



HAL
open science

Should one use term proximity or multi-word terms for Arabic information retrieval?

Abdelkader El Mahdaouy, Éric Gaussier, Saïd Ouatik El Alaoui

► To cite this version:

Abdelkader El Mahdaouy, Éric Gaussier, Saïd Ouatik El Alaoui. Should one use term proximity or multi-word terms for Arabic information retrieval?. *Computer Speech and Language*, 2019, 58, pp.76-97. 10.1016/j.csl.2019.04.002 . hal-02132287

HAL Id: hal-02132287

<https://hal.science/hal-02132287>

Submitted on 22 Oct 2021

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



Distributed under a Creative Commons Attribution - NonCommercial 4.0 International License

Should one Use Term Proximity or Multi-Word Terms for Arabic Information Retrieval?

Abdelkader El Mahdaouy^{a,b,*}, Eric Gaussier^a, Saïd Ouatik El Alaoui^b

^aUniversity Grenoble Alps, CNRS, Grenoble INP, LIG, F-38000 Grenoble, France

^bLIM Laboratory, Sidi Mohamed Ben Abdellah University, Faculty of Sciences Dhar el Mahraz, Fez, Morocco

Abstract

Recently, several Information retrieval (IR) models have been proposed in order to boost the retrieval performance using term dependencies. However, in the context of the Arabic language, most IR researchers have focused on the problem of stemming, which is highly challenging in this language. In this paper, we propose to explore whether term dependencies can help improve Arabic IR systems, and what are the best methods to use. To do so, we consider both explicit term dependencies based on multi-word terms (MWTs) that are extracted using syntactic patterns and statistical filters, as well as implicit ones based on the notion of cross-terms or term proximities. Our experiments, performed on standard TREC Arabic IR collections, show the importance of taking into account term dependencies for Arabic IR. To the best of our knowledge, this is the first study that provides complete extensions, and their comparison, of most standard IR models to deal with term dependencies in the Arabic language.

Keywords: Arabic Information Retrieval, Multi-Word Terms, Term Proximity, Term Dependence IR Models

1. Introduction

Information Retrieval (IR) deals with the representation, storage, organization, and access to information items. The main goal of an IR system is to return a subset of documents whose content is relevant to user information needs, which are expressed by queries. Traditional information retrieval models are based on term independence assumption and thereby they represent documents and queries with bags-of-words. Hence, the estimation of a document's relevance to a query is based on the shared keywords between them. In order to catch the notion of document relevance, many probabilistic IR models, including the Probabilistic Relevance Framework (Robertson et al., 1994), the Language Modeling approach (Ponte and Croft, 1998) and the Divergence from Randomness approach (Amati and Van Rijsbergen, 2002) with the family of Information-Based Models (Clinchant and Gaussier, 2010), have been proposed, relying on statistics such as within-document frequency, inverse document frequency and document length. The drawback of the bag-of-word models is that single terms are often ambiguous and can refer to different concepts according to their contexts (Haddad, 2003; Habert and Jacquemin, 1993). Bag-of-word models based on single terms do not fully take into account the associations between document or query words (Sordoni et al., 2013).

For Arabic IR, most studies focus on developing or comparing word stemming techniques (Abu El-Khair, 2007; Mustafa et al., 2008; Darwish and Magdy, 2014) and thereby rank documents based on the shared stemmed words between documents and queries. These studies can be classified, according to the level of analysis, as heavy stemming (root-based approaches) (Khoja and Garside, 1999) and light stemming (stem-based approaches) (Larkey et al., 2002). Despite the fact that earlier studies show that retrieving Arabic

*Corresponding author

Email addresses: abdelkader.elmahdaouy@imag.fr (Abdelkader El Mahdaouy), eric.gaussier@imag.fr (Eric Gaussier), said.ouatikelalaoui@usmba.ac.ma (Saïd Ouatik El Alaoui)

20 documents based on roots is more effective (Al-Kharashi and Evens, 1994; Abu-Salem et al., 1999), most
 21 recent studies (Larkey et al., 2007; Abdelali et al., 2016) rely on stem-based approaches. A drawback of
 22 heavy stemmers is that they may conflate semantically different words to the same root since each root can
 23 generate hundreds of words of different meanings (Beesley, 1996). On the contrary, most light stemming
 24 methods fail to discriminate conjunctions and prepositions from the core words (Nwesri et al., 2005; Darwish
 25 and Mubarak, 2016). Additionally, most light stemmers cannot extract the correct stem of broken plurals
 26 (Goweder et al., 2004). Hence, light stemmers may conflate words with the same meaning to different stems.
 27 As one can see, both stemming approaches introduce ambiguities in the text representation. Moreover,
 28 other levels of ambiguity present significant challenges to Arabic Natural Language Processing applications
 29 (Maamouri and Bies, 2010). In particular, the absence of the representation of diacritics (short vowels) in
 30 normal texts increases dramatically the number of ambiguities. Farghaly (2004) pointed out that for most
 31 languages, the average of ambiguities for a token at SYSTRAN is estimated around 2.3, whereas it reaches
 32 19.2 in the Modern Standard Arabic (MSA).

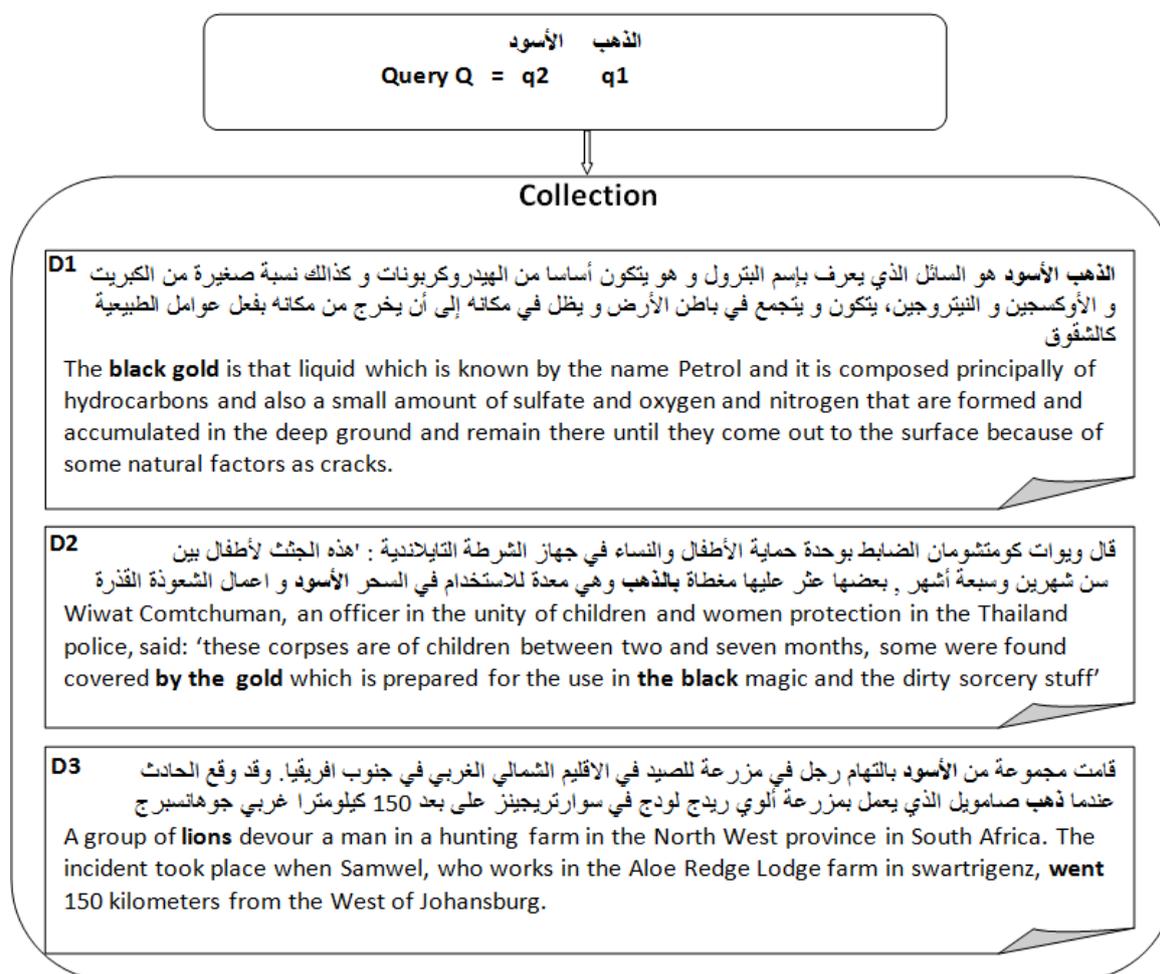


Fig. 1. Example of an ambiguous query in Arabic IR

33 Fig. 1 illustrates one drawback of the bag-of-words representation for Arabic IR. Assume that we have
 34 a query $Q = \{ \text{"الذهب الأسود"} \}$ (translated to "black gold" or "oil") that consists of two terms and let d_1 ,
 35 d_2 and d_3 be three documents in the collection. The first document is the only one relevant to the query
 36 and deals with oil and its chemical elements. The second document contains both query terms, and even

37 though the term "الذهب" refers to "gold" and "الأسود" to "black", these two terms are independent in the
38 document, which is not relevant to the query. In the third one, both query terms appear: the terms "الأسود"
39 and "ذهب" refer to "the lions" and "went" respectively. The problem here is caused by the ambiguity
40 of both query terms in the document due to the absence of diacritics. Thus, any bag-of-word model will
41 assign approximately the same scores to the three documents if they have approximately the same length.
42 Dealing with the aforementioned challenges in the context of Arabic IR requires intelligent models and a
43 more sophisticated representation of documents and queries.

44 Although the field of Arabic information retrieval has witnessed tangible progress, retrieving Arabic
45 documents using term dependencies (term proximity and explicit multi-word terms (MWTs)) remains un-
46 derexplored. To the best of our knowledge, there is only one study that has investigated indexing MWTs
47 for Arabic IR (Boulaknadel et al., 2008b). The evaluation in the latter study is performed using a small
48 corpus from the environment domain (1062 documents containing 475148 words). In this paper, we inves-
49 tigate Arabic document indexing and retrieval on large standard Arabic IR collections using MWTs that
50 are extracted using a complex linguistic filter to deal with MWT variations and more elaborate statistical
51 filter that consider contextual information and both termhood and unithood information (El Mahdaouy
52 et al., 2013). Moreover, we explore a wide range of proximity-based models for Arabic IR based on term
53 dependencies, using and comparing three different stemming approaches, respectively proposed in (Khoja
54 and Garside, 1999; Larkey et al., 2007; Abdelali et al., 2016). Our aim is to evaluate the impact of taking into
55 account (proximity) dependencies among query terms on the accuracy of Arabic IR. To do so, we compare
56 the use of different word level analysis for explicit MWTs and term proximity operators, so as to rely on
57 representations that go beyond bag-of-word IR models for Arabic documents. The questions we address are
58 the following:

- 59 1. Can the proximity-based models and the use of MWTs improve the retrieval performance when apply-
60 ing different levels of word analysis for Arabic documents? This question is of particular interest for
61 the Arabic heavy stemmer (Khoja and Garside, 1999) in which many words with different meanings
62 are grouped in the same index descriptor.
- 63 2. Can explicit MWTs that are extracted using a complex pipeline (linguistic and statistical filtering)
64 significantly outperform term proximity based IR models?

65 Besides these points, we believe that this is the first study that provides: (a) a complete cross-term extension
66 for standard IR models, (b) a complete comparison of the most important IR models integrating term
67 dependencies (18 different models are compared in our experiments), in the context of Arabic IR, and (c) a
68 compound condition that allows to characterize the different models.

69 We focus in this study on Arabic collections for several reasons: (a) the Arabic language is morphologi-
70 cally rich and there is no real consensus, in past experiments, on which stemmer to use for IR; we address
71 this problem by performing an extensive comparison of different stemming approaches, including the recent
72 Farasa stemmer (Abdelali et al., 2016), coupled with five different IR models, from different families; (b) the
73 Arabic language is also rich for compound production; if most languages rely on a single composition mode,
74 either roman, corresponding to *Noun preposition Noun* sequences, or germanic, corresponding to *Noun Noun*
75 sequences, to produce compounds and terms, the Arabic language relies on both; the integration of such
76 elements in IR may thus be more important and may lead to different conclusions than the ones obtained
77 in other languages; (c) lastly, contrary to some other languages, as English, German or French for example,
78 we know of no complete study devoted to the impact of MWTs and proximity operators on Arabic IR; the
79 goal of this study is precisely to assess this impact.

80 The remainder of the paper is organized as follows. Section 2 presents the related work. Section 3
81 describes the method we have retained to extract MWTs. Section 4 describes the different IR models and
82 their extensions retained in this study. Section 7 presents the experiments conducted while Section 8 discusses
83 the results obtained. Finally, Section 9 concludes this work and presents some perspectives.

