USING WIKIPEDIA FOR RETRIEVING ARABIC DOCUMENTS

Mohamed I. Eldesouki*, Waleed Arafa*, Kareem Darwish**, Mervat Gheith*

**Institute of Statistical Studies and Research, Computer Science Department, Cairo University*
5 Dr. Ahmed Zwel Street, Orman, Giza, Egypt

***Microsoft, Microsoft Innovation Center, Smart Village –Building B115, Kilo 28, Cairo/Alex. Desert Road*
Abou Rawash, Egypt
disooqi@ieee.org, waleed_arafa@hotmail.com, kareem@darwish.org, mervat_gheith@yahoo.com

Keywords: Arabic Information Retrieval, Text Processing, Wikipedia, Word Sense Disambiguation

Abstract: Although stemming techniques outperform other techniques of text processing, they miss many cases that needs to be conflated into one class. For instance, synonyms that belong to different roots can't be conflated to the same class using stemming techniques.

In this work, we investigate a new technique for information retrieval for Arabic documents based on concepts to overcome the above problems using the Arabic Wikipedia project. Word sense disambiguation is used for terms that have multiple senses. The new technique has been evaluated with different word sense disambiguation techniques. It also has been examined with different version of Arabic Wikipedia dumps to show that the performance increases evolutionary as Wikipedia develop.

After comparing with the results of experiments that use stemming techniques in (Disooqi and Arafa, 2009), although the stemming technique is still better, the continuous growth of Wikipedia improves the performance. Results show that the information retrieval performance is improving as Wikipedia develops and grows.

1. INTRODUCTION

There are many cases when two words are not quite the same but you would like a match to occur. Conversely, there could be two words that are identical but you wouldn't like match to occur. There are many reasons for such problems; some of these are related to the characteristic of the language itself and other depend on the understanding of the query and documents. In Arabic language, one reason of the first problem is the morphology system that is used to form the various forms of words. Although Arabic morphology system could produce different meaning for different morphological form, sometime you would like matches to occur between these different forms. For example, sometimes you would like a match to occur between a word and its plural form. Another reason is the affixes system of the language, for example, articles in Arabic language concatenate at the beginning of nouns which prevent from matching to nouns without articles and some conjunctions, prepositions and pronouns exist as a prefix for the words. A third reason may arise from habits of writing; some people neglect the writing of HAMZA for the ALEF letter other use diacritics, etc. Forth reason is the existence of multiple synonyms for a word. Fifth, the match could be between word against phrase or phrase against a phrase which, in case of bag of words representation, is not going to match. The second problem, which was the unwillingness matching of two words have the same spelling, happens because the two words have different meaning; the phenomena called "polysemy".

Different techniques have been developed to overcome the difficulties for matching process including normalization process, stemming process, morphological analysis process, n-gram for words, using ontologies, etc. The following section, previous work, discusses some of them. Normalization process is used to address the problem of habits of writing. Normalization removes the diacritics so that words without diacritics match with words that have diacritics and normalize the use of HAMZA and TAA MARBOUTA in words, it also could remove Kashida. Usually normalization is used in conjunction with the aforementioned techniques. It is performed at the beginning of the information retrieval process after

tokenizing the query and the documents. Another technique is stemming the words; it just removes the most frequent prefixes and suffixes of the word to obtain its stem (Aljlayl, 2002), (Larkey, 2002). Stemming technique gives the highest performance till now. It overcomes word n-gram and morphological analysis techniques (Larkey, 2005). However, multiple synonyms, language morphology and polysemy problem are still exist. Some systems use ontologies to help understand the queries and documents to improve the performance (Bhogal, 2007). Some systems use ontologies to handle the query clarification process by regarding the spatial information that the query and documents may have or by involving the different relations of the ontology to solve this kind of problems (Fu, G. et al., 2005).

In this work, we are going to investigate a new technique for text processing for Arabic documents based on a controlled vocabulary extracted from the Arabic Wikipedia project to overcome the above problems. This technique provides a better way to represent the queries and the documents as a set of concepts rather than a bag of words. The technique could be use as an additional step for processing text in IR system. It handles synonymy, polysemy, and the other aforementioned problems. Then, we are going to evaluate the new technique with different word sense disambiguation techniques. After that, the results will be compared to the results of our previous experiments that use stemming techniques (Disooqi and Arafa, 2009).

The rest of the paper is organized as follows: section 2 presents the previous work; section 3 presents the methods of using Wikipedia as source of concepts; section 4 briefly introduces the different disambiguation techniques that have been examined, section 5 describes the experiment carried out to evaluate the stemmers. Results and discussions are provided in section 6 and conclusion is derived in section 7.

