A New Weighting Scheme and Discriminative Approach for Information Retrieval in Static and Dynamic Document Collections

Osman A. S. Ibrahim
School of Computer Science, ASAP Research Group
The University of Nottingham, UK
Computer Science Dept., Minia University, Egypt
psxoi@nottingham.ac.uk

Dario Landa-Silva
School of Computer Science, ASAP Research Group
The University of Nottingham, UK
dario.landasilva@nottingham.ac.uk

Abstract*—This paper introduces a new weighting scheme in information retrieval. It also proposes using the document centroid as a threshold for normalizing documents in a document collection. Document centroid normalization helps to achieve more effective information retrieval as it enables good discrimination between documents. In the context of a machine learning application, namely unsupervised document indexing and retrieval, we compared the effectiveness of the proposed weighting scheme to the '*Term Frequency – Inverse Document Frequency*' or TF-IDF, which is commonly used and considered as one of the best existing weighting schemes. The paper shows how the document centroid is used to remove less significant weights from documents and how this helps to achieve better retrieval effectiveness. Most of the existing weighting schemes in information retrieval research assume that the whole document collection is static. The results presented in this paper show that the proposed weighting scheme can produce higher retrieval effectiveness compared with the TF-IDF weighting scheme, in both static and dynamic document collections. The results also show the variation in information retrieval effectiveness that is achieved for static and dynamic document collections by using a specific weighting scheme. This type of comparison has not been presented in the literature before.

Keywords— term weight; term discrimination; document centroid; TF-IDF; static document collection; dynamic document collection.

I. INTRODUCTION

The time and cost involved in information retrieval (IR) are considerable. More than 90% of the data in the World Wide Web has been generated from 2011 to date [1]. Companies bear the costs for using these data over the web. Feldman [2, page 3] [3] found that each employee spends an average of 13 hours per week checking emails. Also, workers spend increasing amounts of time on social networks and instant messaging systems using them as communication channels. According to Feldman's study, the time spent searching for information averages 8.8 hours per week costing $14,209 per employee per year. Then, analyzing the information soaked up an additional 8.1 hours, costing the organization $13,078 annually per employee. According to the study, each employee spends over a third of his time searching for the information needed and another quarter of his time analyzing it to obtain relevant information. It is then very important that the use of that time is as productive as possible.

The size of document collections increases over time; hence the datasets (collections of documents) in information retrieval systems can be considered as dynamic data streams or in other words, dynamic document collections. However, most of the existing statistical weighting schemes in information retrieval research assume that the whole dataset is a static data stream [4]. Thus, for every major update of the dataset in information retrieval systems, all the terms' weights should be re-computed for all documents in order to maintain the same retrieval effectiveness gained with the weighting scheme in use. In information retrieval, the weight of a given term in a given document represents the importance or the information content of that term to the document.

In this work, we propose a new weighting scheme that can be used in both static and dynamic document collections. To assess the effectiveness of the proposed approach, we compared our weighting scheme to the most commonly used weighting scheme these days, *term frequency – inverse document frequency* (TF-IDF). This scheme has been considered as a prominent weighting scheme, specifically from a statistically unsupervised point of view, with its many possible variations that take additional parameters into consideration. In the comparison carried out here, we illustrate the variation in effectiveness between TF-IDF and our proposed weighting scheme when applied to static and dynamic document collections. Also, we used the document centroid vector as a discriminative approach in this work. The document centroid vector is the average documents vector for all documents in the document collection.

The intended contributions of this paper are summarized as follows:

- We propose a new *term-weighting scheme* aimed at increasing the effectiveness of the information retrieval system. The proposed scheme seeks to use appropriate weighting that corresponds to the real information content of each term in each document in the document collection.
- The effectiveness of the proposed *term weighting scheme* is comparable to the well-know and mostly used TF-IDF weighting scheme.

978-1-4799-5538-1/14/$31.00 ©2014

- We propose a new *discriminative approach* aimed at removing the less significant weights in each document in order to discriminate better between documents in the collection. This proposed discriminative approach employs the document centroid as a threshold for removing less significant terms weights.

- We illustrate the variation in effectiveness that results from using a particular weighting scheme when applied to static and dynamic document collections. Document collections in real IR systems are often updated by removing or adding documents to their IR datasets. This variation in effectiveness shows the possible drawbacks of using dynamic document collection in today's IR systems. This is because IDF and the document centroid vector should be dynamic and that involves higher computational cost for updating them. The proposed weighting scheme seeks to reduce such drawback seen in the TF-IDF scheme.

This paper reports some of the results obtained from our approach. The subsequent sections are organized as follows. Section II gives an account of the basic concepts of IR. Section III reviews related work, particularly on statistical unsupervised term weighting. Section IV describes the proposed term-weighting scheme and discriminative approach. Section V presents experimental setting and results while also discusses the key observations from the study. Finally, Section VI states some conclusions and outlines proposed future work.