84 The notations we use throughout the paper are summarized in Table 1.

Table 1
Notations used in the paper

Notation	Description
x_w^q	Number of occurrences of word w in query q
x_w^C	Number of occurrences of word w in collection C
x_w^d	Number of occurrences of w in document d
t_w^d	Normalized version of x_w^d
x_p^q	Number of occurrences of MWT p in query q
x_p^C	Number of occurrences of MWT p in collection C
x_p^d	Number of occurrences of p in document d
t_p^d	Normalized version of x_p^d
l_d	Length of document d
l_{avg}	Average document length
N	Number of documents in the collection
N_w	Number of documents containing w
$ C $	Number of words in collection C
$RSV(q, d)$	Retrieval Status Value of document d for query q

2. Related Work

Due to the rich and complex morphology of Arabic language, text stemming has received much attention over the last decades. Most early studies in Arabic IR, conducted using small test collection, showed that roots yield to a better performance than stems and words. Al-Kharashi and Evens (1994) investigated manual indexing of Arabic document using small test collection containing 335 documents and 29 queries. They manually built indexing dictionaries of the test collection containing 526 roots, 725 stems, and 1,126 words. The evaluation results showed that the root-based indexing method outperforms both stem-based and word-based indexing methods. In a similar work, Hmeidi et al. (1997) and Abu-Salem et al. (1999) concluded that indexing Arabic documents using roots significantly improves the performance in comparison to stems and words. In order to automatically extract Arabic words roots, several stemmers are proposed (Al-Shawakfa et al., 2010; Al-Kabi et al., 2011). One of the best-known algorithms of the root-based approach was introduced by Khoja and Garside (1999). This algorithm relies on lists of patterns and valid Arabic roots. After every prefix or suffix removal, the algorithm matches the remaining stem with the list of patterns to extract the root. Then, the extracted root is validated using the list of valid roots. If no root is found, the algorithm returns the original word. This algorithm constitutes the basis for many other later works. According to several Arabic IR research studies (Aljlayl and Frieder, 2002; Larkey et al., 2007), the Khoja algorithm significantly outperforms the words-based indexing method.

The other approach to deal with the Arabic morphology is light stemming. This approach is widely considered as the most promising to Arabic IR due to its great success on the Arabic TREC 2001 and TREC 2002 evaluation campaigns (Abu El-Khair, 2007; Darwish and Magdy, 2014). Most algorithms in this approach use lists of prefixes and suffixes to deduce the stem for inflected words. Aljlayl and Frieder (2002) investigated the impact of stemming in improving Arabic IR. They proposed a root extractor based on Khoja algorithm (Khoja and Garside, 1999) and a novel light stemmer. The obtained results on the Arabic TREC 2001 test collection show that their light stemmer significantly outperforms both root and word indexing methods. Larkey et al. (2002) introduced several light stemming algorithms to improve the performance of Arabic IR. These algorithms differ according to the number and the depth of the removed prefixes and suffixes. The Light10 algorithm introduced in (Larkey et al., 2007) consists in combining lists of suffixes and prefixes already used in (Larkey et al., 2002). The obtained results on TREC 2001 and TREC 2002 show that the Light10 is significantly better than Khoja algorithm (Khoja and Garside, 1999). Other researchers proposed machine learning-based methods to perform light stemming through learning models for word segmentation (Habash and Rambow, 2009; Darwish and Mubarak, 2016). Pasha et al. (2014) proposed the MADAMIRA¹ system for Arabic morphological analysis and disambiguation. This

¹<https://camel.abudhabi.nyu.edu/madamira/>

118 system combines the best features of MADA (Habash and Rambow, 2005, 2009; Habash et al., 2013) and
119 AMIRA (Diab, 2009). Indeed, MADAMIRA provides applications for tokenization/segmentation, part-of-
120 speech tagging, morphological disambiguation, lemmatization, diacritization, named entity recognition, and
121 base phrase chunking. Recently, another fast and accurate Arabic text processing toolkit, called Farasa²,
122 has been proposed. This toolkit consists of modules for tokenization/segmentation, part-of-speech tagging,
123 Arabic text diacritization, and dependency parsing. Farasa segmenter is based on SVM-rank using linear
124 kernels (Abdelali et al., 2016; Darwish and Mubarak, 2016). This segmenter relies on several features and
125 lexicons to rank the possible segmentations for a given word. Feature vectors were built for each possible
126 segmentation and mark a valid one for each word. Farasa showed comparable segmentation accuracy to
127 the state-of-the-art MADAMIRA word segmenter, while being significantly faster. In the context of Ara-
128 bic IR, Abdelali et al. (2016) showed significant improvement with Farasa over MADAMIRA and Stanford
129 segmenters as well as surface words on the TREC2001/2002 test collection. Moreover, they reported that
130 Farasa is an order of magnitude faster than Stanford and two orders of magnitude faster than MADAMIRA.
131 A successful integration of Farasa toolkit modules, from word segmentation to constituency parsing, into
132 community Question Answering (cQA) architecture using the UIMA-based framework for Arabic is intro-
133 duced in (Romeo et al., 2017). The proposed UIMA pipeline is used to extract mainly lexical and syntactic
134 features from Arabic texts. These features are used to train their machine learning models for cQA.

135 Ranking documents using term dependencies has been of a central interest in the IR community (Croft
136 et al., 1991; Metzler and Croft, 2005; Sordoni et al., 2013). These dependencies are integrated into IR
137 systems using two approaches (Gao et al., 2004). The first approach consists of extending the representation
138 space of documents and queries. Specifically, term dependencies are indexed as additional features arising
139 from phrases or multiword units. For instance, Fagan (1987) proposed a method in the mid-eighties for
140 automatic phrase indexing. First, he showed that the statistical phrases improve the performance obtained
141 by using single terms. Second, he applied linguistic filtering to compare the statistical and linguistic phrase
142 indexing. The experimental results show that there is no significant difference between the use of syntactic
143 or statistical phrases. Croft et al. (1991) proposed to use phrases in natural language queries to construct
144 structured queries, and use the latter in the probabilistic model based on inference networks. The results on
145 the CACM collection show that this approach improves the performance of single term indexing. Moreover,
146 phrases that are automatically extracted from a natural language query have almost the same performance
147 as those selected manually. Unlike (Fagan, 1987; Croft et al., 1991), Mitra et al. (1997) concluded that
148 the use of phrases (statistical phrases and syntactic phrases) do not have a major effect on the retrieval
149 performance if a good ranking model is considered. Additionally, Haddad (2003) carried out a Noun Phrase
150 indexing and mining for a French IR system. The results show that combining noun phrase indexing with
151 associative relations can improve the retrieval performance.

152 Automatic MWT extraction is a very important task to many NLP applications, such as terminology
153 extraction, information retrieval, question answering, and text classification. Three main approaches have
154 been proposed for MWTs extraction, namely, linguistic approach, statistical approach, and hybrid approach.
155 The linguistic approach relies on the use of linguistic filters to extract *n-grams* that fit the specified syntactic
156 patterns. The statistical approach makes use of association measure that characterizes the strength of the
157 sequence as a unit, which is called Unithood, and the degree of relatedness to a specific domain concept, which
158 is called Termhood (Kageura and Umino, 1996). The hybrid approach extracts MWTs using linguistic filters
159 and then ranks the resulting list of MWT candidates using association measures. Many hybrid methods
160 were proposed for automatic extraction of Arabic MWTs (Boulaknadel et al., 2008a; Bounhas and Slimani,
161 2009; El Mahdaouy et al., 2013). In the field of IR, Jacquemin et al. (1997) proposed a system for MWTs
162 expansion for indexing and retrieval using morphology and syntax. The main idea of the latter work is
163 combining the parsing over a seed of term list with derivational morphology, in order to achieve higher
164 coverage of MWT indexing and retrieval.

165 More recently, Drymonas et al. (2010) proposed the Term Similarity and Retrieval Model (TSRM) that
166 is based on computing the similarity among MWT using lexical and contextual criteria. The experimental

²<http://qatsdemo.cloudapp.net/farasa/>

167 results showed that the TSRM outperforms the Vector Space Model (VSM). [Boulaknadel et al. \(2008b\)](#)
168 adapted a hybrid method for MWT extraction in order to index Arabic documents from the environment
169 domain. They showed an improvement over the baseline BM25 model. [Zhang et al. \(2011\)](#) investigated
170 the text representation problem using three kinds of indexing, namely, TF*IDF, LSI, and MWT units
171 for information retrieval and text categorization. The obtained results showed that indexing MWT is
172 comparable to TF*IDF and both of them are better than LSI for English and Chinese text categorization.
173 For information retrieval, the results showed that MWTs have a better performance than TF*IDF in English
174 document collection, but the best results are obtained by TF*IDF for Chinese document collection. [SanJuan
175 and Ibekwe-SanJuan \(2010\)](#) have investigated the use of MWTs as meaningful text units in order to represent
176 queries for focused IR. The MWTs are used for interactive query expansion, automatic query expansion,
177 and combining both query expansion methods to boost the retrieval performance. Experiments performed
178 on three different collections showed promising results for both standard and focused retrieval.

179 The second approach to term dependence models represents documents and queries using only single
180 word terms ([Gao et al., 2004](#)). Whereas, term dependencies are incorporated into IR models mainly as joint
181 probabilities of their constituents ([Sordoni et al., 2013](#)). Early work in this approach relied on statistical
182 modeling of n-grams to capture terms dependencies. [Song and Croft \(1999\)](#) proposed a general and intuitive
183 language model to capture term dependencies among adjacent words. In order to deal with the data sparsity
184 problem, they used several smoothing techniques, including the Good-Turing estimate, curve-fitting func-
185 tions, and model combinations. [Srikanth and Srihari \(2002\)](#) suggested the biterm language model to relax
186 the word ordering constraint. This model introduces three approximation methods for biterm probabilities.
187 The results showed that the different biterm approximation methods achieve comparable performance to
188 the bigram language model ([Song and Croft, 1999](#)). [Gao et al. \(2004\)](#) proposed the dependence language
189 model to capture dependencies between distant words as well as adjacent word pairs. They introduced
190 the linkage of a query as a hidden variable, that expressed the term dependencies within the query as
191 an acyclic, planar, undirected graph. Although these models showed some improvement over the unigram
192 language model, the improvement was smaller than expected and they have a higher computational cost
193 due to dependency parsing or n-gram models ([Gao et al., 2004](#); [Zhai, 2008](#)). In order to deal with these
194 limitations, [Huston et al. \(2014\)](#) presented a new method for n-gram indexing that reduces the space re-
195 quirement and efficiently approximates their statistics. The obtained results on Robust-04 and GOV2 test
196 collections showed comparable performance to the sequential dependencies of the MRF proximity-based IR
197 model ([Metzler and Croft, 2005](#)). Proximity-based IR models boost the retrieval performance of documents
198 when query terms occur in close proximity. Several models are proposed to enhance the performance of
199 bag-of-words model by exploiting proximity features. For the family of divergence from randomness models,
200 [Peng et al. \(2007\)](#) proposed the incorporation of term dependencies to the DFR framework. In the context
201 of the BM25 model, many researchers proposed to extend the unigram model including ([Zhao et al., 2011](#);
202 [He et al., 2011](#); [Zhu et al., 2012](#)). For the language modeling approach, [Metzler and Croft \(2005\)](#) proposed
203 the Markov Random Field model for term dependencies. In another work, [Lv and Zhai \(2009\)](#) presented
204 the Positional Language Model (PLM) to implement proximity heuristic and passage retrieval in a unified
205 language model. Moreover, [Shi and Nie \(2010\)](#) proposed a variable dependency model in order to consider
206 distant dependencies. Unlike the MRF model, the underlying model weights each dependency according to
207 its utility. They showed that this model is effective for ranking Chinese documents using different types
208 of dependencies indexes. Furthermore, [Sordoni et al. \(2013\)](#) suggested the Quantum Theory (QT) as a
209 framework for modeling term dependencies. Hence, they developed generalized Quantum Language Model
210 (QLM) for IR by adopting the probabilistic framework of QT.

211 However, ranking Arabic documents based on term dependencies remains yet under-explored and there
212 has been no attempt to investigate the impact of MWTs and term proximity in the context of Arabic
213 language. In the remainder, we first review the method we have adopted for Arabic MWT extraction, prior
214 to review standard IR models and their extensions to take into account (either directly or indirectly) term
215 dependencies.

216 3. Arabic MWT extraction

217 We have used the state-of-the-art Arabic MWT extraction method of (El Mahdaouy et al., 2013) that
 218 consists of linguistic and statistic filtering of MWT candidates. At the linguistic filtering step, the corpus is
 219 tagged using the AMIRA Part of Speech (POS) Tagger (Diab, 2009) which is trained from the Penn Arabic
 220 TreeBank (PATB). Then, the linguistic filter extracts MWT candidates that fit the following syntactic
 221 patterns:

- 222 • *Noun* + (*Noun|ADJ*) + (*Noun|ADJ*)
- 223 • *Noun* + (*Noun|ADJ*)
- 224 • *Noun* + *Prep* + *Noun*

225 The last step of linguistic filtering consists of handling the problem of MWT variations by taking into
 226 account the four types of variations (graphical, inflectional, morpho-syntactic and syntactic variations)
 227 mentioned in (El Mahdaouy et al., 2013). Graphical variations of Arabic MWTs are resolved by normal-
 228 izing the text. These variations concern the orthographic errors that occur in writing particular letters.
 229 Inflectional variations are due to the inflectional nature of the Arabic language and concern gender, number
 230 and definiteness. The previous variations do not affect the Part-Of-Speech (POS) tags of MWT compo-
 231 nents, and are handled by removing suffixes and definite articles. The morpho-syntactic variations are
 232 related to the derivational morphology; they affect the internal structure of MWTs without changing the
 233 order of their components. The aforementioned variations concern syntactic POS tags transformations:
 234 *Noun1 Noun2* \Leftrightarrow *Noun1 Adj* \Leftrightarrow *Noun1 Prep Noun*. These transformations are recognized by removing suf-
 235 fixes, prefixes, and prepositions. The syntactic variants rely on the insertion of one or more words to the
 236 MWT without affecting the POS tags of the MWT subsets. The latter variants are identified by searching
 237 MWTs that share the same words. Table 2 presents inflectional, morpho-syntactic, and syntactic
 238 variations of Arabic MWTs.