2. PREVIOUS WORK

Many techniques have been used for beating the problem of information retrieval for the Arabic language. At the very beginning, researchers tried to use dictionaries of roots and stems, built manually, for each word to be indexed. The roots and stems extracted from a very small collection of text (Al-Kharashi & Evens, 1994). This method is not suitable especially when the collection is very big. People tried to use Arabic morphological Analyzers to obtain the roots of the words automatically to be indexed. A lot of analyzers exist in that time have been used and evaluated; for example Khoja Morphological Analyzer (Khoja, 1999), Tim Buckwalter morphological analyzer 1.0 (LDC, 2002), ALPNET morphological analyzer (Beesley, 1996), and Sebawai (Darwish, 2002a).

A controversial issue at that time was whether to use roots or stems as terms for indexing. Several studies have claimed that roots outperform stems (Al-Kharashi & Evens, 1994), (Hmeidi et al 1997), (Abu-Salem, 1999) and (Darwish, 2001). However, most of the resent studies found that using stems as index terms outperform roots; (Aljlayl, 2002), (Larkey, 2002), (Darwish, 2002b), (Larkey, 2005), (Taghva, 2005), (Darwish, Hassan & Emam, 2005). The reason that the former researchers, that found roots better than stems for IR tasks, have done their experiment on small collections of text which is not enough for evaluation.

TREC 2001 and TREC 2002[1] Conferences help a lot for improving the performance of Arabic information retrieval systems. They also helped in evaluating the different techniques for handling Arabic language, in the cross-language Information retrieval tracks. They provided, with help from Linguistic Data Consortium LDC[2], a relatively large text collection to be used in evaluation. This helped in deciding which is more appropriate for use as index term in Arabic information retrieval systems.

Using the TREC-2001 Arabic corpus (LDC, 2001), experiments reveal that roots are not suitable because Arabic consists of a few thousands of roots. Analyzing each word to its root would conflate many words of different meaning to the same class. For example, the Arabic words for *office*, *book*, *Library*, *writer*, and *letter* have same root.

After TREC Arabic cross-language Information retrieval tracks (CLIR) (Gay & Oard, 2002), researchers have directed their research to use stems as index terms. They developed a lot of *stemmers* to handle Arabic Language in IR context. Many studies have been conducted in stemming techniques;

[1] http://trec.nist.gov/
[2] http://www.ldc.upenn.edu/

(Darwish, 2002b), (Aljlayl, 2002), (Larkey, 2002), (Chen & Gay, 2002), (Larkey, 2005), (Al Ameed et al., 2005), (Nwesri, 2005), (Kadri & Nie, 2006), (Nwesri, 2007), and (El-Beltagy & Rafea, 2009).

3. RETRIEVING USING WIKIPEDIA

The continuous growth of the Wikipedia project makes it a good source of a controlled vocabulary. Due to collaboration work of volunteers, the Wikipedia grows constantly and rabidly. This gives it more advantage than other resources which is fixed in size such as Arabic WordNet. The Wikipedia produces a database dump every 15 days. This makes the Wikipedia reflects the reality and makes it up-to-date.

In addition to the controlled vocabulary, Wikipedia provides an internal link structure between articles that could be used as a source of knowledge to perform several NLP task. As for the controlled vocabulary, thousands of tens of contributors are in charge of adding and updating links for articles.

As Wikipedia considered being a good resource, there are some drawbacks. For example, there could be lacking in some forms or affixes for some phrases or terms since they haven't been yet in a context forcing them come in specific morphological forms or with certain affixes. Another drawback is the need for a good sense disambiguation technique to disambiguate between the candidate concepts. Also, there exist some wrong internal links in the Arabic Wikipedia. Fortunately, although these drawbacks have an impact on using Wikipedia as a source of language processing, they disappear after a while as Wikipedia continuously evolves.

3.1. Methods and Algorithms

The database dump[3] of the Arabic Wikipedia project has been used as a resource to help retrieving documents based on concepts. The dump contains an SQL commands that run against different database management system to create the schema and its extension of the database. The main tables that have been used to extract the knowledge needed to our system are pages table, pagelinks table, redirects table, articles xml file.

The pages table contains many different kinds of pages other than the articles such as Meta pages, discussion pages, user pages, etc. Each kind belongs to a so-called namespace that represents its specific type (article, meta, discussion, etc). We are interested in the namespace 0, since it contains the articles. In addition to articles, namespace 0 also contains redirect and disambiguation pages.

After filtering out redirect and disambiguation pages from namespace 0, the remaining pages (the articles) are used to extract a controlled vocabulary that is used later as index terms of the documents to be retrieved.

Although, each of remaining articles could discuss a person, an object, an idea, a concept, or other, we are using the term 'concept' to represent the topic of any these articles. The reason of naming them as concept is just using the original naming in the previous work (Milne and Witten, 2008a).