II. ANTECEDENTS

A. Information Retrieval

An Information Retrieval (IR) system is an information system that stores, organizes and indexes the data for retrieval of relevant information responding to the user's query (user information need) [5]. Basically, an IR system contains the three following main components [6, 7]:

- The *documentary database*. This component stores the documents and their representations of information content. It is related to the indexer module, which generates a representation for each document by extracting the document features (document's terms). A *term* is a keyword or a set of keywords in the document.

- The *user's information need* (user's query or a set of queries) subsystem. In this component, users can state their information needs using a query or set of queries. Also, this component transforms the user's query into its information content by extracting the query features (query's terms) that correspond to document features.

- The *matching mechanism*. This component evaluates the degree to which each document in the documentary database satisfies the user's information need.

B. Information Retrieval Models

An IR model refers to the way in which the IR system organizes, indexes and retrieves information when responding to user's queries (user's information need). The IR model also specifies the method used for the user's query representation. From the literature, there are three prominent IR models: the first is referred to as the Boolean model, the second is known as the Vector Space Model (VSM) and the third is called the Probabilistic model [7].

The *Boolean model* is based on binary algebra for representing the term weights in documents and queries. In this model, the indexer module uses binary indexing for representing terms for each document (i.e., 1 if the term exists in that document and 0 otherwise). In this model, queries are expressed as logical statements using logical operators OR, AND and NOT (e.g. term1 AND term2 NOT term3). The limitations on this Boolean model are the following [8, 9]: (1) it needs a full matching between the user's query and the documents collection; (2) there is no difference expressed in the information content of terms in documents or queries even if one term is repeated frequently and another term occurs once; and (3) it can be difficult to formulate a complex user's information need using logical operators only.

The *probabilistic model* uses probability theory to build probability approaches that estimate the probability of a document being relevant to a certain query or not. Also, this model uses the probability of relevancy to a query for assigning weights to terms in documents and queries according to the queries training set or according to supervised weight learning. The limitation of the probabilistic model lies in the large set of queries used as a training set. The difficult and time-consuming aspects of the estimating mechanism are other limitations of the probability model [8, 9].

The *Vector Space Model* (VSM) is the most widely applied approach by researchers [8, 9]. In this model, a document and a query are represented as vectors in an *n*-dimensional space, where *n* is the number of distinguishing terms that are used as index terms for the documents in a collection. The values of the document dimensions are the weights of the index terms in the documents' space. The similarity between documents vectors and the query vector can be measured using a similarity function. There are many similarity functions that measure the similarity between documents vectors and user's query vector for retrieving relevant information need according their similarity values [7, 9]. In this paper, we use the Cosine Similarity as the matching function (see eq. 1). According to the study by Noreaultet. al. [10], this function is one of the best similarity measures for making angle comparisons between vectors.

$$Cosin\_Sim\ (D,Q) = \frac{\sum_{i=1}^{m} d_i \times q_i}{\sqrt{\sum_{i=1}^{m} {d_i}^2} \times \sqrt{\sum_{i=1}^{m} {q_i}^2}} \qquad (1)$$

In eq. (1) above, Cosin_Sim(D,Q) is the cosine similarity between the query and document vectors, d_i is the term weight of term *i* in document D, q_i is the term weight of term *i* in query Q and *m* is the number of terms in documents' space (documents' collection).

Most textual IR systems use keywords to retrieve documents. These systems first extract keywords from documents to act as index terms and then assign weights to each index term using various approaches. Such systems have two major difficulties. One is how to choose the appropriate keywords to act as index terms precisely. The other is how to assign the appropriate weights for each index term to represent precisely the information content or the importance of that index term in each document in the document collection.

C. Information Retrieval Systems Effectiveness

IR systems effectiveness is the most commonly used aspect for evaluating IR systems [6, 9]. The IR system effectiveness can be measured using the degree of retrieving relevant documents responding to the user's query. Where a given document that is called is relevant to a given query, this means the given document satisfies the information need by this query. *Precision* (P) and *Recall* (R) are the most used methods for measuring IR systems effectiveness. Precision is the ratio between the number of relevant retrieved documents divided by the total number of retrieved documents responding to a query. Recall is the ratio between the number of relevant retrieved documents divided by the total number of relevant documents in the IR document collection. The effectiveness function used here is the non-interpolated average precision, which is similar to average precision but with the cut-off points equivalent to the training documents. In this function, the documents are ranked and then the top-k documents are identified from the total number of retrieved documents [11, 12].

Let $d_1, d_2, \dots, d_{|D|}$ denote the sorted documents by decreasing order of their similarity measure function value, where $|D|$ represents the number of training documents. The function $r(d_i)$ gives the relevance value of a document d_i. It returns 1 if *d* is relevant, and 0 otherwise. The non-interpolated average precision is defined as follows:

$$AvgP = \frac{1}{|D|}\sum_{i=1}^{|D|} r(d_i) \times \sum_{j=1}^{|D|}\frac{1}{j} \qquad (2)$$

Where $r(d_i)$ returns 1 if d_i is relevant and 0 otherwise, and |D| represents the number of documents [11, 12].

