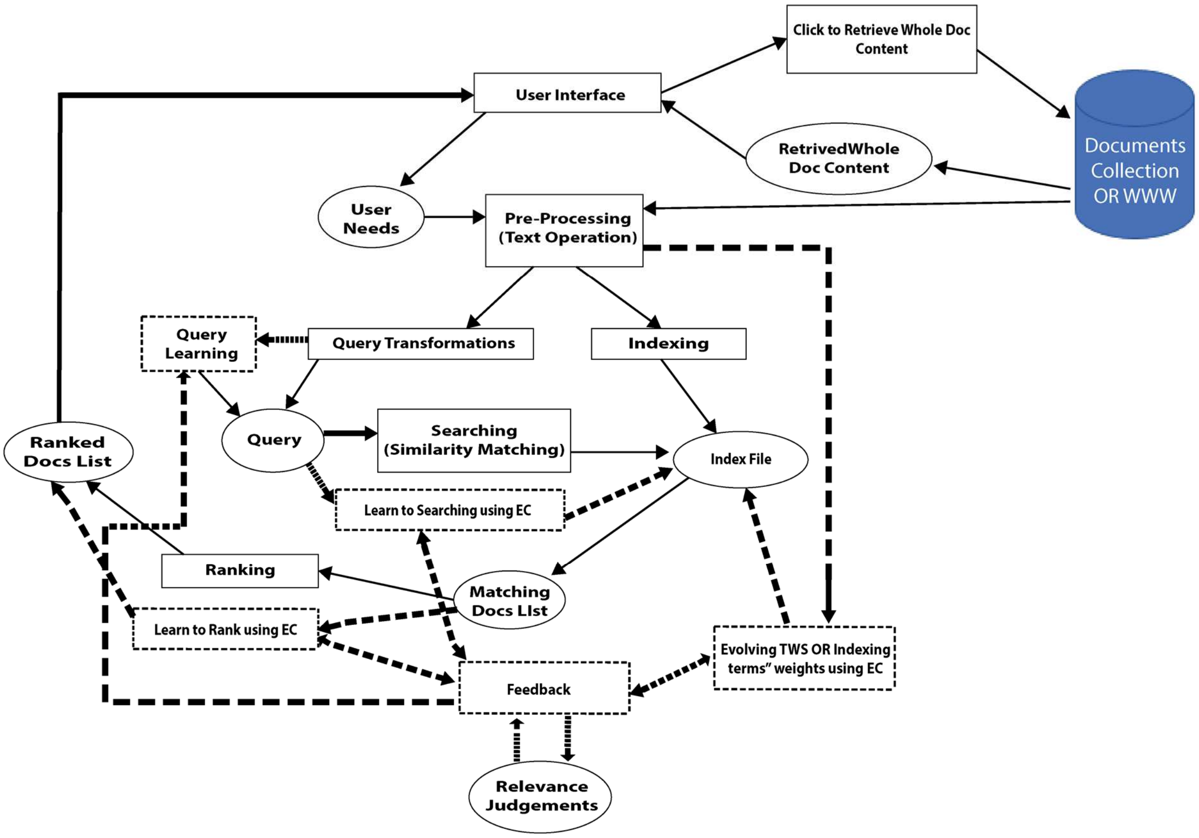
Click to Retrieve Whole Doc Content
User Interface
Documents Collection OR WWW
RetrivedWhole Doc Content
User Needs
Pre-Processing (Text Operation)
Query Learning
Query Transformations
Indexing
Ranked Docs List
Query
Searching (Similarity Matching)
Index File
Learn to Searching using EC
Ranking
Matching Docs LIst
Learn to Rank using EC
Evolving TWS OR Indexing terms" weights using EC
Feedback
Relevance Judgements

Fig. 1 IR Main Components and Processes

Those main components and processes can be explained as follows:

1) The user interface module: this module handles the interaction management between the IR and the user. In this module, the user can query for information. The query will then be preprocessed and transformed and searched for in the index file. The output of the query will be in the structure of document numbers referring to documents or links in the set of documents.

2) Preprocessing module: is responsible for applying lexical analysis, stop word removal, and stemming to the user query and the set of documents.

3) The indexing module builds the index file by implementing the Term Weighting Scheme (TWS) on the documents. The index file is mostly an inverted index, where each term has references to every document that it occurs in, with a specific weight illustrating the term significance in the document within the set of documents. Query transformation after preprocessing is applied to the query of the user to form queries of terms with their correlated weights of the terms that performed in similarity to indexing.

4) Searching module: A similarity matching is performed in the index file between the query of terms and their weights within this module. This process is implemented to create a list of document numbers that refer to documents or create links in the set of documents. The matching list ranking relies on the similarity between the user query and documents only, but now the ranking module can rely on other techniques like the host server.

5) The relevance judgment file includes the group of queries for the document set with their relevant documents from the same document set. Occasionally this file will include the document's relevancy degree of the queries like a number that specifies if the document is irrelevant or totally or even partially relevant. However, it was found that all of the test groups of documents in IR are being only partially judged and not judged because it is not feasible to do full judgments.

This study aims to show the applicability of term frequency with average term occurrence for Arabic retrieval systems. Also, we evaluate the proposed scheme and compare it with well-defined schemes such as *tf-idf* and the Okapi BM 25 to have a criterion for other techniques to compare within terms of Arabic IR effectiveness.