Table 2
 Inflectional, morpho-syntactic, and syntactic MWTs variations

	MWT	Inflectional variant	Morpho-syntactic variant	Syntactic variant
Term	سياسة اقتصادية	السياسات الاقتصادية	سياسة اقتصاد	السياسة الاقتصادية الليبرالية
Transliteration	syAsp AqtSAdyp	AlsYAsAt AlAqtSAdyp	syAsp AqtSAd	AlsYAsp AlAqtSAdyp AllybrAlyp
Pattern	nn_fs jj_fs	det_nns_fp det_jj_fs	nnp nn_fs	det_nn_fs det_jj_fs det_jj_fs
Term	الجراحة التجميلية	الجراحات التجميلية	جراحة التجميل	مستشارة جراحة التجميل
Transliteration	AljrAHp Altjmylyp	AljrAHAt Altjmylyp	jrAHp Altjmyl	mst\$Arp jrAHp Altjmyl
Pattern	det_nn_fs det_jj_fs	det_nns_fp det_jj_fs	nns_fp nn_fs	nn_fs nn_fs det_nn
Term	تتمية زراعية	التنمية الزراعية	تتمية الزراعة	تتمية الزراعة الافريقية
Transliteration	tnmyp zrAEyp	Altnmyp AlzrAEyp	tnmyp AlzrAEp	tnmyp AlzrAEp AlAfryqyp
Pattern	nn_fs jj_fs	det_nn_fs det_jj_fs	nn_fs det_nn_fs	nn_fs det_nn_fs det_jj_fs
Term	تعديل دستوري	التعديلات الدستورية	التعديلات على الدستور	مشروع تعديلات دستورية
Transliteration	tEdyl dstwry	AltEdylAt Aldstwryp	AltEdylAt Ely Aldstwr	m\$rwE tEdylAt dstwryp
Pattern	nn jj	det_nns_fp det_jj_fs	det_nns_fp in det_nn	nn nns_fp jj_fs

239 The statistical filtering of MWT candidates relies on the idea that the more frequent MWTs in the corpus
 240 are more likely to be correct. (El Mahdaouy et al., 2013) introduced the *NLC-value* measure in order to
 241 consider contextual information and both termhood and unithood information. This measure incorporates
 242 the LLR measure Dunning (1993), which is a measure of unithood, to the *C/NC-value* (Frantzi et al., 2000),
 243 which is a measure of termhood. The *NLC-value* score for a given MWT, denoted $p = w_i \dots w_{i+k}$, is given by:

$$NLC\text{-value}(p) = 0.8 \cdot LC\text{-value}(p) + 0.2 \cdot N\text{-value}(p) \quad (1)$$

244 where the $LC\text{-value}(p)$ is given by:

$$LC\text{-value}(p) = \begin{cases} \log_2(|p|) \cdot FL(p) & \text{if } p \text{ is not nested} \\ \log_2(|p|) \cdot (FL(p) - \frac{1}{|T_p|} \sum_{b \in T_p} FL(b)) & \text{otherwise} \end{cases} \quad (2)$$

245 with $FL(p) = x_p \cdot \ln(2 + \min(LLR(p)))$. The $N\text{-value}(p)$ is obtained by:

$$N\text{-value}(p) = \sum_{w \in C_p} x_w^p \cdot \frac{x_w}{n} \quad (3)$$

246 In the above:

- 247 • $|p|$ denotes the length in words of the candidate MWT p ;
- 248 • x_p is the number of occurrences of p ;
- 249 • $T(p)$ denotes the set of longer candidate terms into which p appears;
- 250 • $|T(p)|$ is the cardinality of the set $T(p)$;
- 251 • C_p denotes the set of distinct context words of p , or simply the set of words that appear in the vicinity
- 252 of term p in texts;
- 253 • x_w^p corresponds to the number of times w occurs as context word of p ;
- 254 • x_w the number of candidate terms the word w appears with;
- 255 • n is the total number of terms considered.

256 At the indexing step, the $NLC\text{-value}$ threshold is varied between 0 and 30 and fixed experimentally to 5
 257 based on the best value of the MAP (Mean Average precision) in order to filter and select the best MWTs
 258 candidates.

259 4. Information Retrieval Models

260 We consider five standard models covering the main probabilistic families of IR models. These models
 261 are BM25, the Dirichlet language model (LM), PL2 from the divergence from randomness family and LGD
 262 and SPL from the information based family. We briefly review here their definition.

263 4.1. BM25

264 BM25 is one of the most popular probabilistic models. It was proposed by Robertson et al. (1994) and
 265 is based on a binary independence assumption. The weight of a query term is based on its within-document
 266 term frequency and query term frequency. The relevance score for a given query is defined by:

$$RSV_{BM25}(q, d) = \sum_{w \in q \cap d} \frac{(k_1 + 1) \cdot x_w^d}{K + x_w^d} \cdot \frac{(k_3 + 1) \cdot x_w^q}{k_3 + x_w^q} \cdot \log \frac{N - N_w + 0.5}{N_w + 0.5} \quad (4)$$

267 where $K = k_1 \cdot ((1 - b) + b \cdot \frac{d_l}{l_{avg}})$ is the parameter for the within document frequency normalization, k_1 is
 268 a positive tuning parameter that calibrates the document term frequency scaling, b is the parameter for
 269 normalizing the document length and k_3 is the parameter for tuning the query term frequency.

270 4.2. LM

271 The language modeling approach introduced by [Ponte and Croft \(1998\)](#) relies on ranking documents
 272 based on the probability of their language model generating a given query. For a query $q = w_1, w_2, \dots, w_n$
 273 and a document d , the scoring function estimates $P(q|d)$ the query likelihood given the document d . The
 274 retrieval status value is given by:

$$RSV_{LM}(q, d) = P(q|d) = \prod_{i=1}^n P(w_i|d) \quad (5)$$

275 Several smoothing methods ([Zhai and Lafferty, 2001](#)) have been proposed to overcome the zero probability
 276 problem. We rely in this study on the Dirichlet smoothing method, which has been shown to produce state-
 277 of-the-art results ([Zhai and Lafferty, 2001](#)). It is defined, combined with a standard multinomial distribution,
 278 by:

$$P(w|d) = \frac{1}{l_d + \mu} \left(\frac{x_w^d}{l_d} + \mu \frac{x_w^C}{|C|} \right) \quad (6)$$

279 where μ is the Dirichlet smoothing parameter.

280 4.3. Divergence from Randomness Models

281 The Divergence From Randomness models ([Amati and Van Rijsbergen, 2002](#)) are based on the idea that
 282 the more the within-document term-frequency is divergent from its frequency within the collection, the more
 283 the term is informative in the document d . We used the PL2 model of DFR framework where the RSV for
 284 a document d and a query q is given by:

$$RSV(d, q) = \sum_{w \in q \cap d} \frac{x_w^q}{x_{w_{max}}^q} \cdot \frac{1}{t_w^d + 1} \left(t_w^d \cdot \log_2 \left(\frac{t_w^d}{\lambda_w} \right) + (\lambda_w - t_w^d) \cdot \log_2(e) + 0.5 \cdot \log_2(2\pi \cdot t_w^d) \right) \quad (7)$$

285 where $t_w^d = x_w^d \cdot \log(1 + c \cdot \frac{l_d}{l_{avg}})$ is the normalized term frequency and $\lambda_w = \frac{N_w}{N}$ is a collection-dependent
 286 parameter of the term w . c is the term frequency normalization parameter.

287 4.3.1. Information Based Models

288 The family of information based models for IR has been recently introduced by [Clinchant and Gaussier](#)
 289 ([2010](#)). This family can be seen as sub-family of the DFR family inasmuch as it also relies on computing a
 290 deviation from randomness in the form of Shannon's information. However, it differs from standard DFR
 291 models as it relies on different probabilistic distributions that greatly simplify the DFR framework. The idea
 292 behind these models lies on ranking documents through the quantity of information brought by document
 293 terms on query words. The aim is to measure the behaviour of a term in a document and the collection.
 294 Thus, the difference in the behaviours of a word in the document and collection levels brings information
 295 on the significance of the word for the document. The models are based on the following RSV :

$$RSV(q, d) = \sum_{w \in q \cap d} -\frac{x_w^q}{l_q} \log P(X_w \geq t_w^d | \lambda_w) \quad (8)$$

296 where $t_w^d = x_w^d \cdot \log(1 + c \cdot \frac{l_d}{l_{avg}})$ is the normalized term frequency and $\lambda_w = \frac{N_w}{N}$ is a collection-dependent
 297 parameter of the term w . c is the term frequency normalization parameter.

298 We make use here of the two probability distributions introduced in ([Clinchant and Gaussier, 2010](#)):

299 1. The Log-Logistic model (LGD):

$$RSV_{LGD}(q, d) = \sum_{w \in q \cap d} -\frac{x_w^q}{l_q} \log \left(\frac{\lambda_w}{\lambda_w + t_w^d} \right) \quad (9)$$

2. The Smoothed Power Law model (SPL):

$$RSV_{SPL}(q, d) = \sum_{w \in q \cap d} -\frac{x_w^q}{l_q} \cdot \log\left(\frac{\lambda_w^{\frac{d}{l_w} + \frac{d}{l_w}} - \lambda_w}{1 - \lambda_w}\right) \quad (10)$$

5. Extensions for term dependencies

We now review several extensions of the above models aiming at integrating term dependencies, either explicitly captured through MWTs or implicitly through proximity.

5.1. MWT extensions

The extraction of MWTs leads to a new representation of queries and documents. Indeed, in addition to the representation based on single terms (standard bag-of-word representation), one can now also use a representation based on MWTs (*bag-of-mwt* representation). Hence, each query (and document) can be represented as a set of two "bags", one for single terms, one for multi-word terms: $q = \{q_{st}, q_{mwt}\}$. It is furthermore possible to define the same statistics (term frequency and document frequency) for MWTs as the ones for single terms.

From this, a direct integration of MWTs in IR models simply goes through the linear combination of two contributions, one based on single terms, the other on MWTs:

$$RSV(q, d) = (1 - \lambda) \cdot RSV(q_{st}, d_{st}) + \lambda \cdot RSV(q_{mwt}, d_{mwt}) \quad (11)$$

where λ controls the influence of each representation. Eq. 11 simply amounts to scoring queries and documents along two different representations and to combining linearly the scores obtained. This approach has been used in the past in (Shi and Nie, 2009; Metzler and Croft, 2005; Zhao et al., 2011) in order to incorporate phrases or proximity features (term dependencies) into existing IR models.

All the base models we have presented can be directly extended through Eq. 11, leading to the models $BM25_{MWT}$, LM_{MWT} , $PL2_{MWT}$, LGD_{MWT} , SPL_{MWT} .

5.2. Cross-term extensions

The CROSS TERM Retrieval (CRTER) model, proposed by Zhao et al. (2011), introduces a pseudo-term, namely, Cross Term (CT), to model term proximity for boosting retrieval performance. The idea behind this model is that an occurrence of a query term is assumed to have an impact towards its neighbouring terms, which gradually decreases with the increase of the distance to the place of occurrence. The Cross Term occurs when two query terms appear close to each other and their impact shape functions (kernel densities) have an intersection. To facilitate the incorporation of the latter terms into the ranking function, they defined (1) the within-document frequency, (2) the document frequency and (3) the within-query frequency of Cross Terms.

(1) The within-document frequency of CT $x_{p_{i,j}}^d$ in document d is the accumulation of $x_{p_{i,j}}^d$ values: $x_{p_{i,j}}^d = \sum_{k_1=1}^{x_{w_i}^d} \sum_{k_2=1}^{x_{w_j}^d} \text{Kernel}(\frac{1}{2}|pos_{k_1,i} - pos_{k_2,j}|)$ where *Kernel* is a density function.
 (2) The document frequency ($N_{p_{i,j}}$) relies on counting the number of documents where the CT appears ($x_{p_{i,j}}^d \neq 0$):

$$N_{p_{i,j}} = \sum_{d \in index} \mathbf{1}_{x_{p_{i,j}}^d \neq 0} \quad (12)$$

(3) The within-query CT frequency is obtained by assuming that query terms are adjacent to each other and considering all possible pairs formed of query terms:

$$x_{p_{i,j}}^q = \text{Kernel}(\frac{1}{2}) \cdot \min(x_{w_i}^q, x_{w_j}^q) \quad (13)$$

Several kernel functions have been used:

- 335 • The gaussian kernel: $\text{Kernel}(u) = \exp(\frac{-u^2}{2\sigma^2})$
- 336 • The triangle kernel: $\text{Kernel}(u) = (1 - \frac{u}{\sigma}) \cdot \mathbf{1}_{u \leq \sigma}$
- 337 • the circle kernel: $\text{Kernel}(u) = \sqrt{1 - \frac{u^2}{\sigma^2}} \cdot \mathbf{1}_{u \leq \sigma}$
- 338 • The cosine kernel: $\text{Kernel}(u) = \frac{1}{2}[1 + \cos(\frac{u\pi}{\sigma})] \cdot \mathbf{1}_{u \leq \sigma}$

339 where μ is the distance between two query terms and σ is a parameter to tune, which controls the spread
 340 of kernel curves. The final ranking scheme is defined by:

$$341 \text{CRTER}(d, q) = (1 - \lambda) \sum_{w \in q \cap d} \omega(x_w, d) + \lambda \sum_{1 \leq i \leq j \leq K} \omega(x_{p_{i,j}}, d) \quad (14)$$

341 where λ controls the influence of single terms and CTs. In the original model, ω is the BM25 (Robertson
 342 et al., 1994) scoring function. It can, however, be replaced by any scoring function, leading to extended
 343 versions of the base IR models introduced in Section 4.