III. Related Work

In this section, we briefly review statistical unsupervised term-weighting schemes. Weighting schemes can be classified into supervised and unsupervised approaches [13, 14]. From the literature on unsupervised statistical weighting schemes, we found that most of the term-weighting schemes proposed by researchers are a variation of the TF-IDF weighting scheme that was proposed by Salton and Buckley [15]. Some examples of TF-IDF term-weighting scheme variations include: ATC, Okapi [16] and Pivoted Document Length-IDF (LTU) [17, 18]. The equations used for each of these weighting schemes are as follows:

1) Basic TF-IDF weighting scheme [15, 17]:

$$W_{ij} = tf_{ij} \times \log\left(\frac{N}{n_i}\right) \qquad (3)$$

Where W_{ij} is the weight of term i in document j and tf_{ij} is the number of occurrences of term i in that document j. N is the number of documents in the document collection and n_i is the number of documents that contains term i in this document collection. From this equation, $\mathrm{IDF}_i = \log(N/n_i)$.

2) Augmented maximum term normalization-IDF (ATC) [15, 16]:

$$W_{ij} = \frac{\left(0.5+0.5 \times \frac{tf_{ij}}{\max\_tf_j}\right) \times \log\left(\frac{N}{n_i}\right)}{\sqrt{\sum_{i=1}^{m}\left[\left(0.5+0.5 \times \frac{tf_{ij}}{\max\_tf_j}\right) \times \log\left(\frac{N}{n_i}\right)\right]^2}} \qquad (4)$$

Where *m* is the number of terms in the documents space and max_tf_j is the maximum term frequency in document *j* (i.e., the term frequency for the highest term repeated in document *j*).

3) Okapi term-weighting scheme [16]:

$$\mathrm{W_{ij}} = \left(\frac{\mathrm{tf_{ij}}}{0.5+1.5 \times \frac{\mathrm{dl_j}}{\mathrm{avg\_dl}} + \mathrm{tf_{ij}}}\right) \times \log\left(\frac{\mathrm{N}-\mathrm{n_i}+0.5}{\mathrm{tf_{ij}}+0.5}\right) \qquad (5)$$

Where dl_j is the document length of document *j* (i.e., the summation of all terms frequencies in document *j*) and *avg_dl* is the average documents length on the document collection. Finally, the last term weighting scheme function is the LTU expressed below.

4) Pivoted document length normalization-IDF (LTU) [16]:

$$W_{ij} = \left(\frac{1+\log(tf_{ij})}{0.8+0.2 \times \frac{dl_j}{avg\_dl}}\right) \times log\left(\frac{N}{n_i}\right) \qquad (6)$$

All the above term-weighting schemes as well as other schemes mentioned in the literature, use some of the document collection characteristics, such as total numbers of documents in the collection and document frequency for a term (number of documents in the document collection that contain this term). In real-world IR systems, these characteristics should be variables over time because document collections are mostly dynamic instead of static nowadays.

IV. A New Term-weighting Scheme and Discriminative Approach

How to assign appropriate weights to terms is one of the critical issues in automatic term-weighting approaches. The new weighting scheme proposed here is called *Term*

Frequency Average Term Occurrences (TF-ATO) and is expressed by:

$$W_{ij} = \frac{tf_{ij}}{\#\ \text{ATO in document}\ j} \qquad (7)$$

and

$$\#\ \text{ATO in document}\ j = \frac{\sum_{i=1}^{m_j} tf_{ij}}{m_j} \qquad (8)$$

Where, m_j represents the number of unique terms in the document *j*. In other words, it is the number of index terms that exist in document *j*. This term weight differs from other variations of TF-IDF in that the discrimination approach uses the documents centroid as a threshold to remove less-significant weights from the documents. This discriminative approach can be represented by:

$$W_{ij} = \begin{cases} W_{ij} & if\ c_i < W_{ij} \\ 0 & if\ c_i \geq W_{ij} \end{cases} \qquad (9)$$

Where, C_i is the weight of term *i* in documents centroid vector and W_{ij} is the term weight of term *i* in document *j*. This discriminative approach is applied to every term weight W_{ij} in every document in the collection. The documents centroid vector is given by:

$$C = (C_1, C_2, \dots, C_i) \qquad (10)$$

and

$$\mathrm{C_i} = \frac{1}{\mathrm{N}} \sum_{\mathrm{j=1}}^{\mathrm{N}} \mathrm{W_{ij}} \qquad (11)$$

Where, *N* is the number of documents in the documents collection, C_i is the weight of term *i* in the centroid vector and W_{ij} is the term weight of term *i* in document *j*.