II. Materials and Method

TWS is mostly categorized into three groups: supervised learning, unsupervised learning, and non-learning. Different versions of TF-IDF were mainly used for these TWS and applied on different test groups for the IR. It was observed that TW composes a critical element in the IR systems and can remarkably upgrade retrieval effectiveness. Indexing includes allocating each document a group of index terms for representing its content in the set of documents. For the majority of IR systems, every single word is considered an index term [12]. TF-IDF and BM25 were widely used in IR studies, and they have shown a big success in effectiveness. These techniques employ different queries and documents to compute each term's final weight. They have also been implemented in Arabic, applications requiring weighting [13]; [12].

A function of similarity can be implemented to match user queries with vectors of the documents. In this research, cosine similarity is used for matching as it is very effective for finding angle differentiations for the vectors. Cosine similarity calculations are demonstrated in equation 1.

$$cosine\ similarity(D,Q) = \frac{\sum_{i=1}^{n} W_{id} \cdot W_{iq}}{\sqrt{\sum_{i=1}^{n} W_{id}^2} \cdot \sqrt{\sum_{i=1}^{n} W_{iq}^2}} \qquad (1)$$

This formula will find the cosine similarity for query Q and document D.
n denotes the index terms number occurring in both query and document.
Wid denotes i term weight in D.
Wiq denotes i term weight in Q.

The majority of IR systems that are considered textual use the mechanism that depends on finding the keyword from the documents to represent index terms after that allocate each one of them different weight by applying some techniques [14]. For that, drawbacks of this system would be selecting adequate keywords in an accurate way to form the index terms. Allocating accurate weights to each index term for accurately determine the index term significance for every document in the group [8]. There were many equations used for the calculations which can be explained as:

A. Basic TF-IDF

The formula of basic TF-IDF is expressed in equation 2 [15].

$$W_{ij} = tf_{ij}.\log(\frac{N}{n_i}) \qquad (2)$$

Wij the term *i* weight in document j.
tfij denotes the term *i* occurrences number in the specific document *j.*
N denotes documents number in the group of documents.
ni denotes the documents number which include the term *i* in the group of documents.

The formula in equation 3 was widely used as a weighting function for its effectiveness in IR that exceeds the other functions.

$$IDF_i = \log(N/n_i) \qquad (3)$$

The weights of terms in both the document and query vector demonstrate the term's significance in clarifying the query and document's meaning. The most effective applied factors are TF and IDF. The term weights can be estimated by the two factors product, which is TF-IDF. TF denotes the number of occurrences of a term in a document, and it is normalized using the max TF of any term inside the document to represent a TF range between 0 and 1. Terms that occur in a large number of documents will have major TF.

The primary equation of IDF for [6] was remarkably powerful. It is used in nearly all the techniques of ranking. This led to the platform Okapi BM25 which is a ranking function build since probabilistic IR model and considers the TF and length of the document. IDF is not considered a completely heuristic and does not hold a theoretic ambiguity. It is also a segment of the TF-IDF [7]. It supposes that documents containing words that frequently occur in many documents don't have the same significance as the one that contains them in a less frequent manner. TF-IDF is the most used and widespread term weighting method in IR nowadays. However, many has believed that TF-IDF is heuristic and/or empirical [16]. In this research, basic TF-IDF is applied using equation 2.

For Arabic studies, [17] searched proximity statistic by applying 2 techniques of stemming to differentiate between IR probabilistic models. They found that light stemming performs more effectively than Khoja. They stated that the Arabic IR root-based technique is referred to different words' meanings and having the same root at the same time. Meanwhile, [6] used basic TF-IDF in a unique study by making a comparison by applying VSM and cosine similarity to differentiate between root indexing and the full indexing of word.in Arabic. They found that Root indexing is more effective since the preprocessing is accomplished in less time, storage area is decreased which provides larger data retrieved that maybe relevant to the query entered by the users.

B. Augmented Maximum Term Normalization-IDF (ATC)

The formula of the Augmented maximum term normalization-IDF (ATC) by is expressed in equation 4 [18].

$$\boldsymbol{W_{ij}} = \frac{\left(\mathbf{0.5+0.5}\ .\ \frac{tf_{ij}}{\max tf_j}\right).\log(\frac{N}{n_i})}{\sqrt{\sum_{i=1}^{m}\left[\left(\mathbf{0.5+0.5}\ .\ \frac{tf_{ij}}{\max tf_j}\right).\log(\frac{N}{n_i})\right]^2}} \tag{4}$$

m denotes the terms number within the space of documents. *Max tfj* denotes the maximum value of document j term frequency.

English, Arabic and other languages of IR used many normalization techniques including equation 4 to obtain better results of retrieval [18]; [5]. But also, a drawback of this model was noticed that not all documents retrieved are exactly relevant.

Zubi *et al.* [19] implemented Normalized TF-IDF to get rid of document lengths various issues. He presented Arabic Text Classifier (ATC), which compares the outcomes of (CK-NN, CNB) and then chooses the best outcomes rates for average accuracy. k-fold cross-validation was applied for accuracy evaluations.

In another study, Habib [20] used Normalized TF-IDF, (document-pivoted categorization – DPC) and (category-pivoted categorization – CPC). The thesis introduced the Arabic system for categorization. Outcomes of experiments reveal the factors that affect categorization performance like feature selection, preprocessing, representation of documents, plus the techniques of categorization.

C. Okapi TWS and Okapi BM 25

The drawback of this TWS is expressed by producing a negative value for the term weight in the case an index exists in more than half of the group of documents. This prevents the term information from being adequately presented. The formula for the okapi TWS is defined in equation 5 [17].

$$\boldsymbol{W_{ij}} = \left(\frac{tf_{ij}}{\mathbf{0.5+1.5}\ .\frac{dl_j}{avg_{dl}}+tf_{ij}}\right).\log\left(\frac{N-n_j+0.5}{tf_{ij}+0.5}\right) \tag{5}$$

dlj denotes document j length like the amount of adding all the document j term frequencies.

avgdl denotes the document length average for the group of documents.