344 5.3. Extensions specific to the LM model: MRF, PLM and QLM

345 We now review three extensions specific to the language model family.

346 5.3.1. MRF: Markov Random Field Model

347 The MRF model (Metzler and Croft, 2005) is a generalization of the language model approach where
 348 arbitrary features are incorporated as evidence into the scoring function, via the Markov Random Field
 349 framework. The model draws up three different levels of term dependencies: (1) the full independence (FI)
 350 model is based on single term occurrences and is equivalent to the baseline language model, (2) the sequential
 351 dependence (SD) consists of incorporating the scores of ordered phrases into the ranking function, and (3)
 352 the full dependence (FD) relies on unordered phrases. This model aims to construct a graph G from query
 353 terms and a document d . The different possible configurations allow different dependency assumptions. The
 354 score of each document is estimated using the joint distribution over the random variable in G , by the means
 355 of potential functions over clique configurations associated with different features (single terms, ordered and
 356 unordered phrases). The retrieval status value is given by:

$$357 \text{RSV}(d, q) = \sum_{c \in T} \lambda_T f_T(c) + \sum_{c \in O} \lambda_O f_O(c) + \sum_{c \in O \cup U} \lambda_U f_U(c) \quad (15)$$

where λ_T is the weight of single terms, λ_O is the weight of ordered phrases and λ_U is the weight of unordered
 phrases. T is defined as the set of 2-cliques involving a query term and a document d , O is the set of cliques
 containing the document node and two or more query terms that appear contiguously within the query,
 and U is the set of cliques containing the document node and two or more query terms appearing non-
 contiguously within the query. For all dependency features, the potential functions are estimated using the
 Dirichlet Language Model (Zhai and Lafferty, 2001). For a single term w , the potential function is defined
 by $f_T(c = (w; d)) = \log[(1 - \alpha_d) \frac{x_w^d}{l_d} + \alpha_d \frac{x_w^c}{|C|}]$ where α_d is a smoothing parameter. The potential functions $f_O()$
 and $f_U()$ are obtained by generalizing $f_T()$ for incorporating FD and FI features to the RSV. For ordered
 phrases, or sequential dependencies, the potential function is calculated using the following formula:

$$358 f_O(c = (w_i^q, \dots, w_{i+k}^q; d)) = \log[(1 - \alpha_d) \frac{x_{(w_i, \dots, w_{i+k})}^d}{l_d} + \alpha_d \frac{x_{(w_i, \dots, w_{i+k})}^c}{|C|}]$$

where $(w_i^q, \dots, w_{i+k}^q)$ is an ordered query phrase. $x_{(w_i, \dots, w_{i+k})}^d$ and $x_{(w_i, \dots, w_{i+k})}^c$ are the numbers of occurrences
 of $(w_i^q, \dots, w_{i+k}^q)$ in a document d and the collection C respectively. For the full dependency features, the
 potential function is defined by:

$$359 f_U(c = (w_i^q, \dots, w_j^q; d)) = \log[(1 - \alpha_d) \frac{x_{N(w_i, \dots, w_j)}^d}{l_d} + \alpha_d \frac{x_{N(w_i, \dots, w_j)}^c}{|C|}]$$

357 where (w_i^q, \dots, w_j^q) is an unordered query phrase. $x_{N(w_i, \dots, w_j)}^d$ and $x_{N(w_i, \dots, w_j)}^C$ are the number of times (w_i^q, \dots, w_j^q)
 358 appears ordered or unordered within a window of fixed length N in a document d and the collection C
 359 respectively.

360 5.3.2. PLM: Positional Language Model

361 The Positional Language Model (PLM) was introduced by [Lv and Zhai \(2009\)](#) with the aim to implement
 362 proximity and passage retrieval heuristics in a unified language model. The main idea is to estimate a
 363 language model for each position of a document, and rank a document based on the scores obtained at each
 364 position. A virtual document is created at each position, the count of any word being higher if it occurs
 365 closer to the position. More formally, one defines a language model at position i of document d as follows:

$$p(w|d, i) = \frac{c'(w, i)}{\sum_{w' \in V} c'(w', i)}$$

366 with $c'(w, i) = \sum_{j=1}^{d_i} x_w^{d, j} K(i, j)$ being the total propagated count of term w at position i from the occurrences
 367 of w in all the positions. $x_w^{d, j}$ is the count of term w at position j in document d , which is 0 if w does not
 368 occur at position j in d and 1 otherwise. $K(i, j)$ is the propagated count from position j to i and is estimated
 369 using kernel functions. For example, for the Gaussian kernel: $K(i, j) = \exp[\frac{-(i-j)^2}{2\sigma^2}]$.

370 One can then compute a score for the PLM at position i and the query, using a standard KL-divergence
 371 retrieval model:

$$S(q, d, i) = - \sum_{w \in V} p(w|q) \log \frac{p(w|q)}{p(w|d, i)} \quad (16)$$

372 The final score for the complete document is then obtained through best position, multi-position or multi- σ
 373 strategies. In this study, we rely on the best position strategy to estimate the PLM of a given query term.
 374 We selected the Gaussian kernel and the parameter σ is fixed between 25 and 300 through cross-validation
 375 ([Lv and Zhai, 2009](#)).

376 5.3.3. QLM: Quantum Language Model

377 The Quantum Language Model (QLM) has been proposed by [Sordoni et al. \(2013\)](#) to avoid the weight-
 378 normalization problem caused by counting term dependencies and single terms. The basic idea of QLM is
 379 that term dependencies are counted at the estimation phase as a superposition of component terms that
 380 constitute a compound dependency. Thus, single terms are represented as a set of projectors (quantum events
 381 that represent the occurrence of a query terms) on the standard basis: $X = \{|e_i\rangle\langle e_i|\}_{i=1}^n$, i.e $|e_i\rangle = (\delta_{1i}, \dots, \delta_{ni})^T$,
 382 called ket vectors, and $\langle e_i| = (\delta_{1i}, \dots, \delta_{ni})$, called bra vectors, where $\delta_{ij} = 1$ iff $i = j$. Single terms are
 383 mapped to quantum events by $X_w = m(x_w) = |e_{x_w}\rangle\langle e_{x_w}|$ which consists of associating the occurrence of w
 384 to a dyad $|e_{x_w}\rangle\langle e_{x_w}|$. For a term dependency $k = \{x_{w_1}, x_{w_2}, \dots, x_{w_k}\}$, the mapping to projector is defined by
 385 $X_k = m(\{x_{w_1}, x_{w_2}, \dots, x_{w_k}\}) = |k\rangle\langle k|$ such that $|k\rangle = \sum_{i=1}^k \sigma_i |e_{x_{w_i}}\rangle$. The dyad $|k\rangle\langle k|$ is a superposition event of
 386 observing k ; σ_i are real coefficients and $\sum_{i=1}^k \sigma_i^2 = 1$ in order to ensure proper normalization of $|k\rangle$. The
 387 event $|k\rangle\langle k|$ adds a fractional occurrence to the event of its component terms $|e_{x_w}\rangle\langle e_{x_w}|$.

388 As a first step, the model builds the sequence of projectors by adding both term dependencies and all
 389 their component terms. In the second step, the density matrices for documents, query, and the collection
 390 are estimated using Maximum Likelihood. Let $X_d = \{X_1, \dots, X_M\}$ be the set of observed projectors of single
 391 terms and term dependencies for a given document d . The density likelihood is given by:

$$\mathcal{L}_{X_d}(\rho) = \prod_{i=1}^M \text{tr}(\rho X_i) \quad (17)$$

where $\text{tr}(\rho X_i)$ is the probability of observing X_i . For a given number of iterations, the maximization of
 the density ρ is approximated by the iterative $R\rho R$ algorithm ([Lvovsky, 2004](#)) which consists of solving the

following problem:

$$\begin{cases} \underset{\rho}{\text{maximize}} \log \mathcal{L}_{\mathcal{X}_d}(\rho) \\ R(\rho) = \sum_{i=1}^M \frac{1}{\text{tr}(\rho X_i)} X_i \\ \hat{\rho}(k+1) = \frac{1}{Z} R(\hat{\rho}(k)) \hat{\rho}(k) R(\hat{\rho}(k)) \text{ where } Z = \text{tr}(R(\hat{\rho}(k)) \hat{\rho}(k) R(\hat{\rho}(k))) \end{cases}$$

where the density matrix $R(\rho)$ serves to discover the set of $\hat{\rho}$ that maximizes the likelihood and Z is a normalization factor that guarantees the constraint of the unitary trace. The convergence is ensured through density damping when the likelihood decreases. For example, if the likelihood decreases at the iteration $k+1$, the damped update of $\hat{\rho}(k+1)$ is defined by $\tilde{\rho}(k+1) = (1-\gamma)\hat{\rho}(k) + \gamma\hat{\rho}(k+1)$ where $\gamma \in [0, 1)$ is a parameter that controls the amount of damping. Moreover, the iterative process starts with the initial matrices of a given document $\rho(0)_d = \text{diag}(\frac{x_{w_{1,1}}^d}{l_d}, \frac{x_{w_{2,2}}^d}{l_d}, \dots, \frac{x_{w_{l_q, l_q}}^d}{l_d}, \frac{l_d - \sum_{i=1}^{l_q} x_{w_{i,i}}^d}{l_d})$, query $\rho(0)_q = \text{diag}(\frac{x_{w_{1,1}}^q}{l_q}, \frac{x_{w_{2,2}}^q}{l_q}, \dots, \frac{x_{w_{l_q, l_q}}^q}{l_q}, 0)$, and the collection $\rho(0)_C = \text{diag}(\frac{x_{w_{1,1}}^C}{N}, \frac{x_{w_{2,2}}^C}{|C|}, \dots, \frac{x_{w_{l_q, l_q}}^C}{|C|}, \frac{|C| - \sum_{i=1}^{l_q} x_{w_{i,i}}^C}{|C|})$. The dimension of the diagonal matrices is $l_q + 1$, while the additional dimension stores probability mass for the other terms in the vocabulary.

After density maximization, the document density is smoothed to avoid the zero-probability problem using the formula: $\rho_d = (1 - \alpha_d)\hat{\rho}_d + \alpha_d\hat{\rho}_C$, $\alpha_d = \frac{\mu}{\mu + M}$ being a smoothing parameter. The scoring function is obtained by the negative query-to-document Von-Neumann divergence given by the formula:

$$\begin{aligned} RS V(q, D) &= -\Delta_{VN}(\rho_q \| \rho_d) \\ &\stackrel{\text{rank}}{=} \text{tr}(\rho_q \log \rho_d) \\ &\stackrel{\text{rank}}{=} \sum_i \lambda_{qi} \sum_j \log \lambda_{dj} \langle q_i | d_j \rangle^2 \end{aligned} \tag{18}$$

where $\rho_q = \sum_i \lambda_{qi} |q_i\rangle \langle q_i|$ and $\rho_d = \sum_j \lambda_{dj} |d_j\rangle \langle d_j|$ are the eigendecompositions of the density matrices ρ_d and ρ_q respectively.

5.4. DFR term dependence model

The DFR term dependence model (Peng et al., 2007) consists of incorporating term dependencies into the Divergence From Randomness (DFR) framework (Amati and Van Rijsbergen, 2002). The proposed model assigns scores to pairs of query terms, as well as single terms. The general framework of the proposed model is as follows:

$$RS V(d, q) = \lambda_1 \cdot \sum_{w \in q} \text{score}(w, d) + \lambda_2 \cdot \sum_{p \in q_2} \text{score}(p, d) \tag{19}$$

where $\text{score}(w, d)$ is the score assigned to the query term w for the document d , p corresponds to a dependency feature that consists of a pair of query terms, $\text{score}(p, d)$ is the score assigned to p for the document d , and q_2 is the set of dependency features of the query q . The $\text{score}(w, d)$ can be estimated by any DFR weighting model. In this paper, we rely on the PL2 document weighting model (Amati and Van Rijsbergen, 2002). For full independence (FI), the term dependencies are ignored, i.e. $\lambda_1 = 1$ and $\lambda_2 = 0$. Concerning the sequential and the full dependencies, the weighting parameters are set as $\lambda_1 = 1$ and $\lambda_2 = 1$. The proximity-based randomness model is used to compute the weight $\text{score}(p, d)$ without considering the collection frequency of the pair of query terms. In particular, it is based on the binomial Randomness model given by:

$$\begin{aligned} \text{score}(d, p) &= \frac{1}{t_p^d + 1} \cdot (-\log_2(l_d - 1)! + \log_2 t_p^d!) \\ &\quad + \log_2(l_d - 1 - t_p^d)! - t_p^d \log_2(p_p)! \\ &\quad - (-l_d - 1 - t_p^d) \log_2(p'_p) \end{aligned} \tag{20}$$

where $p_p = \frac{1}{l_d - 1}$ and $p'_p = 1 - p_p$, and t_p^d is the normalized frequency of the pair of query terms p using Normalization 2 (Amati and Van Rijsbergen, 2002): $t_p^d = x_p^d \cdot \log_2(1 + c \frac{l_{\text{avg}} - 1}{l_d - 1})$. In this normalization, c is a

parameter usually tuned by cross-validation and x_p^d is the number of times a pair of query terms appear in a document d .