V. Implementation AND Experimental results

A. Documents Collections

The two documents collections used in this research are subsets of the TREC-9 filtering track (OHSUMED collection) [19, 20] and CISI documents collection [21]. The documents collections contain three textual materials which are: a set of documents, a set of queries and relevance judgement between documents and queries. For each query, a list of relevant documents is associated to it. The OHSUMED collection was set up by Hersh et al. [20] and has been used by several researchers [22]. This collection consists of abstracts from the Medline database from 1988 to 1991. The first subset consists of 70,825 documents from 1988 (OHSU88). The second collection consists of 74,869 documents from 1989 (OHSU89). The third subset consists of 148,162 documents from 1990 and 1991 (OHSU90-91). Each collection consists of a subset of queries. The relevance between documents and queries is graded as definitely or possibly relevant. We make no distinction between definitely and possibly relevant documents in our experiments and regard both grades as *relevant*. The CISI collection was selected from previous collections assembled at the Institute of Scientific Information (ISI). The CISI documents selected are about information science and consist of 1460 documents [21]. Table I shows the documents collections' characteristics.

TABLE I. DOCUMENTS COLLECTIONS CHARACTERISTICS

Collection	***No. Docs***	***No. of Queries***	***Vocabulary***
CISI	1,460	76	8,342
OHSU88	70,825	61	175,021
OHSU89	74,869	63	185,304
OHSU90-91	148,162	63	287,807

B. Building the Information Retrieval System

Information Retrieval systems manage their data resources (documents collection) by processing their words to extract and assign a descriptive content that is represented as index terms to documents or queries. In text documents, words are formulated with many morphological variants, even if they referred to the same concept. Therefore, the documents often undergo a pre-processing procedure before building the information retrieval system model.

The model proposed here is based on the vector space model (VSM) in which both documents and queries are represented as vectors (see Section II). The following are the procedures used in our IR system model for each document:

1) *Lexical analysis and tokenization of text* with the objective of treating punctuation, digits and the case of letters.

2) *Elimination of stop-words* with the objective of filtering out words that have very low discrimination values for matching and retrieval purposes.

3) *Stemming of the remaining words using Porter stemmer* with the objective of removing affixes (prefixes and suffixes) and allowing the retrieval of documents containing syntactic variations of query terms.

4) *Index terms selection* by determining which words/stems will be used as index terms.

5) *Assign weights in each document for each index term* using one of the weighting schemes mentioned in our implementation which gives the importance of that index term to a given document.

6) *Create documents' vectors of terms weights in the documents collection space* (create inverted and directed files using terms weights for documents from the documents collection).

7) *Apply the previous steps (1-6) in queries* to build queries' vectors.

8) For our proposed weighting scheme only (TF-ATO), there are two additional steps:

- *Create documents centroid vector from documents vectors by equations (9) and (10).*
- *Use documents centroid for normalizing documents vectors*. This can be done by removing small non-discriminative weights using documents centroid as a threshold.

9) Matching between documents vectors and each query using cosine similarity and retrieving corresponding documents under fixed 9-points recall values.

10) Rank the retrieved documents according to their cosine similarity measures in descending order and then get the top-10, top-15 and top-30 documents.

11) Compute precision values for top-10, top-15 and top-30 retrieved documents for each corresponding recall value for each query.

12) Compute average precision values for 100 queries in 9-points recall values for top-10, top-15 and top-30 retrieved documents. Also, compute non-interpolated average precision.

13) Repeat steps 5 to 12 for each weighting scheme and compare the results.

The above procedure has been used for experiments with static data stream. For the case of dynamic data stream, there are two approaches. The first one is to re-compute terms' weights for each document in the documents' collection by conducting the above procedure for each update to the documents collection using unsupervised machine learning. This of course, adds extra computation cost for every data update in a dynamic data stream. The second approach is to use partial supervised machine learning. This involves using IDF or the documents centroid in the next approach that is measured from the initial document collection; and then assigning terms weights to the new documents using the term frequency in the document multiplied by the corresponding IDF for the term that computes by the initial documents collection or using the discriminative approach in the second approach. Also, for the term-weighting approach proposed here, the old documents centroid vector is used for eliminating non-discriminative terms weights from the added documents. The second approach costs less in computation time but there is less effectiveness in both the proposed weighting scheme and TF-IDF. The cause of this drawback is the variation between the actual values of IDF or documents centroid in dynamic documents collection compared with the old values that are computed by the initial dataset.

Most of the proposed terms weighting schemes have drawbacks in their effectiveness if they do not re-compute their weighting scheme after every major update in the dataset. However, this issue has not been mentioned explicitly and this represents a drawback in the IR system effectiveness when considering dynamic data streams as well as static ones. The cost in effectiveness behind this issue has not been reported in published papers to the best of our knowledge.