Also, the okapi basic formula is probabilistic that relies with its constants on the documents that are relevant to the query. For that reason, most of the group documents will not be relevant for the queries in the group of queries and this is because test and real groups have partial judgments.

Research and investigation in the IR field of Arabic language has been a new sector which demands a lot of work and inspection. Different IR models which were new at the time in the period between 1970 and 1980 needed experimentation to inspect their effectiveness. At first, only small data sets (only few thousand articles) were available which creates an issue of uncertainty regarding their effectiveness with large data sets. Then Text Retrieval Conference (TREC) was constructed to make a revolution of modifications and developing of new and previously existing methods to accomplish better effectiveness. "TREC is a series of evaluation conferences sponsored by various US Government agencies under the auspices of NIST, which aims at encouraging research in IR from large text collections." (Singhal, 2001). "The basic design of the TREC experiments (TRECs 1 and 2, specifically) has been oriented towards batch search operations, and it is a little difficult to incorporate any fundamentally interactive techniques into experiments conducted under TREC rules." [21].

The Okapi Best Match 25 BM25 is one of the best conventional retrieval models and has maintained its modern status in information retrieval (IR) for nearly 20 years since its commencement. From the perspective of information it uses, BM25 is similar to the even better-known cosine *tf.idf* model. Both models rely on document frequencies of the terms of a query, as collected across the indexed document collection, and their frequencies within each candidate document [22].

The Okapi BM25 model is a non-twofold model created as a major aspect of the Okapi Basic Search System in the TREC Conferences. Okapi BM25 is a probabilistic model that depends on the probabilistic theory. The model is a good term weighting scheme that regains its relevant results by integrating weight term scheme TF-IDF, and length normalization of a given document [23]. BM25 is a bag-of-words retrieval function that ranks documents according to their relevant results. Okapi BM25 not just considers the recurrence of the inquiry terms yet additionally the entire length of the document under assessment

The Okapi BM25 model calculates the retrieval status value of a given document to determine the relevance as shown in equation 6 [24].

$$RSV_d = \sum_{t\in q}\log\ \frac{N}{dft} * \frac{(k_1+1)\ tf_{td}}{k_1\left((1-b)+bx\left({}^{L_d}/_{L_{ave}}\right)\right)+tf_{td}} \tag{6}$$

Where:

Retrieval Status Value ($\mathbf{RSV_d}$) : relevancy scores of a document.

N : represents documents in a given collection.

dft : the frequency of a query term in a document.

$\boldsymbol{t \in q}$: t is an element of query q.

t : term
q : query
tf td : signifies the frequency of a term in document d
***Ld* (Lave)** : used to calculate the average document length in the whole collection
k1 : tuning parameter set to 1.2
b : tuning parameter set to 0.75

D. Pivoted Document Length Normalization-IDF (LTU)

This TWS was applied by English, Arabic IR systems besides other languages [12]. It was found that this scheme is not efficient with short documents, but it has been useful for Optical Character Recognition and also for the documents that tend to be longer. Another drawback would be the changing nature of documents from static to dynamic in the current time, where LTU depends on the total documents number in the group plus the term frequency of the document. The formula for Pivoted document length normalization-IDF (LTU) is shown in equation 7.

$$\boldsymbol{W_{ij}} = \left(\frac{\mathbf{1+\log(}tf_{ij}\mathbf{)}}{\mathbf{0.8+0.2}\ .\frac{dl_j}{avg_{dl}}} \right) \mathbf{.\log}\left(\frac{N}{n_i}\right) \qquad (7)$$

However, the previous TF-IDF versions and TWS did not provide any preference contrasting to the TF-IDF for test group presentation of information when applying cosine similarity for dynamic records [17]. It is worth mentioning that there are a large number of TF-IDF versions, which makes choosing a formula adequate for these versions a serious process requiring skill in studying a new set of data. Unfortunately, there were not many studies about those schemes, including the Arabic language, so that this thesis would create an additional benefit regarding this field. Hopefully, after our experiments are finished, the results will be used as a threshold. The model in this paper is built on the foundation of VSM and includes the procedures as in Figure 2.

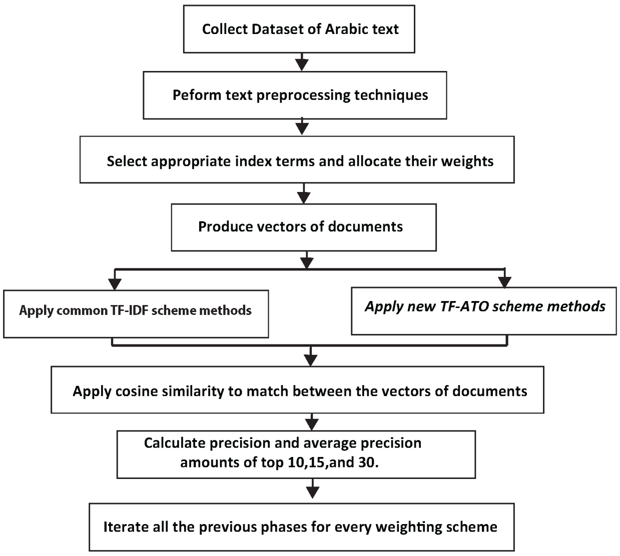
Collect Dataset of Arabic text
Peform text preprocessing techniques
Select appropriate index terms and allocate their weights
Produce vectors of documents
Apply common TF-IDF scheme methods
Apply new TF-ATO scheme methods
Apply cosine similarity to match between the vectors of documents
Calculate precision and average precision amounts of top 10,15,and 30.
Iterate all the previous phases for every weighting scheme

Fig. 2 The Overall Design of the Research Phases

Figure (2) explains the overall design of the research phases. In this paper, VSM will be applied due to its effective and broad applications in research. Also, this was supported by many studies for discriminative and normalization techniques, studies, and searches. Therefore, VSM is seen to be the best to implement in our research [25]. The model in this paper is built on the foundation of VSM and includes the procedures shown in Fig. 2, which explains the overall design phases of the research.