By considering either ordered or unordered pairs of query terms in addition to single terms, one ends up with the same term dependencies used in the MRF model (FI, SD, and FD).

5.5. Summary

The above extensions show that there are several ways to deal with term dependencies in IR in general and Arabic IR in particular. Starting with the three main models (or model families) introduced in Section 4, we end up with the following models that are able to take into account term dependencies:

1. In the language model family: LM_MWT, LM_CT, MRF, PLM, QLM
2. In the DFR family: LGD_MWT, LGD_CT, SPL_MWT, SPL_CT, PL2_MWT, PL2_CT, DFR_TD
3. For BM25: BM25_MWT, BM25_CT

where $_MWT$ denotes the multi-word term extension (Section 5.1), $_CT$ the cross-term extension (Section 5.2) and $_TD$ the term dependence extension of the DFR model (Section 5.4).

6. Compound heuristic condition

In the vein of the studies on IR heuristic constraints ((Fang et al., 2004; Clinchant and Gaussier, 2011; Fang and Zhai, 2014) for example), we introduce a formal condition that IR models should satisfy to deal adequately with term dependencies. This condition simply states that if a query consists of two dependent terms, in the form of a compound, then, *mutatis mutandis*, a document that contains more occurrences of the compound should receive a higher score than a document with less occurrences.

Condition 1 (compound condition) *Let $q = \{w_1, w_2\}$ be a query consisting of one MWT $p = \{w_1, w_2\}$, and d_1 and d_2 two documents of equal length such that $x_{w_1}^{d_1} = x_{w_1}^{d_2}$, $x_{w_2}^{d_1} = x_{w_2}^{d_2}$. If $x_p^{d_1} > x_p^{d_2}$, then $RSV(q, d_1) > RSV(q, d_2)$.*

It is easy to see that the MWT and CT extensions of the models considered above satisfy the compound condition. This is due to the fact these extensions are based on linear combinations of single term and dependent term contributions, and that all the standard IR models considered satisfy the term frequency (TF) condition ((Fang et al., 2004)). The same reasoning and result hold for DFR_TD, as well as for MRF. For QLM, representing a dependency feature as a superposition event adds a fractional occurrence to the dependency term components, so that the compound condition is also satisfied.

The situation is slightly more complex for PLM. In this model, and for all the strategies to combine the positional language models, the retrieval status value of a document will increase with the proximity of the query words. Inasmuch as term dependency usually entails term proximity, PLM has a tendency to satisfy the compound condition. This said, this behavior is not guaranteed; it is indeed possible that the two words constituting a compound in one document are separated by different words (as adjectives inserted in a noun-noun sequence for example) and are finally away from each other, whereas they are close to each other in another document without forming a compound (if they are just separated by a comma and belong to two different propositions, for example). One can construct such instances so that the difference in the number of occurrences does not overcome the proximity factor.

As a summary, all the models we are considering but PLM satisfy the compound condition. From this perspective, they are, again with the exception of PLM, valid models to deal with term dependencies in IR. As we will see in the next section, PLM is the worst model compared to the other model relying on term dependencies.

461 **7. Experiments**

462 *7.1. Test Collections and Evaluation*

463 In order to assess the different models presented above, we performed experiments using the TREC-
 464 2001 and TREC-2002 topics and relevance judgements on the Arabic Newswire LDC Catalog³. The corpus
 465 consists of 383,872 documents from the AFP (France Press Agency) Arabic Newswire, containing 76 million
 466 tokens for 666,094 unique words. These documents are newspaper articles covering the period from May
 467 1994 until December 2000. The set of topic TREC-2001 (25 topics) and TREC-2002 (50 topics) are merged
 468 in TREC-2002/2001 to have a sufficient number of topics (75 topics) to perform 5-folds cross validation.
 The dataset is described in Table 3.

Table 3
 Dataset description

Corpus	Test sets	Query Ids	Query fields	#Documents
LDC2001T55	TREC 2001	1–25	title, title-description	383872
	TREC 2002	26–75	title, title-description	383872
	TREC 2002/2001	1–75	title, title-description	383872

469 The experiments are accomplished by extending Terrier⁴ IR Platform v3.5. Whenever possible, we used
 470 the existing Terrier implementations of the aforementioned models; we nevertheless had to implement the
 471 PLM, BM25_CT and QLM models, as well as the MWT extensions for all models. The extensions are
 472 also compared with several non bag-of-words models using the Mean Average Precision (MAP) and the
 473 precision at 10 documents (P10). The best performances on the MAP and the P10 values are shown by
 474 bold and bold-italic respectively. Moreover, we performed a significance paired t-test and attached \uparrow to the
 475 performance number in the tables when the test passes at 90%. Table 4 summarizes the IR models, the set
 476 of parameters, and the values that are used for cross validation.

Table 4
 Cross validation parameter values

Model	Parameter	Values
LGD and its extensions SPL and its extensions	c	0.1, 0.5, 1.0, 1.5, 2.0, 2.5, 3.0, 3.5, 4.0
PL2/DFR_TD		4.5, 5.0, 6.0, 7.0, 8.0, 9.0, 10.0, 20
LM/MRF QLM/PLM QLM_MWT	μ	10, 25, 50, 75, 100, 200, 300, 400, 500, 600, 700, 800, 900, 1000, 1500, 2000, 2500, 3000, 4000, 5000
DFR_TD MWT extensions CT extensions	λ	0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9
BM25 and its extensions	b	0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.75, 0.8, 0.9, 1.0 1.25, 1.5, 1.75, 2.0, 2.25, 2.5, 2.75, 3.0

477

478 *7.2. Results*

479 Firstly, in order to evaluate the effectiveness of the state-of-the-art IR models for Arabic content retrieval,
 480 experiments are performed using the heavy stemming and light stemming approaches for text preprocessing.
 481 The main goal of these experiments is to answer the question: *Which standard IR models and stemming*
 482 *approaches are appropriate for Arabic IR?*

³<http://catalog ldc.upenn.edu/LDC2001T55>

⁴www.terrier.org

Table 5

A summary of the results for bag-of-words IR models using Farasa, light and heavy stemming approaches on title queries. For statistical significance, *f* = better than Farasa stemming, *l* = better than light stemming, and *h* = better than heavy stemming

		Farasa Stemming			Light Stemming			Heavy Stemming		
Model	Metric/TERC	2002/2001	2001	2002	2002/2001	2001	2002	2002/2001	2001	2002
LGD	MAP	30.09 ^{l,h}	33.94 ^{l,h}	28.17 ^{l,h}	27.24 ^h	32.99 ^h	23.79	23.79	26.40	22.49
	P10	44.13	54.80	38.80	37.33	49.20	37.87	37.87	52.00	30.80
SPL	MAP	30.48 ^{l,h}	34.64 ^{l,h}	28.39 ^{l,h}	27.13 ^h	32.90 ^h	24.24	24.92	27.25	23.74
	P10	47.07	62.40	39.40	42.53	58.72	34.40	41.87	54.80	34.20
PL2	MAP	30.58 ^{l,h}	34.60 ^{l,h}	28.57 ^{l,h}	27.23 ^h	32.54 ^h	24.59 ^h	24.34	27.47	22.78
	P10	47.33	62.00	40.00	41.33	55.60	35.80	41.47	54.40	35.00
BM25	MAP	31.50^{l,h}	35.84^{l,h}	29.32^{l,h}	27.65^h	33.22^h	24.86 ^h	23.67	26.63	22.19
	P10	47.07	60.80	40.20	40.13	52.80	33.80	37.20	50.80	30.40
LM	MAP	29.67 ^{l,h}	32.73 ^{l,h}	28.14 ^{l,h}	27.03 ^h	31.25 ^h	24.95^h	23.68	26.34	22.35
	P10	44.93	54.40	40.20	42.40	52.40	37.40	39.87	52.80	33.40

Table 6

A summary of the results for bag-of-words IR models using Farasa, light and heavy stemming approaches on title-description queries. For statistical significance, *f* = better than Farasa stemming, *l* = better than light stemming, and *h* = better than heavy stemming

		Farasa Stemming			Light Stemming			Heavy Stemming		
Model	Metric/TERC	2002/2001	2001	2002	2002/2001	2001	2002	2002/2001	2001	2002
LGD	MAP	32.42 ^{l,h}	35.92 ^{l,h}	31.77 ^{l,h}	28.94 ^h	31.87 ^h	27.47^h	24.97	27.34	23.78
	P10	47.33	58.40	45.00	44.20	53.60	37.40	41.07	54.00	34.60
SPL	MAP	33.51^{l,h}	36.30^{l,h}	32.12^{l,h}	28.72 ^h	32.32 ^h	26.93 ^h	25.28	26.45	24.70
	P10	50.67	63.60	44.20	44.80	62.40	36.00	45.73	55.20	41.00
PL2	MAP	33.22 ^{l,h}	36.10 ^{l,h}	31.77 ^{l,h}	28.95^h	32.91 ^h	26.98 ^h	25.86	28.37	24.61
	P10	50.53	61.60	45.00	42.80	57.60	35.40	44.13	56.80	37.80
BM25	MAP	33.42 ^{l,h}	36.32 ^{l,h}	31.96 ^{l,h}	28.93 ^h	33.21^h	26.78 ^h	25.17	28.14	23.68
	P10	49.60	60.40	44.20	42.93	58.80	35.00	44.40	56.40	38.40
LM	MAP	31.15 ^{l,h}	33.11 ^{l,h}	30.18 ^{l,h}	27.85 ^h	30.22 ^h	26.66 ^h	25.22	27.56	24.05
	P10	46.93	56.00	42.40	43.07	52.80	38.20	43.87	55.60	38.00

483 [Table 5](#) and [Table 6](#) summarize the obtained results for title and title-description queries respectively.
484 These results show that the Farasa stemming approach outperforms significantly the classical stemming ap-
485 proaches for Arabic IR, which is explained by the high accuracy of word segmentation with Farasa ([Darwish](#)
486 [and Mubarak, 2016](#)). In line with previous studies, the light stemming approach improves significantly the
487 heavy stemming approach. The low performance of the heavy stemming approach is explained by the fact
488 that the root-based stemmer conflates words with different meanings into the same root. For title queries,
489 small improvements are obtained using Farasa and light stemmers for the BM25 model in comparison to the
490 other models. The best P10 values are obtained by the SPL for both TREC-2002/2001 and TREC-2001.
491 For the root-based approach, the models that rank documents based on the informative content of a query
492 term for a given document are more effective. Hence, the best performance is achieved by the SPL and the
493 PL2 models. For title-description queries, a slightly better performance is obtained by the SPL model using
494 the Farasa Stemmer. The comparison, furthermore, shows that the SPL, PL2 and BM25 models achieve a
495 better performance than the LGD and the LM models using the three stemming approaches.

496 Secondly, we compare the effectiveness of proximity-based models (using the cross-term extensions as well
497 as the specific extensions for DFR, DFR_TD, and the language model, PLM, MRF, and QLM) and their
498 bag-of-words baselines for Arabic IR using the three stemming approaches. [Table 7](#) and [Table 8](#) illustrate the

499 results obtained for title and title-description queries respectively. For the cross-term extensions, a Gaussian
500 kernel is used; the parameter σ varies in 2, 5, 10, 15, 20, 25, 50, 75, 100 and is selected through cross-validation.
501 For this aim, we investigate various term dependence models including the DFR dependence model (noted
502 DFR_TD in this paper), the MRF, the PLM, and the QLM from the language modeling approach, as well
503 as the CRTER model for incorporating term dependencies to the BM25 model. Moreover, we incorporate
504 CTs to the family of information based models as well as the PL2 model and the language model. Hence,
505 we applied different settings of the parameter $\sigma = 2, 5, 10, 15, 20, 25, 50, 75, 100$ and selected the best value
506 for each model and the Gaussian kernel function.