C. *Experimental Results and Analysis*

In our experiments, the combination of the two documents collections was used with 100 queries as a training set in two different experiments. The first experiment considers that the dataset is static and we used three weighting schemes for comparing their effectiveness in that static dataset. Tables II, III and IV represent the first approach and from table II we can observe that the proposed weighting scheme TF-ATO gives high effectiveness compared to the TF-IDF weighting scheme. The average improving rate in precision (non-interpolated average precision) between TF-IDF and TF-ATO without discriminative approach is 6.921% when retrieving the top-10 documents for an average of 100 queries. The improvement is 41% between TF-IDF and TF-ATO when retrieving the top-10 documents using the discriminative approach. Table III shows the effectiveness for the weighting schemes when retrieving top-15 documents. The improvement ratio is 6.0794% between TF-IDF and TF-ATO without discriminative approach. The improvement value is 39.9% between TF-IDF and TF-ATO with the discriminative approach. In table IV, we see that the average improvement ratio is 8.993% between TF-IDF and TF-ATO without the discriminative approach when retrieving the top-30 documents for each query for an average of 100 queries. Finally, the average improving rate in precision (non-interpolated average precision) is 50.6% when retrieving the top-30 documents between TF-IDF and TF-ATO with discriminative approach.

TABLE II. AVERAGE RECALL-PRECISION FOR 100 QUERIES USING (TF * IDF) WEIGHTING SCHEME AND THE NEW WEIGHTING SCHEME WITH AND WITHOUT THE DISCRIMINATIVE APPROACH FOR TOP-10 DOCUMENTS RETRIEVED IN STATIC DATASET (STATIC DOCUMENTS COLLECTION) AS IN [12]

Recall	**Average Precision for 100 queries using TF-IDF Weighting Scheme and the new weighting scheme TF-ATO with and without the discriminative approach for top-10 documents retrieved (static dataset)**		
	TF-IDF	***TF-ATO without discriminative approach***	***TF-ATO with discriminative approach***
0.1	0.694337	0.780168	0.866938
0.2	0.491521	0.562977	0.692396
0.3	0.37271	0.4117	0.560153
0.4	0.268795	0.282411	0.428217
0.5	0.220324	0.207692	0.356833
0.6	0.189441	0.163513	0.269035
0.7	0.138761	0.143924	0.215532
0.8	0.120458	0.12396	0.158392
0.9	0.109246	0.109569	0.126471
Non-Interpolated Average Precision	0.289510333	0.309546	0.408218556

TABLE III. AVERAGE RECALL-PRECISION FOR 100 QUERIES USING TF-IDF WEIGHTING SCHEME AND THE PROPOSED WEIGHTING SCHEME TF-ATO WITH AND / WITHOUT CENTROID DISCRIMINATIVE APPROACH FOR TOP-15 DOCUMENTS RETRIEVED IN STATIC DATASET (STATIC DOCUMENTS COLLECTION)

Recall	Average Precision for 100 queries using TF-IDF Weighting Scheme and the new weighting scheme TF-ATO with and without the discriminative approach for top-15 documents retrievedin static documents collection		
	TF-IDF	*TF-ATO without discriminative approach*	*TF-ATO with discriminative approach*
0.1	0.748915	0.83113	0.893356
0.2	0.443985	0.480603	0.614755
0.3	0.339464	0.367001	0.52535
0.4	0.247683	0.236046	0.380559
0.5	0.19852	0.19936	0.322543
0.6	0.152038	0.155758	0.241101
0.7	0.13595	0.143715	0.213803
0.8	0.117432	0.120345	0.159072
0.9	0.110956	0.112662	0.139949
Non-Interpolated Average Precision	0.277215889	0.294068889	0.387832

TABLE IV. AVERAGE RECALL-PRECISION FOR 100 QUERIES USING TF-IDF AND THE NEW WEIGHTING SCHEME TF-ATO WITH AND WITHOUT DOCUMENT CENTROID DISCRIMINATIVE APPROACH IN STATIC DOCUMENTS COLLECTION

Recall	Average Precision for 100 queries using TF-IDF Weighting Scheme and the new weighting scheme TF-ATO with and without the discriminative approach for top-30 documents retrieved in static documents collection		
	TF-IDF	*TF-ATO without discriminative approach*	*TF-ATO with discriminative approach*
0.1	0.54473	0.614331	0.727991
0.2	0.332414	0.367622	0.543905
0.3	0.233081	0.259505	0.408616
0.4	0.181745	0.19655	0.321742
0.5	0.156084	0.164589	0.265651
0.6	0.137364	0.143184	0.209802
0.7	0.124346	0.132056	0.167472
0.8	0.115888	0.119354	0.14487
0.9	0.10914	0.111591	0.123859
Non-Interpolated Average Precision	0.214976889	0.234309111	0.323767556

From the results of this first experiment, it is clear that the proposed TF-ATO weighting scheme gives more effectiveness (high average precision values) when compared to TF-IDF in static documents collection. Also, there is an improvement by using the documents centroid as a discriminative approach with the proposed weighting scheme. Also, the proposed discriminative approach reduces the size of the documents in the dataset by removing non-discriminative terms and less significant weights for each document. Using the documents centroid gives an average reduction in size of 2.3% from the actual dataset size compared to 0% reduction when using TF-IDF. Further, from Fig. 1, we can observe the difference between each weighting scheme in retrieving top-k documents where k equals to 10 or 15 or 30. This figure represents the variation in the applied weighting schemes in static documents collection.