E. Proposed TF-ATO Scheme

Automatic TWS encounter a serious question of how to allocate terms with proper weights. In order to solve all the problems that exist in the evolving methods and TF-IDF, TF-ATO was used by Ibrahim and Landa-Silva [11] on English text IR and proved its effectiveness and ability to raise IR performance. They also used a discriminative approach that calculates vectors centroid besides the stop word removal approach to eliminate stop words. In this research, the stop word removal approach is used besides another approach which is stemming. Also, term weights are calculated by the equations explained below.

The following equations 8 and 9 explain it [11].

$$W_{ij} = \frac{tf_{ij}}{\#\, ATO\ in\ document\ j} \qquad (8)$$

and

$$\#\, ATO\ in\ document\ j = \frac{\sum_{i=1}^{m_j} tf_{ij}}{m_i} \qquad (9)$$

tf_{ij} denotes the *i*-term frequency in j document
ATO denotes term occurrences in documents average where it is calculated in every document.
m_j denotes document j unique terms number also can be denoted by document j index terms number.

TF-IDF scheme and its different versions, term weight relies on attributes of the document group in a global portion. So TF-ATO treats global weights as they are the same for any of the term weights that holds a value equals to one, and this is for any term that occurs within the group [11]. ATO is calculated in every document. The Main purpose of the proposed new scheme is to eliminate less significant terms by using proposed equations and thus overcome traditional TF-IDF weaknesses.

F. Arabic corpus

Automatic text analysis is now available by the corpus-based techniques that produced novel features for different applications and linguistics. Open-Source Arabic Corpora (OSAC) by [26] was extremely important to be built because of the persistent demand for a corpus in the Arabic language and also the lack of availability of Arabic corpus in the last years. This corpus was constructed to serve many purposes, including linguistics. Also, there was another challenge which is the unavailability of open-source Arabic corpus. Many studies stated that available corpora are very bounded in the genres and types. Due to the Arabic language lacking corpora, it is difficult to display textual content and quantitative data of Arabic.

The documents used in this research are 316 documents taken from the Open-Source Arabic **CNN** corpora**.** This corpus comprises 5,070 text documents gathered from bbcarabic.com. Each one of the text documents belongs to one of the following six categories:

- Middle East News: 1462.
- Entertainments: 474.
- Business: 836.
- Science & Technology: 526.
- Sports: 762
- World News: 1010.

The corpus contains 2,241,348 (2.2M) words with 144,460 distinct keywords after stop words removal.

G. Perform Preprocessing Techniques

Each document should be preprocessed using various techniques such as stemming, tokenization, and stop word removal. This is done for search engines to facilitate using them as documents and applying their algorithms [27]. Within this phase, three processes are going to be performed, which are:

- The first process is applying tokenization and lexical analysis on the text to handle letters, punctuation, and digits.
- The second step is to stop words removal to filter out words with a minimum amount of discrimination for retrieval and matching.
- The third process is stemming for the rest of the words for eliminating affixes (suffixes plus prefixes), then permit the document retrieval that includes syntactic queries terms variances.

H. Select Appropriate Index Terms and Allocate Their weights

This involves choosing the index terms based on what stems or words are to be implemented as the index terms—then allocating index terms weights for every document by employing a specified weighting scheme that will determine the significance of that index term in a particular document.

I. Produce Vectors of Documents

This involves producing vectors of the documents of term weights within the group of documents (generate directed with inverted records by assigning for documents their term weights from the group of documents).

- Apply common TF-IDF Scheme Methods. This phase includes calculating the weights of terms using equation 2.
- Apply common Okabi BM 25 Scheme Methods. This phase includes calculating the weights of terms using equation 5.
- Apply New TF-ATO Scheme Methods. This phase includes calculating the weights of terms using equations 8 and 9.

J. Apply Cosine Similarity to Match between the Vectors of Documents

This phase includes applying cosine similarity using equation 1 to match the vectors of documents with every query to retrieve correlated documents.

K. Apply Evaluation Metrics

There are many measures used for evaluating IR systems. The most used and popular are recall, precision, and F-measure. In this phase, the proposed IR system and scheme are evaluated using precision, recall, F1 measure, average precision, and Mean Average Precision (MAP). The evaluation is proportional, and subjects are the recall and precision. Precision is subject to the retrieved documents, while recall depends on relevant documents in the group of documents. According to that, the great values of precision indicate just the retrieved files, but in contrast, the recall locates just all files retrieved, considering them relevant. Mathematically precision can be demonstrated as the ratio of the relevant records retrieved to the total number of records retrieved (irrelevant and relevant). It is usually expressed as a percentage, as shown in equation 10.