Table 7

A summary of the comparison results for proximity-based models against their bag-of-words models using light and heavy stemmers on title queries

		Farasa Stemming			Light Stemming			Heavy Stemming		
Model	Metric/TREC	2002/2001	2001	2002	2002/2001	2001	2002	2002/2001	2001	2002
LGD	MAP	30.09	33.94	28.17	27.24	32.99	23.79	23.79	26.40	22.49
	P10	44.13	54.80	38.80	37.33	49.20	37.87	37.87	52.00	30.80
LGD_CT	MAP	32.47 ↑	36.04 ↑	30.68 ↑	28.44 ↑	34.12 ↑	25.60 ↑	24.49	26.80	23.33
	P10	45.60	56.00	40.40	42.67	55.20	36.40	39.73	54.00	32.60
SPL	MAP	30.48	34.64	28.39	27.13	32.90	24.24	24.92	27.25	23.74
	P10	47.07	62.40	39.40	42.53	58.72	34.40	41.87	54.80	34.20
SPL_CT	MAP	32.05 ↑	36.47 ↑	29.84 ↑	29.59 ↑	34.59 ↑	27.09 ↑	25.61	28.32 ↑	24.26
	P10	48.13	64.40	40.00	45.60	58.40	39.20	42.93	56.80	36.00
PL2	MAP	30.58	34.60	28.57	27.23	32.54	24.59	24.34	27.47	22.78
	P10	47.33	62.00	40.00	41.33	55.60	35.80	41.47	54.40	35.00
PL2_CT	MAP	32.41 ↑	36.69 ↑	30.27 ↑	29.67 ↑	35.39 ↑	26.81 ↑	25.44 ↑	28.50 ↑	23.85 ↑
	P10	48.27	63.20	40.80	45.07	57.20	39.00	43.50	56.00	37.25
DFR_TD	MAP	32.00↑	36.20↑	29.90↑	29.59↑	35.21↑	26.78↑	25.36↑	28.33↑	23.87 ↑
	P10	48.40	63.60	40.80	44.27	56.00	38.40	43.07	56.80	36.20
BM25	MAP	31.50	35.84	29.32	27.65	33.22	24.86	23.67	26.63	22.19
	P10	47.07	60.80	40.20	40.13	52.80	33.80	37.20	50.80	30.40
CRTER	MAP	33.31 ↑	37.96 ↑	30.99 ↑	29.61 ↑	35.41 ↑	26.71 ↑	24.68 ↑	27.24 ↑	23.40 ↑
	P10	48.93	63.20	41.80	43.65	53.35	38.80	40.93	55.20	33.80
LM	MAP	29.67	32.73	28.14	27.05	31.25	24.95	23.68	26.34	22.35
	P10	44.93	54.40	40.20	42.40	52.40	37.40	39.87	52.80	33.40
LM_CT	MAP	31.90 ↑	34.45 ↑	30.63 ↑	28.50 ↑	33.32↑	26.10 ↑	25.36 ↑	28.40 ↑	23.84 ↑
	P10	46.00	56.00	41.00	43.33	52.40	38.80	42.02	56.05	35.00
PLM	MAP	29.97	32.96	28.47	27.38	32.02	25.07	24.04	27.40	22.36
	P10	45.07	54.80	40.20	42.13	50.80	37.80	40.27	53.00	33.90
MRF	MAP	31.38↑	33.95↑	30.10↑	28.02	32.50	25.78	25.24↑	28.01↑	23.86↑
	P10	45.87	56.00	40.80	42.67	52.40	37.80	41.60	55.60	34.60
QLM	MAP	31.50↑	34.03↑	30.23↑	28.29↑	34.52 ↑	25.18	24.66↑	28.01↑	23.15↑
	P10	46.00	56.00	41.00	41.47	54.80	35.60	41.87	54.81	35.40

507 The obtained results for title queries show that proximity-based models improve significantly the accuracy
508 of Arabic content retrieval for the three stemming algorithms for all test collections. Thus, proximity
509 among query terms is very useful for boosting the retrieval performance of Arabic documents. Moreover,
510 incorporating CT weights to the evaluated IR models leads to significant improvement over their baselines.
511 For Farasa stemmer, a better performance is obtained by incorporating cross terms into the BM25, the PL2,
512 the SPL, the LGD models. In the context of the language modeling approach, the LM_CT achieves slightly
513 better performance than the PLM, the MRF, and the QLM models on all test collection, while QLM shows
514 a comparable P10 performance to the LM_CT. For the light stemmer, the best results are obtained by
515 the DFR_TD, PL2_CT, the CRTER, the SPL_CT, and the PL2_CT models on TREC-2002/2001 and
516 TREC-2002. Besides, the PL2_CT, DFR_TD, and the BM25_CT models achieve better performance on
517 TREC-2001. In the context of the language modeling approach, the incorporation of CTs (LM_CT) leads to
518 better performance than the QLM, the PLM, and the MRF models on TREC-2002/2001 and TREC-2002,

519 while QLM performs better on TREC-2001. For the heavy stemming approach, further improvements are
520 achieved by DFR_TD, SPL_CT, MRF, and LM_CT on TREC-2002/2001 and TREC-2001. Additionally,
521 SPL_CT achieves the best performance on TREC-2002. Further, combining CT scores and single term
522 scores using SPL and PL2 models leads to a better performance than the BM25_CT model.

Table 8

A summary of the comparison results for proximity-based models against their bag-of-words models using light and heavy stemmers on title-description queries

Model	Metric/TREC	Farasa Stemming			Light Stemming			Heavy Stemming		
		2002/2001	2001	2002	2002/2001	2001	2002	2002/2001	2001	2002
LGD	MAP	32.42	35.92	31.77	28.94	31.87	27.47	24.97	27.34	23.78
	P10	47.33	58.40	45.00	44.20	53.60	37.40	41.07	54.00	34.60
LGD_CT	MAP	34.23 ↑	38.34 ↑	32.18 ↑	29.92	34.06 ↑	27.85	27.48 ↑	30.51 ↑	25.96 ↑
	P10	48.27	60.80	42.00	44.00	59.20	36.40	43.73	56.40	37.40
SPL	MAP	33.51	36.30	32.12	28.72	32.32	26.93	25.28	26.45	24.70
	P10	50.67	63.60	44.20	44.80	62.40	36.00	45.73	55.20	41.00
SPL_CT	MAP	35.28 ↑	39.12 ↑	33.36 ↑	31.66 ↑	35.23 ↑	29.87 ↑	27.82 ↑	29.70 ↑	26.87 ↑
	P10	50.93	63.60	44.60	47.07	62.40	39.40	45.60	56.40	40.20
PL2	MAP	33.22	36.10	31.77	28.95	32.91	26.98	25.86	28.37	24.61
	P10	50.53	61.60	45.00	42.80	57.60	35.40	44.13	56.80	37.80
PL2_CT	MAP	34.99 ↑	38.40 ↑	33.29 ↑	31.24 ↑	34.76↑	29.48 ↑	28.29 ↑	31.31 ↑	26.79↑
	P10	51.20	63.20	45.20	46.67	62.40	38.80	46.13	58.40	40.00
DFR_TD	MAP	34.24	37.45	32.63	30.90↑	35.20 ↑	28.75↑	28.13↑	31.08↑	26.65 ↑
	P10	50.80	62.00	45.20	47.07	60.80	40.20	46.00	58.00	40.00
BM25	MAP	33.42	36.32	31.96	28.93	33.21	26.78	25.17	28.14	23.68
	P10	49.60	60.40	44.20	42.93	58.80	35.00	44.40	56.40	38.40
CRTER	MAP	35.02 ↑	37.96 ↑	33.54 ↑	31.56 ↑	35.55 ↑	29.56 ↑	27.12 ↑	31.01 ↑	25.17 ↑
	P10	50.93	61.20	45.80	47.73	62.00	40.60	43.47	56.80	36.80
LM	MAP	31.15	33.11	30.18	27.85	30.22	26.66	25.22	27.56	24.05
	P10	46.93	56.00	42.40	43.07	52.80	38.20	43.87	55.60	38.00
LM_CT	MAP	32.85 ↑	34.91 ↑	31.82 ↑	29.33↑	32.09↑	27.95	27.41 ↑	30.13↑	26.05 ↑
	P10	47.47	57.60	42.40	45.07	58.00	38.60	44.53	56.40	38.60
PLM	MAP	31.66	33.65	30.66	28.49	31.03	27.23	26.05	28.93	24.61
	P10	46.93	56.80	42.00	43.73	54.80	38.20	44.27	57.20	37.80
MRF	MAP	32.04	33.95	31.09	29.37↑	32.72↑	27.69	27.27↑	30.15 ↑	25.83↑
	P10	47.07	56.80	42.20	44.13	56.80	37.80	44.53	56.80	38.40
QLM	MAP	32.54↑	34.58↑	31.52↑	29.44 ↑	32.22 ↑	28.05↑	27.22↑	29.92↑	25.88↑
	P10	47.07	56.80	42.20	44.67	55.60	39.20	44.40	56.40	38.40

523 In accordance with the obtained results for title queries, proximity-based models, using title-description
524 queries, improve significantly the accuracy of Arabic IR for the three stemming algorithms for all test collec-
525 tions. Additionally, incorporating CT weights to the evaluated IR models leads to significant improvement
526 over their baselines. For all stemming approaches, incorporating cross terms into the SPL, PL2, and BM25
527 models yields to a better performance than the other IR models. Finally, if the proximity-based models
528 significantly improve their corresponding bag-of-words models for most test sets using the three stemming
529 approaches and both title and title-description queries, their performance is higher with the Farasa stemmer
530 than with the light and the heavy stemmers.

531 Thirdly, we investigate the incorporation of Arabic MWTs into the families of models we have considered.
532 For stemming purposes, we selected the Farasa and light stemmers since they outperform the heavy stemmer
533 either for all models. For MWTs candidates filtering, the *NLC-value* threshold is varied between 0 and 30
534 and fixed experimentally to 5 based on the best value of the MAP. At the query level, MWT candidates are
535 extracted based on the linguistic filter only; for example, for the first title query $q_1 = \{ \text{فنون العرض و المؤسسات} \}$
536 العربي الاسلامية في العالم العربي}, the extracted MWTs are "فنون العرض" (Performing arts), "المؤسسات الاسلامية"
537 العربي), and "العالم العربي" (the Arab world). Table 9 and Table 10 show the obtained results for MWTs

extensions and their bag-of-words baselines using title and title-description queries respectively.

Table 9

A summary of the comparison results of MWT extensions and their bag-of words models using the Farasa and light stemming approaches on title queries

TREC	Farasa Stemming						Light Stemming					
	2002/2001		2001		2002		2002/2001		2001		2002	
Model/Metric	MAP	P10	MAP	P10	MAP	P10	MAP	P10	MAP	P10	MAP	P10
LGD	30.09	44.13	33.94	54.80	28.17	38.80	27.24	37.33	32.99	49.20	24.37	31.40
LGD_MWT	33.09 ↑	46.53	36.58 ↑	56.80	31.31 ↑	42.20	28.90 ↑	41.73	34.63 ↑	52.80	26.03 ↑	36.20
SPL	30.48	47.07	34.64	62.40	28.39	39.40	27.13	42.53	32.90	58.72	24.24	34.40
SPL_MWT	32.34 ↑	48.13	36.80 ↑	64.80	30.11 ↑	39.80	29.96 ↑	45.20	36.18 ↑	60.80	28.85 ↑	37.40
PL2	30.58	47.33	34.60	62.00	28.57	40.00	27.23	41.33	32.54	55.60	24.59	34.40
PL2_MWT	32.74 ↑	48.67	37.11 ↑	63.60	30.55 ↑	41.20	29.56 ↑	43.60	35.50 ↑	56.80	26.59 ↑	37.00
BM25	31.50	47.07	35.84	60.80	29.32	40.20	27.65	40.13	33.22	52.80	24.86	33.80
BM25_MWT	33.73 ↑	49.33	38.58 ↑	63.60	31.31 ↑	42.20	30.50 ↑	44.27	36.85 ↑	57.20	27.32 ↑	37.40
LM	29.67	44.93	32.73	54.40	28.14	40.20	27.05	42.40	31.25	52.40	24.95	37.40
LM_MWT	31.63 ↑	46.00	34.14 ↑	56.00	30.38 ↑	41.00	28.12 ↑	41.27	33.18 ↑	54.50	25.59	36.20

538

Table 10

A summary of the comparison results of MWT extensions and their bag-of words models using the Farasa and light stemming approaches on title-description queries

TREC	Farasa Stemming						Light Stemming					
	2002/2001		2001		2002		2002/2001		2001		2002	
Model/Metric	MAP	P10	MAP	P10	MAP	P10	MAP	P10	MAP	P10	MAP	P10
LGD	32.42	47.33	35.92	58.40	31.77	45.00	28.94	44.20	31.87	53.60	27.47	37.40
LGD_MWT	34.96 ↑	48.67	39.23 ↑	61.20	32.82 ↑	42.40	30.56 ↑	46.67	34.81 ↑	63.60	28.43	38.20
SPL	33.51	50.67	36.30	63.60	32.12	44.20	28.72	44.80	32.32	62.40	26.93	36.00
SPL_MWT	35.88 ↑	51.87	39.92 ↑	65.60	33.86 ↑	45.00	31.83 ↑	49.33	36.14 ↑	65.60	29.68 ↑	41.20
PL2	33.22	50.53	36.10	61.60	31.77	45.00	28.95	42.80	32.91	57.60	26.98	35.40
PL2_MWT	35.55 ↑	51.20	39.25 ↑	62.80	33.70 ↑	45.40	31.50 ↑	47.87	35.85 ↑	61.60	29.32 ↑	41.00
BM25	33.42	49.60	36.32	60.40	31.96	44.20	28.93	42.93	33.21	58.80	26.78	35.00
BM25_MWT	35.56 ↑	51.47	38.60 ↑	61.20	34.04 ↑	46.60	31.86 ↑	48.93	36.94 ↑	65.20	29.32 ↑	40.80
LM	31.15	46.93	33.11	56.00	30.18	42.40	27.85	43.07	30.22	52.80	26.66	38.20
LM_MWT	33.32 ↑	47.87	35.19 ↑	58.00	32.38 ↑	42.80	29.62 ↑	46.00	32.90 ↑	58.00	27.98 ↑	40.00

539 According to these results, incorporating MWTs to IR models significantly improves the performance
540 of Arabic content retrieval. Conforming to the incorporation of cross terms into the evaluated IR models,
541 the BM25_MWT, SPL_MWT and PL2_MWT models achieve better performance on most test sets than
542 the LGD_MWT, and the LM_MWT models. Although the MWT extensions significantly improve their
543 corresponding bag-of-words models for most test sets both stemming approaches and both query types (title
544 and title-description queries), their performance is higher with the Farasa stemmer than with the light
545 stemmer.