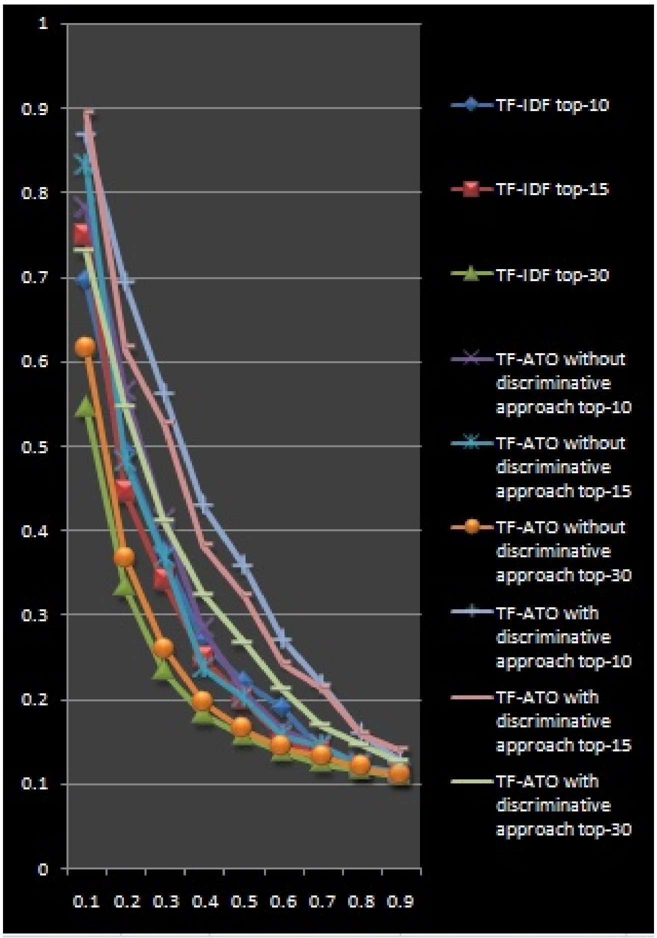

Fig. 1. **Representations of precisions in static documents collections for tables II, III and IV.**

The second experiment uses the dataset as a dynamic dataset by dividing the documents collection into 31 parts. Then, one part has been taken and processed as in steps 1-8 in the first experiment for getting the IDF terms values or the documents centroid vector in our approach for the whole collection. The part is considered as an initial document collection and then the collection is updated by increasing its size for 31 times the initial one without re-computing (updating) the initial measures for IDF or document centroid in the system by assigning IDF weights using the initial IDF weights that are computed by the first part of the 31 parts dataset or by using the initial document centroid of the first part to discriminate the whole 31 parts collection. This is because, every update for re-computing IDF and assigning the new weights for terms in the dynamic dataset will cost O(N^2MLogM) for updating the documents in the documents collection [4] where N is the number of documents in the collection and M is the number of terms in the term space. Thus, there is more cost for updating the system in both approaches but there is no extra cost for using the proposed weighting scheme without normalization. In this second experiment, we check the effectiveness of both weighting schemes using dynamic data streams. Tables V and VI show the results of the second experiment. From tables V and VI, we observe that there is a reduction in effectiveness compared to the case with static data streams but the proposed weighting scheme TF-ATO still gives better effectiveness values than those produced with the TF-IDF weighting scheme. From tables V and VI, we find that the average improvement ratio (for 100 queries) of the proposed weighting scheme compared to TF-IDF is 42.380% when retrieving the top-10 documents responding to each given query. The improving rate is 34.932% when retrieving the top-15 documents and 23.71% when retrieving the top-30 documents. Further, from Fig. 2, we can observe the difference between each weighting scheme in retrieving top-k documents where k equals to 10 or 15 or 30. This figure represents the variation in the two applied weighting schemes (TF-IDF and TF-ATO with discriminative approach) in the case of a dynamic documents collection.