$$Precision = \frac{RelRetrieved}{Retrieved} \tag{10}$$

Recall denotes the relevant documents retrieved contrasted to all accessible files. Recall can be presented by (number of relevant documents retrieved/number of relevant documents in the document group). Also, recall can be demonstrated as the ratio of the number of relevant records retrieved to the total number of relevant records in the database. It is usually expressed as a percentage. Precision and recall are both used to measure the relevance in the IR system, and they are inversely related. The recall equation is shown in equation 11.

$$Recall = \frac{RelRetrieved}{Rel\ in\ collection} \tag{11}$$

For measuring effectiveness using a single measure, the F1 measure is used. F-measure is the harmonic mean of both recall and precision; it is calculated in our IR system since it is essential for establishing a trade-off between precision and recall. The standard F- measure is F1 that provides equivalent significance to recall and precision. F1 measure calculation is explained in equation 12.

$$F_1 = 2.\frac{precision\ .recall}{precision + recall} \tag{12}$$

where an F_1 score reaches its best value at 1 (perfect precision and recall) and worst at 0.

For this research, evaluations for both system performance and effectiveness will be used to assess the effects of stop words elimination. Also, the index files rate for each single case study is used here as a performance measurement. For effectiveness, Average Precision (AvgP) and Mean Average Precision (MAP) are also implemented. The working mechanism is based on the study of Ibrahim [11]. It is required to consider the documents stored order descending by their similarity values of the measure function as d_1, d_2, d_3,……. $d_{|D|}$.

|D| = testing document number

$r(d_i)$ = this provides d_i relevance value.

This works by presenting 1 if the document d_i is relevant and 0 for non-relevant. It is shown in equation 13 as follows:

$$\boldsymbol{Avg\ P(q) = \frac{1}{|D|}\sum_{i=1}^{|D|}(r(d_i)).\sum_{j=1}^{|D|}\frac{1}{j}} \tag{13}$$

L. IR System

There is no available open-source Arabic IR system that can do the complete preprocessing of data with TF-IDF calculations, Okabi BM 25, and cosine similarity. We built an Arabic IR system that was constructed using JAVA, IDE net beans. The system applied Khoja stemmer for Shireen and accomplished all the data preprocessing phases.

Specifications of the computer are CPU intel i5, 8GB Ram, Windows 10 pro, NetBeans 8.1, and Java 8. Fig. 3 shows the flow chart of constructed IR system used in this research.

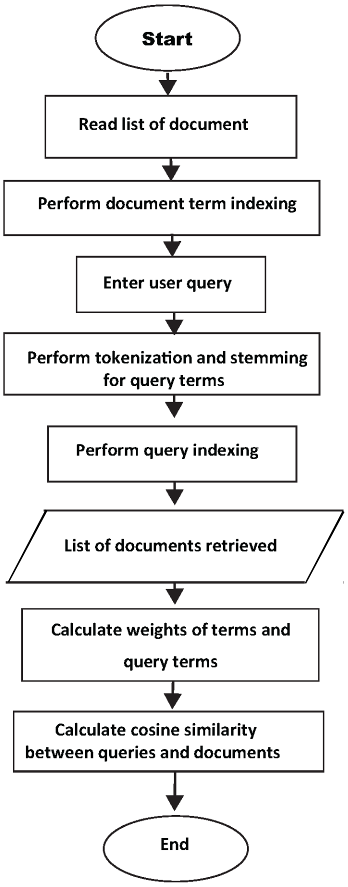
Start
Read list of document
Perform document term indexing
Enter user query
Perform tokenization and stemming for query terms
Perform query indexing
List of documents retrieved
Calculate weights of terms and query terms
Calculate cosine similarity between queries and documents
End

Fig. 3 Flow chart of the IR Constructed System.

The IR system reads 316 documents of our dataset after files have been filtered manually to remove mistakes or redundancy for the 2 topics of Science and Technology (علوم وتكنولوجيا) and Mixture (منوعات). Data preparation was done where all collected corpora were converted to utf-8 encoding, html tags were removed. The corpora are available publicly. After reading files of documents, the system performs indexing of those files after they have been preprocessed plus indexing query terms. Relevant documents are retrieved depending on the query entered. Then the system calculates TF of each term with the file name where it exists and that is after the query is entered. It also calculates IDF, TF-IDF, BM 25, TF-ATO, and cosine similarity between each query entered with the documents that contain query terms. After performing the calculations and acquiring the results, we noticed that Khoja stemmer has a problem and is not precise in calculating word stems. Finally, calculations were completed manually to calculate the evaluations metrics mentioned in chapter 3 using equations 9, 10, 11, 12, and 13.

M. Case Studies and Calculations:

We worked on 4 case studies to examine the performance of the used techniques and methods. Table 1 will explain all the cases with the techniques applied along with each case.

TABLE I
CASE STUDIES USED.

Case study	**Techniques applied**	
	Stop word removal	**Stemming**
Case 1	X	√
Case 2	X	X
Case 3	√	√
Case 4	√	X

III. RESULT AND DISCUSSION

We evaluated our results using the evaluation metrics and their calculations mentioned in the methodology section. Precision, recall, and F-measure was calculated. The following tables present the results. Tables 2 and Table 3 provide examples for Average Recall-Precision of the query "Security forces" (قوات الأمن) for case #1 for top-30 documents retrieved. Table 2 shows the TF-IDF results, Table 3 for the TF-ATO, and Table 4 for the BM 25.