546 Lastly, in order to compare the two approaches for ranking documents based on term dependencies,
547 we evaluate the proximity-based models and MWT extensions for Arabic IR. Thus, we select the obtained
548 results for Farasa and light stemmers for both proximity IR models and MWT extensions. [Table 11](#) and
549 [Table 12](#) summarize the obtained results for title and title-description queries respectively.

Table 11

Comparison of the the accuracy of proximity-based models and MWT extensions on title queries

TREC	Farasa Stemming						Light Stemming					
	2002/2001		2001		2002		2002/2001		2001		2002	
Model/Metric	MAP	P10	MAP	P10	MAP	P10	MAP	P10	MAP	P10	MAP	P10
LGD_CT	32.47	45.60	36.04	56.00	30.68	40.40	28.44	<i>42.67</i>	34.12	<i>55.20</i>	25.60	<i>36.40</i>
LGD_MWT	33.09	<i>46.53</i>	36.58	<i>56.80</i>	31.31	<i>42.20</i>	28.90	41.73	34.63	52.80	26.03	36.20
SPL_CT	32.05	48.13	36.47	64.40	29.84	40.00	29.59	45.60	34.59	58.40	27.09	<i>39.20</i>
SPL_MWT	32.34	<i>48.13</i>	36.80	<i>64.80</i>	30.11	<i>39.80</i>	29.96	<i>45.20</i>	36.18 ↑	<i>60.80</i>	28.85	37.40
PL2_CT	32.41	48.27	36.69	63.20	30.27	40.80	29.67	<i>45.07</i>	35.39	<i>57.20</i>	26.81	<i>39.00</i>
DFR_TD	32.00	48.40	36.20	63.60	29.90	40.80	29.59	44.27	35.21	56.00	26.78	38.40
PL2_MWT	32.74	<i>48.67</i>	37.11	<i>63.60</i>	30.55	<i>41.20</i>	29.56	43.60	35.50	56.80	26.59	37.00
CRTER	33.31	<i>48.93</i>	37.96	<i>63.20</i>	30.99	<i>41.80</i>	29.61	<i>43.65</i>	35.41	<i>53.35</i>	26.71	<i>38.80</i>
BM25_MWT	33.73	<i>49.33</i>	38.58	<i>63.60</i>	31.31	<i>42.20</i>	30.50	<i>44.27</i>	36.85 ↑	<i>57.20</i>	27.32	37.40
LM_CT	31.90	<i>46.00</i>	34.45	<i>56.00</i>	30.63	<i>41.00</i>	28.50	<i>43.33</i>	33.32	52.40	26.10	<i>38.80</i>
PLM	29.97	45.07	32.96	54.80	28.47	40.20	27.38	42.13	32.02	50.80	25.07	37.80
MRF	31.38	45.87	33.95	<i>56.00</i>	30.10	40.80	28.02	42.67	32.50	52.40	25.78	37.80
QLM	31.50	<i>46.00</i>	34.03	<i>56.00</i>	30.23	<i>41.00</i>	28.29	41.47	34.52	54.80	25.18	35.60
LM_MWT	31.63	<i>46.00</i>	34.14	<i>56.00</i>	30.38	<i>41.00</i>	28.12	41.27	33.18	<i>54.50</i>	25.59	36.20

Table 12

Comparison of the accuracy of proximity-based models and MWT extensions on title-description queries

TREC	Farasa Stemming						Light Stemming					
	2002/2001		2001		2002		2002/2001		2001		2002	
Model/Metric	MAP	P10	MAP	P10	MAP	P10	MAP	P10	MAP	P10	MAP	P10
LGD_CT	34.23	48.27	38.34	60.80	32.18	42.00	29.92	44.00	34.06	59.20	27.85	36.40
LGD_MWT	34.96	<i>48.67</i>	39.23	<i>61.20</i>	32.82	<i>42.40</i>	30.56	46.67	34.81	<i>63.60</i>	28.43	<i>38.20</i>
SPL_CT	35.28	50.93	39.12	63.60	33.36	44.60	31.66	47.07	35.23	62.40	29.87	39.40
SPL_MWT	35.88	<i>51.87</i>	39.92	<i>65.60</i>	33.86	<i>45.00</i>	31.83	<i>49.33</i>	36.14	<i>65.60</i>	29.68	<i>41.20</i>
PL2_CT	34.99	51.20	38.40	63.20	33.29	45.20	31.24	46.67	34.76	62.40	29.48	38.80
DFR_TD	34.24	50.80	37.45	62.00	32.63	45.20	30.90	47.07	35.20	60.80	28.75	40.20
PL2_MWT	35.55	<i>51.20</i>	39.25	<i>62.80</i>	33.70	<i>45.40</i>	31.50	<i>47.87</i>	35.85	<i>61.60</i>	29.32	<i>41.00</i>
CRTER	35.02	50.93	37.96	61.20	33.54	45.80	31.56	47.73	35.55	62.00	29.56	40.60
BM25_MWT	35.56	<i>51.47</i>	38.60	<i>61.20</i>	34.04	<i>46.60</i>	31.86	<i>48.93</i>	36.94 ↑	<i>65.20</i>	29.32	<i>40.80</i>
LM_CT	32.85	47.47	34.91	57.60	31.82	42.40	29.33	45.07	32.09	58.00	27.95	38.60
PLM	31.66	46.93	33.65	56.80	30.66	42.00	28.49	43.73	31.03	54.80	27.23	38.20
MRF	32.04	47.07	33.95	56.80	31.09	42.20	29.37	44.13	32.72	56.80	27.69	37.80
QLM	32.54	47.07	34.58	56.80	31.52	42.20	29.44	44.67	32.22	55.60	28.05	39.20
LM_MWT	33.32	<i>47.87</i>	35.19	<i>58.00</i>	32.38	<i>42.80</i>	29.62	<i>46.00</i>	32.90	<i>58.00</i>	27.98	<i>40.00</i>

550 The overall comparison results show that incorporating MWT into the IR models, except the language
551 model, yields to a slightly better improvement over the proximity-based models. For Farasa stemmer,
552 LGD_MWT and BM25_MWT achieve the best MAP performance. For the light stemmer, the best MAP
553 values for all test collections are obtained by the SPL_MWT and the BM25_MWT models. Moreover,
554 both SPL extensions (SPL_CT and SPL_MWT) yield to a better P10 than the other models. Even though
555 the SPL_MWT and BM25_MWT improve significantly the CRTER and SPL_CT on the TREC-2001, the
556 overall comparison results show that there is no significant difference between models that are based on CTs
557 or explicit MWTs. Unlike the LM model, integrating MWT to the BM25, SPL, PL2 and LGD models shows
558 a small enhancement over the CT models for light stemming approach. Furthermore, the obtained results for
559 title-description queries show that all MWT models achieve a better performance than the proximity-based

560 models on most test collection. Besides, the difference between MWT models and proximity-based models
561 is not statistically significant.

562 8. Discussion

563 The results illustrate that word level analysis has a real impact on the accuracy of Arabic IR for all evalu-
564 ated models. Concerning stemming approaches, the Farasa stemming approach achieves better performance
565 than the light and the heavy stemming approaches, in agreement with the previous study (Abdelali et al.,
566 2016). The latter is explained by the high accuracy of Farasa word segmentation (Darwish and Mubarak,
567 2016) and the light stemmer fails to discriminate conjunctions and prepositions from the core words. In line
568 with previous studies(Larkey et al., 2002; Goweder et al., 2004), the light stemming approach outperforms
569 the heavy stemming approach. The low performance of the heavy stemming approach is explained by the
570 fact that the root-based stemmer conflates words with different meanings into the same root. The results of
571 the earlier studies, which showed that heavy stemmers are more effective than light stemmers for Arabic IR,
572 were mostly obtained on relatively small corpora; on such corpora, relying on roots increase the probability
573 of matching query terms to document terms (Abu El-Khair, 2007). Moreover, IR models that capture the
574 informativeness of a term for a given document (LGD, SPL, and PL2) are more appropriate for the heavy
575 stemmer approach.

576 Concerning the integration of term dependencies, both approaches, incorporating explicit MWTs or
577 relying on term proximity (especially the cross term extension) significantly improve the performance of
578 IR models, be it in conjunction with the three stemmers (IR models based on term dependencies are thus
579 useful for retrieving the Arabic content where stemming techniques introduce a certain amount of noise in
580 document representations). These findings confirm that term dependencies or term proximity are useful
581 for enhancing the retrieval performance on noisy content representation (Ye et al., 2013). Furthermore, the
582 comparison of the results of models based on explicit MWTs and models based on term proximity show that
583 incorporating MWTs to the SPL and BM25 yields the best accuracy (in terms of MAP and precision at 10)
584 for Arabic IR, although the results are not statistically different on most test collections. The good behavior
585 of CT-based extensions can be explained by the fact that they capture distant term dependencies; their
586 importance gradually increases with the decrease of the distance between query terms. Their drawback,
587 however, is that the IR system has to look over each document to calculate CT statistics (within-document
588 frequency and document frequency) for each query. On the other hand, extracting MWTs based on linguistic
589 and statistic parameters leads to better document and query representations. The disadvantage of using
590 MWTs as term dependencies lies on indexing additional terms, which increase the size of the index and
591 more off-line processing (tagging the corpus and MWT extraction).

592 Lastly, as conjectured in Section 6, PLM is the model that performs the worst. It is here the only model
593 that does not satisfy the compound condition: PLM does not integrate compound dependencies effectively.

594 9. Conclusion

595 In this paper, we have investigated the impact of term proximity and explicit MWTs for Arabic IR based
596 on term dependencies, using and comparing three different stemming approaches (Farasa, light and heavy
597 stemming). Our analysis led us to conclude that:

- 598 1. The Farasa stemmer is in general preferable to the classical light and the heavy stemming approaches;
- 599 2. The use of cross-terms and MWT extensions for all the standard models (LM, BM25, LGD, and SPL)
600 led to significant improvements; this has to be contrasted with the absence of significant improvements
601 obtained with the positional, Markov random field and quantum (models PLM, MRF and QLM)
602 extensions of the language model;
- 603 3. The best overall results are obtained by integrating MWTs to the SPL and BM25 models. The model
604 CRTER is particularly interesting on the Arabic collections used in this study. More generally, if the
605 integration of MWTs leads to slightly better results than the use of cross-terms, the difference is not

606 significant in most cases. Thus, the choice for one or the other method depends on other considerations
607 than mere IR performance.

608 To the best of our knowledge, this is the first study that provides (a) a complete cross-term extension
609 for standard IR models, (b) a complete comparison of the most important IR models integrating term
610 dependencies (18 different models are compared in our experiments), in the context of Arabic IR, and (c) a
611 compound condition that allows one to characterize the different models. Future work will focus on trying
612 to go one step further in the integration of term dependencies by trying to capture semantic dependencies
613 through word embedding and to integrate such dependencies within the models we have considered here.