TABLE V. AVERAGE RECALL-PRECISION FOR 100 QUERIES USING (TF * IDF) WEIGHTING SCHEME IN DYNAMIC DATASET (DYNAMIC DOCUMENTS COLLECTION)

Recall	**Average Precision for 100 queries using TF*IDF Weighting Scheme for dynamic data stream (30 times) added size to the initial dataset size**		
	Precision top-10	***Precision top-15***	***Precision top-30***
0.1	0.516069	0.559997	0.399095
0.2	0.328678	0.306925	0.241575
0.3	0.259872	0.241953	0.200439
0.4	0.201592	0.176876	0.169386
0.5	0.158748	0.162268	0.156651
0.6	0.138249	0.136316	0.146238
0.7	0.125598	0.131291	0.136308
0.8	0.116629	0.121365	0.126631
0.9	0.110685	0.115916	0.116953
Non-Interpolated Average Precision	0.217346667	0.216989667	0.188141778

TABLE VI. AVERAGE RECALL-PRECISION FOR 100 QUERIES USING (TF / AVERAGE TERMS OCCURRENCE WITH DOCUMENT CENTROID NORMALIZATION) WEIGHTING SCHEME IN DYNAMIC DATASET (DYNAMIC DATA STREAM)

Recall	**Average Precision for 100 queries using TF/average of term occurrence Weighting Scheme Proposed Weighting Scheme with centroid normalization for dynamic data stream (30 times added size to the initial dataset size)**		
	Precision top-10	***Precision top-15***	***Precision top-30***
0.1	0.775654	0.81337	0.585324
0.2	0.560882	0.467495	0.362318
0.3	0.401773	0.368764	0.266102
0.4	0.282682	0.241149	0.199866
0.5	0.212777	0.205037	0.167849
0.6	0.170258	0.158062	0.145352
0.7	0.145926	0.14567	0.134668
0.8	0.12516	0.12147	0.121146
0.9	0.11002	0.114088	0.112121
Non-Interpolated Average Precision	0.309459111	0.292789444	0.232749556

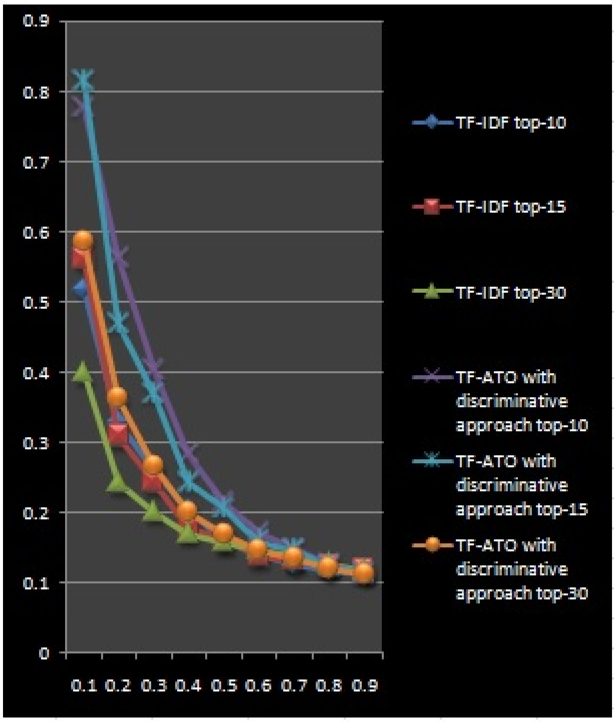

Fig. 2. **Representations of precisions in dynamic documents collections for tables V and VI.**

VI. CONCLUSIONS AND FUTURE WORK

The proposed *Term Frequency - Average Term Occurrences* (TF-ATO) weighting scheme can be considered competitive based on the results from our experiments. This scheme gives higher effectiveness compared to the TF-IDF weighting scheme in both cases of static and dynamic documents collections. At the same time, the documents centroid vector can act as a threshold in normalization to discriminate between documents or better effectiveness in retrieving relevant documents. Also, we observed in our experiment results that there is a variation and a reduction in system effectiveness when using dynamic documents collections instead of static documents collections. We also observed that there is a cost behind every major update in documents collection.

There is further research work to be carried out based on the results presented in this paper. The proposed *Term Frequency Average Term Occurrences* (TF-ATO) weighting scheme can be further examined in different aspects and approaches of information retrieval to check if it is valuable compared to other weighting approaches in Web and XML retrieval. Also, TF-ATO can be used in other fields that require a weighting scheme for text features. For example, document summarization, recommender systems and big data analytics. The discriminative approach can be used in documents clustering and categorization for better results. In this work we used static clustering only and future work can consider dynamic and incremental clustering approaches.

REFERENCES

[1] SINTEF. "Big Data, for better or worse: 90% of world's data generated over last two years," Science Daily, 22 May 2013.

www.sciencedaily.com/releases/2013/05/130522085217.htm.

[2] Grant S. Ingersoll, Thomas S. Morton, and Andrew L. Farris, Taming Text: How to Find, Organize, and Manipulate it. Manning Publications Co., Greenwich, CT, USA, pp. 2-3, 2013.