TABLE II
AVERAGE RECALL-PRECISION OF TF-IDF FOR TOP-30 OF THE QUERY "SECURITY FORCES" (قوات الأمن) FOR CASE #1

Recall	**Interpolated AvgP of TF-IDF for the top 30 documents**
0.0	1
0.1	1
0.2	1
0.3	1
0.4	0.5
0.5	0.5
0.6	0.5
0.7	0.17
0.8	0.17
0.9	0.17
1.0	0.17

TABLE III
AVERAGE RECALL-PRECISION OF TF-ATO FOR TOP-30 OF THE QUERY "SECURITY FORCES" (قوات الأمن) FOR CASE #1

Recall	**Interpolated AvgP of TF-ATO for the top 30 documents**
0.0	1
0.1	1
0.2	1
0.3	1
0.4	0.5
0.5	0.5
0.6	0.5
0.7	0.6
0.8	0.6
0.9	0.6
1.0	0.6

TABLE IV
AVERAGE RECALL-PRECISION OF BM 25 FOR TOP-30 OF THE QUERY "SECURITY FORCES" (قوات الأمن) FOR CASE #1

Recall	**Interpolated AvgP of BM 25 for the top 30 documents**
0.0	1
0.1	1
0.2	1
0.3	1
0.4	1
0.5	1
0.6	0.6
0.7	0.6
0.8	0.6
0.9	0.5
1.0	0.5

After finding the recall and precision values for all 4 cases and all TWS, 11 interpolated recall values were calculated with the AvgP for the 24 queries set. Table 5 displays 11 values of recall where each value is accompanied by the average precision value obtained by TF-IDF, BM 25, and TF-ATO for the top 10 documents retrieved.

TABLE V
INTERPOLATED AVERAGE RECALL-PRECISION OBTAINED OF ALL QUERIES FOR THE 4 CASE STUDIES

Recall	Interpolated Average precision for case studies											
	Case 1			Case 2			Case 3			Case 4		
	TF-IDF	BM 25	TF-ATO	TF-IDF	BM 25	TF-ATO	TF-IDF	BM 25	TF-ATO	TF-IDF	BM 25	TF-ATO
0.0	0.81	0.87	0.76	1	1	0.96	0.52	0.51	0.61	0.96	0.92	0.96
0.1	0.81	0.87	0.76	1	1	0.96	0.52	0.50	0.61	0.96	0.92	0.96
0.2	0.78	0.80	0.73	0.99	1.0	0.94	0.42	0.40	0.51	0.95	0.92	0.95
0.3	0.79	0.80	0.73	0.99	1.0	0.95	0.40	0.38	0.43	0.94	0.92	0.94
0.4	0.76	0.77	0.70	0.95	0.96	0.91	0.38	0.38	0.43	0.92	0.90	0.92
0.5	0.77	0.77	0.73	0.95	0.95	0.91	0.24	0.24	0.34	0.91	0.89	0.91
0.6	0.72	0.71	0.69	0.90	0.91	0.86	0.24	0.24	0.37	0.89	0.89	0.89
0.7	0.70	0.71	0.68	0.84	0.91	0.80	0.19	0.19	0.29	0.81	0.78	0.81
0.8	0.68	0.70	0.65	0.80	0.78	0.76	0.19	0.17	0.29	0.79	0.78	0.79
0.9	0.65	0.70	0.63	0.77	0.75	0.73	0.19	0.17	0.29	0.78	0.76	0.78
1.0	0.65	0.70	0.63	0.77	0.75	0.73	0.19	0.17	0.29	0.75	0.72	0.75

MAP was also calculated for all 4 cases and all TWS (TF-IDF, BM 25, and TF-ATO). The results are shown in Table 6.

TABLE VI
MAP OBTAINED FROM EACH CASE STUDY OF THE EXPERIMENTS

Case number	TWS	MAP
Case #1	TF-IDF	0.73
	BM 25	0.77
	TF-ATO	0.69
Case#2	TF-IDF	0.90
	BM 25	0.92
	TF-ATO	0.86
Case#3	TF-IDF	0.31
	BM 25	0.32
	TF-ATO	0.40
Case#4	TF-IDF	0.87
	BM 25	0.83
	TF-ATO	0.73

MAP encapsulates all queries ratings by calculating the average of AvgP. MAP is used largely for research papers. Also it is based on the supposition that for every query in the query set, the user concerns about locating many relevant documents. F1 measure is known in IR to be used for combining both measures of precision and recall to result in measuring the effectiveness of the IR system.

TABLE VII
F-MEASURE OBTAINED FROM EACH CASE STUDY FOR THE EXPERIMENTS

Case number	TWS Used	F1- measure
Case #1	TF-IDF	0.51
	BM 25	0.53
	TF-ATO	0.50
Case#2	TF-IDF	0.65
	BM 25	0.66
	TF-ATO	0.55
Case#3	TF-IDF	0.28
	BM 25	0.28
	TF-ATO	0.34
Case#4	TF-IDF	0.86
	BM 25	0.83
	TF-ATO	0.86

Table 7 displays 11 values of recall where each value is accompanied with the F1- measure calculated from the average precision values obtained by TF-IDF, BM 25 and TF-ATO for the top 30 documents retrieved.

N. Results in Arabic

For the Arabic language, the results showed that BM 25 provided either the same Interpolated AvgP results or better Interpolated AvgP results than TF-IDF and TF-ATO in cases 1, 2 (table5) at low recall values. This is due to the retrieval of long documents by BM 25. As for TF-IDF, it provided better results than TF-ATO in cases 1, 2, and 4 (refer to table 5) because of its capability of eliminating some of the words that are non-significant in the documents, which is done by allocating those repeated words in all the document collection an amount of zero.