614 References

- 615 Abdelali, A., Darwish, K., Durrani, N., Mubarak, H., 2016. Farasa: A fast and furious segmenter for arabic, in: Proceedings
616 of the Demonstrations Session, NAACL HLT 2016, The 2016 Conference of the North American Chapter of the Association
617 for Computational Linguistics: Human Language Technologies, San Diego California, USA, June 12-17, 2016, pp. 11–6.
- 618 Abu El-Khair, I., 2007. Arabic information retrieval. *Annual review of information science and technology* 41, 505–33.
- 619 Abu-Salem, H., Al-Omari, M., Evens, M.W., 1999. Stemming methodologies over individual query words for an arabic infor-
620 mation retrieval system. *Journal of the American Society for Information Science* 50, 524–9.
- 621 Al-Kabi, M., Al-Radaideh, Q.A., Akkawi, K.W., 2011. Benchmarking and assessing the performance of arabic stemmers. *J.*
622 *Information Science* 37, 111–9.
- 623 Al-Kharashi, I.A., Evens, M.W., 1994. Comparing words, stems, and roots as index terms in an arabic information retrieval
624 system. *Journal of the American Society for Information Science* 45, 548–60.
- 625 Al-Shawakfa, E., Al-Badarnah, A., Shatnawi, S., Al-Rabab'ah, K., Bani-Ismael, B., 2010. A comparison study of some arabic
626 root finding algorithms. *JASIST* 61, 1015–24.
- 627 Aljlal, M., Frieder, O., 2002. On arabic search: Improving the retrieval effectiveness via a light stemming approach, in:
628 Proceedings of the Eleventh International Conference on Information and Knowledge Management, pp. 340–7.
- 629 Amati, G., Van Rijsbergen, C.J., 2002. Probabilistic models of information retrieval based on measuring the divergence from
630 randomness. *ACM Trans. Inf. Syst.* 20, 357–89.
- 631 Beesley, K.R., 1996. Arabic finite-state morphological analysis and generation, in: Proceedings of the 16th Conference on
632 Computational Linguistics - Volume 1, Association for Computational Linguistics, Stroudsburg, PA, USA. pp. 89–94.
- 633 Boulaknadel, S., Daille, B., Aboutajdine, D., 2008a. A multi-word term extraction program for arabic language., in: LREC,
634 European Language Resources Association. pp. 380–3.
- 635 Boulaknadel, S., daille, B., driss, A., 2008b. Multi-word term indexing for arabic document retrieval, in: Computers and
636 Communications, 2008. ISCC 2008. IEEE Symposium on, pp. 869–73.
- 637 Bounhas, I., Slimani, Y., 2009. A hybrid approach for arabic multi-word term extraction, in: Natural Language Processing
638 and Knowledge Engineering, 2009. NLP-KE 2009. International Conference on, pp. 1–8.
- 639 Clinchant, S., Gaussier, E., 2010. Information-based models for ad hoc ir, in: Proceedings of the 33rd International ACM
640 SIGIR Conference on Research and Development in Information Retrieval, ACM, New York, NY, USA. pp. 234–41.
- 641 Clinchant, S., Gaussier, E., 2011. Retrieval constraints and word frequency distributions a log-logistic model for ir. *Inf. Retr.*
642 14, 5–25.
- 643 Croft, W.B., Turtle, H.R., Lewis, D.D., 1991. The use of phrases and structured queries in information retrieval, in: Proceedings
644 of the 14th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, ACM,
645 New York, NY, USA. pp. 32–45.
- 646 Darwish, K., Magdy, W., 2014. Arabic information retrieval. *Found. Trends Inf. Retr.* 7, 239–342.
- 647 Darwish, K., Mubarak, H., 2016. Farasa: A new fast and accurate arabic word segmenter, in: Proceedings of the Tenth
648 International Conference on Language Resources and Evaluation LREC 2016, Portorož, Slovenia, May 23-28, 2016.
- 649 Diab, M.T., 2009. Second generation amira tools for arabic processing: Fast and robust tokenization, pos tagging, and base
650 phrase chunking, in: In 2nd International Conference on Arabic Language Resources and Tools.
- 651 Drymonas, E., Zervanou, K., Petrakis, E.G.M., 2010. Exploiting multi-word similarity for retrieval in medical document
652 collections: the TSRM approach. *JDIM* 8, 316–22.
- 653 Dunning, T., 1993. Accurate methods for the statistics of surprise and coincidence. *Comput. Linguist.* 19, 61–74.
- 654 El Mahdaouy, A., El Alaoui Ouatik, S., Gaussier, E., 2013. A Study of Association Measures and their Combination for Arabic
655 MWT Extraction, in: Terminology and Artificial Intelligence, Paris, France. pp. 45–52.
- 656 Fagan, J., 1987. Automatic phrase indexing for document retrieval, in: Proceedings of the 10th Annual International ACM
657 SIGIR Conference on Research and Development in Information Retrieval, ACM, New York, NY, USA. pp. 91–101.
- 658 Fang, H., Tao, T., Zhai, C., 2004. A formal study of information retrieval heuristics, in: Proceedings of the 27th Annual
659 International ACM SIGIR Conference on Research and Development in Information Retrieval, ACM, New York, NY, USA.
660 pp. 49–56.
- 661 Fang, H., Zhai, C., 2014. Axiomatic analysis and optimization of information retrieval models, in: Proceedings of the 37th
662 International ACM SIGIR Conference on Research & Development in Information Retrieval, ACM, New York, NY,
663 USA. pp. 1288–.

- 664 Farghaly, A., 2004. Computer processing of arabic script-based languages. current state and future directions, in: Farghaly, A.,
665 Megerdooian, K. (Eds.), COLING 2004 Computational Approaches to Arabic Script-based Languages, COLING, Geneva,
666 Switzerland. pp. 1–.
- 667 Frantzi, K., Ananiadou, S., Mima, H., 2000. Automatic recognition of multi-word terms: the c-value/nc-value method. Inter-
668 national Journal on Digital Libraries 3, 115–30.
- 669 Gao, J., Nie, J.Y., Wu, G., Cao, G., 2004. Dependence language model for information retrieval, in: Proceedings of the 27th
670 Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 170–7.
- 671 Goweder, A., Poesio, M., De Roeck, A., 2004. Broken plural detection for arabic information retrieval, in: Proceedings of
672 the 27th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, ACM, New
673 York, NY, USA. pp. 566–7.
- 674 Habash, N., Rambow, O., 2005. Arabic tokenization, part-of-speech tagging and morphological disambiguation in one fell swoop,
675 in: Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics, Association for Computational
676 Linguistics. pp. 573–80.
- 677 Habash, N., Rambow, O., 2009. Mada+token: a toolkit for arabic tokenization, diacritization, morphological disambiguation,
678 pos tagging, stemming and lemmatization, in: Proceedings of the Second International Conference on Arabic Language
679 Resources and Tools, pp. 102–9.
- 680 Habash, N., Roth, R., Rambow, O., Eskander, R., Tomeh, N., 2013. Morphological analysis and disambiguation for dialectal
681 arabic, in: Human Language Technologies: Conference of the North American Chapter of the Association of Computational
682 Linguistics, Proceedings, June 9-14, 2013, Westin Peachtree Plaza Hotel, Atlanta, Georgia, USA, pp. 426–32.
- 683 Habert, B., Jacquemin, C., 1993. Noms composés, termes, dénominations complexes: problématiques linguistiques et traite-
684 ments automatiques. TAL. Traitement automatique des langues 34, 119–38.
- 685 Haddad, H., 2003. French noun phrase indexing and mining for an information retrieval system, in: Nascimento, M., de Moura,
686 E., Oliveira, A. (Eds.), String Processing and Information Retrieval. Springer Berlin Heidelberg. volume 2857 of *Lecture*
687 *Notes in Computer Science*, pp. 277–86.
- 688 He, B., Huang, J.X., Zhou, X., 2011. Modeling term proximity for probabilistic information retrieval models. Inf. Sci. 181,
689 3017–31.
- 690 Hmeidi, I., Kanaan, G., Evens, M., 1997. Design and implementation of automatic indexing for information retrieval with
691 arabic documents. Journal of the American Society for Information Science 48, 867–81.
- 692 Huston, S., Culpepper, J.S., Croft, W.B., 2014. Indexing word sequences for ranked retrieval. ACM Trans. Inf. Syst. 32,
693 3:1–3:26.
- 694 Jacquemin, C., Klavans, J.L., Tzoukermann, E., 1997. Expansion of multi-word terms for indexing and retrieval using morphol-
695 ogy and syntax, in: Proceedings of the 35th Annual Meeting of the Association for Computational Linguistics and Eighth
696 Conference of the European Chapter of the Association for Computational Linguistics, Association for Computational Lin-
697 guistics, Stroudsburg, PA, USA. pp. 24–31.
- 698 Kageura, K., Umino, B., 1996. Methods of automatic term recognition: A review. Terminology 3, 259–89.
- 699 Khoja, S., Garside, R., 1999. Stemming Arabic Text, in: Computing Department. Lancaster University.
- 700 Larkey, L., Ballesteros, L., Connell, M., 2007. Light stemming for arabic information retrieval, in: Soudi, A., Bosch, A.d.,
701 Neumann, G. (Eds.), Arabic Computational Morphology. Springer Netherlands. volume 38 of *Text, Speech and Language*
702 *Technology*, pp. 221–43.
- 703 Larkey, L.S., Ballesteros, L., Connell, M.E., 2002. Improving stemming for arabic information retrieval: Light stemming
704 and co-occurrence analysis, in: Proceedings of the 25th Annual International ACM SIGIR Conference on Research and
705 Development in Information Retrieval, New York, NY, USA. pp. 275–82.
- 706 Lv, Y., Zhai, C., 2009. Positional language models for information retrieval, in: Proceedings of the 32Nd International ACM
707 SIGIR Conference on Research and Development in Information Retrieval, New York, NY, USA. pp. 299–306.
- 708 Lvovsky, A.I., 2004. Iterative maximum-likelihood reconstruction in quantum homodyne tomography. Journal of Optics B:
709 Quantum and Semiclassical Optics 6, 556–9.
- 710 Maamouri, M., Bies, A., 2010. The penn arabic treebank, in: Farghali, A. (Ed.), Arabic Computational Linguistics. CSLI
711 studies in Computational Linguistics, Stanford, CA, pp. 103–35.
- 712 Metzler, D., Croft, W.B., 2005. A markov random field model for term dependencies, in: Proceedings of the 28th Annual
713 International ACM SIGIR Conference on Research and Development in Information Retrieval, New York, NY, USA. pp.
714 472–9.
- 715 Mitra, M., Buckley, C., Singhal, A., Cardie, C., 1997. An analysis of statistical and syntactic phrases, in: Proceedings of RIAO,
716 pp. 200–14.
- 717 Mustafa, M., AbdAlla, H., Suleman, H., 2008. Current Approaches in Arabic IR: A Survey. Springer Berlin Heidelberg, Berlin,
718 Heidelberg. pp. 406–7.
- 719 Nwesri, A., Tahaghoghi, S., Scholer, F., 2005. Stemming arabic conjunctions and prepositions, in: Consens, M., Navarro, G.
720 (Eds.), String Processing and Information Retrieval. Springer Berlin Heidelberg. volume 3772 of *Lecture Notes in Computer*
721 *Science*, pp. 206–17.
- 722 Pasha, A., Al-Badrashiny, M., Diab, M., Kholy, A.E., Eskander, R., Habash, N., Pooleery, M., Rambow, O., Roth, R.,
723 2014. Madamira: A fast, comprehensive tool for morphological analysis and disambiguation of arabic, in: Chair, N.C.C.,
724 Choukri, K., Declerck, T., Loftsson, H., Maegaard, B., Mariani, J., Moreno, A., Odijk, J., Piperidis, S. (Eds.), Proceedings
725 of the Ninth International Conference on Language Resources and Evaluation (LREC'14), European Language Resources
726 Association (ELRA), Reykjavik, Iceland.
- 727 Peng, J., Macdonald, C., He, B., Plachouras, V., Ounis, I., 2007. Incorporating term dependency in the dfr framework,
728 in: Proceedings of the 30th Annual International ACM SIGIR Conference on Research and Development in Information

729 Retrieval, ACM, New York, NY, USA. pp. 843–4.

730 Ponte, J.M., Croft, W.B., 1998. A language modeling approach to information retrieval, in: Proceedings of the 21st Annual
731 International ACM SIGIR Conference on Research and Development in Information Retrieval, ACM, New York, NY, USA.
732 pp. 275–81.

733 Robertson, S.E., Walker, S., Jones, S., Hancock-Beaulieu, M., Gatford, M., 1994. Okapi at trec-3, in: TREC'94, pp. 109–26.

734 Romeo, S., Martino, G.D.S., Belinkov, Y., Barrón-Cedeño, A., Eldesouki, M., Darwish, K., Mubarak, H., Glass, J., Moschitti,
735 A., 2017. Language processing and learning models for community question answering in arabic. Information Processing &
736 Management .

737 SanJuan, E., Ibekwe-SanJuan, F., 2010. Multi word term queries for focused information retrieval, in: Gelbukh, A. (Ed.),
738 Computational Linguistics and Intelligent Text Processing. Springer Berlin Heidelberg. volume 6008 of *Lecture Notes in*
739 *Computer Science*, pp. 590–601.

740 Shi, L., Nie, J.Y., 2009. Integrating phrase inseparability in phrase-based model, in: Proceedings of the 32Nd International
741 ACM SIGIR Conference on Research and Development in Information Retrieval, ACM, New York, NY, USA. pp. 708–9.

742 Shi, L., Nie, J.Y., 2010. Using various term dependencies according to their utilities, in: Proceedings of the 19th ACM
743 International Conference on Information and Knowledge Management, ACM, New York, NY, USA. pp. 1493–6.

744 Song, F., Croft, W.B., 1999. A general language model for information retrieval, in: Proceedings of the Eighth International
745 Conference on Information and Knowledge Management, pp. 316–21.

746 Sordoni, A., Nie, J.Y., Bengio, Y., 2013. Modeling term dependencies with quantum language models for ir, in: Proceedings of
747 the 36th International ACM SIGIR Conference on Research and Development in Information Retrieval, ACM, New York,
748 NY, USA. pp. 653–62.

749 Srikanth, M., Srihari, R., 2002. Biterm language models for document retrieval, in: Proceedings of the 25th Annual International
750 ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 425–6.

751 Ye, Z., He, B., Wang, L., Luo, T., 2013. Utilizing term proximity for blog post retrieval. *Journal of the American Society for*
752 *Information Science and Technology* 64, 2278–98.

753 Zhai, C., 2008. Statistical language models for information retrieval a critical review. *Found. Trends Inf. Retr.* 2, 137–213.

754 Zhai, C., Lafferty, J., 2001. A study of smoothing methods for language models applied to ad hoc information retrieval,
755 in: Proceedings of the 24th Annual International ACM SIGIR Conference on Research and Development in Information
756 Retrieval, ACM, New York, NY, USA. pp. 334–42.

757 Zhang, W., Yoshida, T., Tang, X., 2011. A comparative study of tf*idf, lsi and multi-words for text classification. *Expert Syst.*
758 *Appl.* 38, 2758–65.

759 Zhao, J., Huang, J.X., He, B., 2011. Crter: Using cross terms to enhance probabilistic information retrieval, in: Proceedings
760 of the 34th International ACM SIGIR Conference on Research and Development in Information Retrieval, ACM, New York,
761 NY, USA. pp. 155–64.

762 Zhu, Y., Xue, Y., Guo, J., Lan, Y., Cheng, X., Yu, X., 2012. Exploring and exploiting proximity statistic for information
763 retrieval model, in: Hou, Y., Nie, J.Y., Sun, L., Wang, B., Zhang, P. (Eds.), *Information Retrieval Technology*. Springer
764 Berlin Heidelberg. volume 7675 of *Lecture Notes in Computer Science*, pp. 1–13.