[3] Feldman, Susan, "Hidden Costs of Information Work: A Progress Report," International Data Corporation, 2009.

[4] Joel W. Reed, Yu Jiao, Thomas E. Potok, Brian A. Klump, Mark T. Elmore, and Ali R. Hurson. TF-ICF: A new term weighting scheme for clustering dynamic data streams. In ICMLA '06: Proceedings of the 5th International Conference on Machine Learning and Applications, pages 258-263, Washington, DC, USA, 2006. IEEE Computer Society.

[5] Gerard Salton and Michael J. McGill, Introduction to Modern Information Retrieval, McGraw-Hill, Inc., New York, NY, USA, 1986.

[6] O. Cordon, E. Herrera-Viedma, C. López-Pujalte, M. Luque, and C. Zarco, "A review on the application of evolutionary computation to information retrieval," International Journal of Approximate Reasoning, vol. 34:2-3, pp. 241-264, 2003.

[7] Ricardo Baeza-Yates and Berthier Ribeiro-Neto, Modern Information Retrieval (2nd ed.), Addison-Wesley Publishing Company, USA, 2011.

[8] A. Vinciarelli, "Application of information retrieval techniques to single writer documents," Pattern Recognition Letters, Vol. 26, Issue 14, pp. 2262-2271, 2005. http://dx.doi.org/10.1016/j.patrec.2005.03.036.

[9] E. Greengrass. Information retrieval: A survey. Technical Report TR-R52-008-001, Centre of Architectures for Data-Driven Information Processing (CADIP), University of Maryland, US, 2000.

[10] T. Noreault, M. McGill, and M. Koll, "A performance evaluation of similarity measures, document term weighting schemes and representations in a Boolean environment," Information Retrieval Research, London: Butterworths, 1981.

[11] C. H. Chang and C. C. Hsu. The design of an information system for hypertext retrieval and automatic discovery on WWW. Ph.D. thesis,Department of CSIE, National Taiwan University, 1999.

[12] K. L. Kwok. "Comparing representations in Chinese information retrieval". ACM SIGIR'97, Philadelphia, PA, USA, pp. 34 -41, 1997.

[13] R. Jin and J. Y. Chai, "Learn to weight terms in information retrieval using category information," The 22nd International Conference on Machine Learning (ICML2005), Germany, ACM International Conference Proceeding Series, Vol. 119, pp. 353 - 360, Aug 7-11, 2005.

[14] Sa-Kwang Song and Sung Hyon Myaeng. 2012. "A novel term weighting scheme based on discrimination power obtained from past retrieval results," Information Processing and Management, Vol. 48, Issue 5, pp. 919-930, September 2012.

[15] Gerard Salton, Christopher Buckley, "Term-weighting approaches in automatic text retrieval," Information Processing and Management: an International Journal, v.24 n.5, p.513-523, 1988.

[16] R. Jin, C. Falusos, and A. G. Hauptmann, "Meta-scoring: Automatically Evaluating Term Weighting Schemes in IR without Precision-Recall," In Proceedings of the 24th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '01), ACM, New York, NY, USA, 83-89, 2001. http://doi.acm.org/10.1145/383952.383964.

[17] Karen Spärck Jones, "A statistical interpretation of term specificity and its application in retrieval," Journal of Documentation, Volume 60 Number 5, pp. 493-502, 2004, and previously from Journal of Documentation, Volume 28, Number 1, pp. 11-21, 1972.

[18] Amit Singhal, Chris Buckley, and Mandar Mitra, "Pivoted Document Length Normalization," In Proceedings of the 19th Annual International ACM SIGIR Conference On Research and Development in Information Retrieval (SIGIR '96), ACM, New York, NT, USA, pp. 21-29, 1996.

[19] Text Retrieval Conference (TREC), TREC-9 Filtering Track Collections, Last updated: Tuesday, 10 Jul, 2007. http://trec.nist.gov/data/t9_filtering.html.

[20] William Hersh, Chris Buckley, T. J. Leone, and Davide Hickam, "OHSUMED: an interactive retrieval evaluation and new large test collection for research," In Proceedings of the 17th Annual International ACM SIGIR Conference on Research and Development In Information Retrieval (SIGIR '94), W. Bruce Croft and C. J. Van Rijsbergen (Eds.), Springer-Verlag, New York Inc., New York, NY, USA, pp. 192-201, 1994.

[21] E. A. Fox, "Characterization of Two New Experimental Collections in Computer and Information Science Containing Textual and Bibliographic Concepts," Computer Science Technical Reports, Cornell University, 1983. ftp://ftp.cs.cornell.edu/pub/smart.

[22] Fabrizio Sebastiani, "Machine Learning in Automated Text Categorization," ACM Computing Surveys, Vol. 34, No. 1, pp. 1-47, March 2002, http://doi.acm.org/10.1145/505282.505283.