In case 4, results were the same for TF-IDF and TF-ATO, and both are better than BM 25. This is because no stemming was used. In case 1, however, when no stop word removal was used, TF-IDF provided higher results. However, TF-ATO provided better results than TF-IDF values in case 3, where results were more precise. Both methods (stop word removal and stemming) have a positive impact on the TF-ATO.

Case 2 showed that it is best to apply to get the best results for TF-IDF where neither of the 2 approaches were used. In case 3, TF-ATO outperformed TF-IDF when both stop word removal and stemming were used. Also, many queries did not retrieve any relevant document which is: Arabian Summit (قمة العربية), Fine Artist (فنان جميل), Residents of Riyadh (سكان الرياض), Film Activity (النشاط السينمائي) and public sector efforts (جهود القطاع العام).

Whereas, for queries that have only one relevant document, in some cases, TWS retrieved the one relevant document at first, which resulted in a precision value of 1 for all 11 interpolated values. As we can see from Table 7 the F1 measure value was higher with BM 25 for case 2 (apply TWS without stop word removal and without stemming) while they were approximately the same for case 1 and exactly the same for case 4. The F1 measure for TF-ATO was higher for case 3 (apply TWS with stop word removal and stemming).

Finally, the highest score of the harmonic mean was obtained in case 4, which was 0.86.

After all, we can notice that TF-ATO can improve the effectiveness of the IR system. Also, some problems with the Khoja stemmer were noticed. One issue was that the stemmer sometimes produced stems for stop words and included the same stem for stop words with other words, which may affect the results.

O. A comparative study with results in English

Comparing to the results obtained with English language and dataset, it was noticed according to Ibrahim [11] that stop word removal has a favorable impact for enhancing the effectiveness and the performance of the IR on both TF-IDF and TF-ATO. Similarly, using Arabic, TF-ATO results were better, and effectiveness has enhanced. But for TF-IDF, stop word removal enhanced its results only when no stemming was used. Stemming was used instead of their discriminative approach with stop word removal. Their study revealed that only when both stop word removal and discriminative approach are not used, TF-IDF outperforms TF-ATO. In our results, only when both stop word removal and stemming are used; TF-ATO outperformed TF-IDF.

IV. Conclusion

The main goal of this research is to explore the effects of the new proposed TF-ATO on the Arabic language by applying it besides to (TF-IDF) TWS on the same corpus using 4 cases with the IR system built. The effects of stop word removal and stemming were tested. The proposed novel TWS called TF-ATO was experimented on the Arabic open-source corpora OSAC using the two topics: Science and Technology (علوم وتكنولوجيا) and mix (منوعات). An IR system was built using JAVA to perform needed processes for Arabic language and experiments. After preprocessing was made to the corpus, TWS (TF-IDF, BM 25, and TF-ATO) were used to complete investigations with the 4 cases set. All cases were applied to check the influence of stemming and stop word removal on the 2 TWS.

After experiments were completed, it was found out that the new proposed TWS (TF-ATO) can be remarkably effective if used in some cases comparing with TF-IDF and BM 25. It is well known that TF-IDF can eliminate some of the keywords that are non-significant while TF-ATO doe does not have that ability. The results also showed that BM 25 has given greater results than TF-IDF and TF-ATO in cases 1 and 2 (longer documents indexed and retrieved), and gave lower results in cases 3 and 4 (lower documents indexed and retrieved). Applying both approaches of stop word removal and stemming with TF-ATO in case 3 has an effective influence for taking off the non-significant keywords where results for TF-ATO were greater than TF-IDF and BM 25. So it is very useful to apply the proposed TF-ATO with stop word removal and stemming from raising the effectiveness and performance of IR.

It was shown that only when both stop-words removal and stemming are applied together; TF-ATO outperforms TF-IDF and BM 25. It was noticed after studying and experiments that the TF-ATO has an impact on Arabic language, but it was lower than the impact that was made on English language according to the results obtained. This research as a reference for later related Arabic studies since it is the first study that used the Arabic language with the new term weighting scheme (TF-ATO). No other related studies were made using TF-ATO on the Arabic language; only studies on the English language were made using it.

As future work, another stemmer can be applied for obtaining better results if possible. A larger collection of documents of the same corpus and topics can be applied. Also, it is intended to apply the proposed discriminative approach that was exercised on English with TF-ATO to compare the results together.

References

[1] R. Baeza-Yates and B. Ribeiro-Neto, *Modern Information Retrieal*, vol. 9. ACM Press NewYourk, 1999.

[2] E. Amigó, F. Giner, J. Gonzalo, and F. Verdejo, "On the foundations of similarity in information access," *Inf. Retr. J.*, vol. 23, no. 3, pp. 216–254, 2020, doi: 10.1007/s10791-020-09375-z.

[3] D. Harman, "Information Retrieval: The Early Years," *Found. Trends® Inf. Retr.*, vol. 13, no. 5, pp. 425–577, 2019.

[4] G. Domeniconi, G. Moro, R. Pasolini, and C. Sartori, "A study on term weighting for text categorization: A novel supervised variant of tf.idf," *DATA 2015 - 4th Int. Conf. Data Manag. Technol. Appl. Proc.*, pp. 26–37, 2015, doi: 10.5220/0005511900260037.

[5] Z. H. Deng, K. H. Luo, and H. L. Yu, "A study of supervised term weighting scheme for sentiment analysis," *Expert Syst. Appl.*, vol. 41, no. 7, pp. 3506–3513, 2014, doi: 10.1016/j.eswa.2013.10.056.

[6] D. Jones *et al.*, "Improving engineering information retrieval by combining TD-IDF and product structure classification," *Proc. Int. Conf. Eng. Des. ICED*, vol. 6, no. DS87-6, pp. 41–50, 2017.

[7] S. Robertson, "Understanding inverse document frequency: On theoretical arguments for IDF," *J. Doc.*, vol. 60, no. 5, pp. 503–520, 2004, doi: 10.1108/00220410410560582.

[8] I. A. & F. A. Belal Abuata, "Improving arabic question answering system by merging aner technique, updated question classification technique and stop words technique," *J. Theor. Appl. Inf. Technol.*, vol. 98, no. 23, pp. 24–38, 2020.

[9] K. Chen, Z. Zhang, J. Long, and H. Zhang, "Turning from TF-IDF to TF-IGM for term weighting in text classification," *Expert Syst. Appl.*, vol. 66, pp. 1339–1351, 2016, doi: 10.1016/j.eswa.2016.09.009.

[10] A. El Mahdaouy, S. O. El Alaoui, and E. Gaussier, "Semantically enhanced term frequency based on word embeddings for Arabic information retrieval," *Colloq. Inf. Sci. Technol. Cist*, vol. 0, pp. 385–389, 2016, doi: 10.1109/CIST.2016.7805076.

[11] O. A. S. Ibrahim and D. Landa-Silva, "Term frequency with average term occurrences for textual information retrieval," *Soft Comput.*, vol. 20, no. 8, pp. 3045–3061, 2016, doi: 10.1007/s00500-015-1935-7.

[12] R. Bentrcia, S. Zidat, and F. Marir, "Extracting semantic relations from the Quranic Arabic based on Arabic conjunctive patterns," *J. King Saud Univ. - Comput. Inf. Sci.*, vol. 30, no. 3, pp. 382–390, 2018, doi: 10.1016/j.jksuci.2017.09.004.

[13] B. Abuata and A. Al-Omari, "A rule-based stemmer for Arabic Gulf dialect," *J. King Saud Univ. - Comput. Inf. Sci.*, vol. 27, no. 2, pp. 104–112, 2015, doi: 10.1016/j.jksuci.2014.04.003.

[14] A. El Mahdaouy, É. Gaussier, and S. O. El Alaoui, "Exploring term proximity statistic for Arabic information retrieval," *Colloq. Inf. Sci. Technol. Cist*, vol. 2015-Janua, no. January, pp. 272–277, 2015, doi: 10.1109/CIST.2014.7016631.

[15] A. A. A. A. Abdulla, H. Lin, B. Xu, and S. K. Banbhrani, "Improving biomedical information retrieval by linear combinations of different query expansion techniques," *BMC Bioinformatics*, vol. 17, no. 2, 2016, doi: 10.1186/s12859-016-1092-8.

[16] A. Aizawa, "An information-theoretic perspective of tf-idf measures," *Inf. Process. Manag.*, vol. 39, no. 1, pp. 45–65, 2003, doi: 10.1016/S0306-4573(02)00021-3.

[17] R. Jin, C. Falusos, and A. G. Hauptmann, "Meta-scoring: Automatically evaluating term weighting schemes in IR without precision-recall," *SIGIR Forum (ACM Spec. Interes. Gr. Inf. Retrieval)*, pp. 83–89, 2001.

[18] G. Salton and C. Buckley, "Term-weighting approaches in automatic text retrieval," *Inf. Process. Manag.*, vol. 24, no. 5, pp. 513–523, 1998.

[19] Z. S. Zubi, "Using some web content mining techniques for Arabic text classification," *Proc. 8th WSEAS Int. Conf. Data Networks, Commun. Comput. DNCOCO '09*, pp. 73–84, 2009.

[20] M. Habib, "An intelligent system for automated arabic text categorization," 2008.

[21] S. E. Robertson, S. Walker, and M. M. Hancock-Beaulieu, "Large test collection experiments on an operational, interactive system: Okapi at TREC," *Inf. Process. Manag.*, vol. 31, no. 3, pp. 345–360, 1995, doi: 10.1016/0306-4573(94)00051-4.

[22] S. Jimenez, S. P. Cucerzan, F. A. Gonzalez, A. Gelbukh, and G. Dueñas, "BM25-CTF: Improving TF and IDF factors in BM25 by using collection term frequencies," *J. Intell. Fuzzy Syst.*, vol. 34, no. 5, pp. 2887–2899, 2018, doi: 10.3233/JIFS-169475.

[23] G. Pandey, Z. Ren, S. Wang, J. Veijalainen, and M. de Rijke, "Linear feature extraction for ranking," *Inf. Retr. J.*, vol. 21, no. 6, pp. 481–506, 2018, doi: 10.1007/s10791-018-9330-5.

[24] G. A. Tinega, P. W. Mwangi, and D. R. Rimiru, "Text Mining in Digital Libraries using OKAPI BM25 Model," *Int. J. Comput. Appl. Technol. Res.*, vol. 7, no. 10, pp. 398–406, 2018, doi: 10.7753/ijcatr0710.1003.

[25] A. Lipani, T. Roelleke, M. Lupu, and A. Hanbury, *A systematic approach to normalization in probabilistic models*, vol. 21, no. 6. Springer Netherlands, 2018.

[26] M. Saad and W. Ashour, "OSAC: Open Source Arabic Corpora," *6th Int. Conf. Electr. Comput. Syst. (EECS'10), Nov 25-26, 2010, Lefke, Cyprus.*, pp. 118–123, 2010.

[27] Nicola Ferro, "Reproducibility Challenges in Information Retrieval Evaluation," *J. Data Inf. Qual.*, vol. 8, no. 2, pp. 1–4, 2017.