Here, each remaining article is used as concepts; each article in the Arabic Wikipedia project is corresponding to one concept that represents it. Each concept takes an identifier, *concept id*, which is used later in the information retrieval process. Another kind of pages in Wikipedia project is called redirect pages. These redirect pages represent other names that articles could take such as synonyms, acronyms, and abbreviations; each redirect page represents one different name of the article. Each single article page could have many redirect pages and each redirect page points to only one article page. The redirect pages have been used to form part of the names (or the roughly synonyms) that may represent concepts.

Referring to an article through the text of another article allow the article name to take many forms to suit the context of the text by, for example, adding articles, prefixes, suffixes, number, gender, etc. These new forms are considered the rest of the names of the concepts.

The list of concepts along with their different names or synonyms is used to build the synonym-concept_ids dictionary (or *synDic* in Table 1) that is going to be used in next section.

3.2. Documents Processing and Retrieval

The main idea, here, is to substitute terms in the documents by the right concept ids. The substitution

[3] http://download.wikipedia.org/arwiki/

solves most of the problems that have been mentioned earlier because a single concept id represents all the synonyms, acronyms, and abbreviations that the replaced term might be. Meanwhile, the concept ids differentiate between two terms have the same spelling but differ in meaning (polysemy problem).

The substitution process of terms considers two steps. The first step is to detect phrases and assign for each of them the corresponding concept id(s) (in case of polysemy, a term can have several concept ids), and second step is to disambiguate between concepts for phrases that have multiple concepts.

The term detection task goes as follows: after tokenizing the document, the tokens are normalized using the unified normalization used in (Disooqi and Arafa, 2009). The document then is processed to generate word n-grams. The n-gram generation process differs from the usual way of producing n-gram; the concept ids are assigned during the n-gram generation process. See Algorithm in Table 1. While the system generates n-grams, it tries to match the n-gram to the synonyms of each different concept and assign the concept id(s) to the term in case a match occurs. Phrases or n-grams of several concept ids are saved for latter disambiguation in the second step. The size of the n-gram, n, is equal to longest synonym length. Although, there is small likelihood to produce wrong phrases, the customized method for generating n-gram has the advantage of reducing ambiguity by trying to detect longer phrases first.

The stopwords removal process begins after the detection process and the reason for that is some phrases may contain stopwords, which will not be matched if we remove the stopwords before the n-gram process. (In Wikipedia, stopwords don't have corresponding articles, so stopwords don't exist as concepts in the synonym-concept_ids dictionary).

In case of polysemy problem, phrases might lie under several concepts. The second step arise here as a technique to disambiguate the right sense of a term. Disambiguation techniques are illustrated in the next section. As a result unwilling match is prevented with the same spelling but with different in meanings phrase.

Our approach could be treated as an additional text processing step in the IR system. Therefore, several combinations are examined to assess the effect of existence of other processing techniques such as normalization and stemming in our technique. All these combination has examined and evaluated in result and discussion section.

Table 1: Algorithm of generating n-grams each with its prospective concept(s).

Input: *TokensQ* (queue of all document tokens), *synDic* (synonym-concept_ids dictionary), *n* (size of n-gram)
Output: list of phrases, each with its candidate concept(s)
Algorithm:

1) If TokensQ size = 0, then return;
2) Else If TokensQ size >= n, Choose first *n* tokens from the TokensQ into *nList* (a list of n-gram size).
3) Else, choose ***all*** tokens from the TokensQ into *nList*.
4) Constitute a phrase by concatenating all the tokens in *nList***.**
5) Try to find a corresponding *synonym(s)* for the phrase.
6) If (synonym(s) found in synDic)
   a) Assign the *concept_id(s)* to the phrase.
   b) Empty *nList* and dequeue the tokens of the phrase from the *TokensQ*
   c) Go to step 1.
7) Else (the phrase has no corresponding synonym)
   a) Then remove one token from the end of *nList*.
   b) Check the size of *nList* after removal
      i) If number of tokens that exist in *nList* = 0, dequeue the last removed token from TokenQ and go to step 1.
      ii) If number of tokens that exist in *nList* > 0, then go to step 4.

After replacing all the phrases and terms by their right concept id, we treat the document as if it is "bag of words"; however, it is actually a page of concepts. The rest of the information retrieval steps remain the same. An extra step is to go through the previous steps manually for substituting the query's terms by the intended concepts ids.

4. DISAMBIGUATION TECHNIQUES

If an n-gram has multiple concepts, then a concept disambiguation is going to happen. The disambiguation process starts after the phrase detecting process is completed for the document. The disambiguation process and its related techniques are originally introduced in (Milne and Witten, 2008a), (Milne and Witten, 2008b). We have to note here that this work is concerned more with evaluating retrieval effectiveness of using index term generated using

these techniques. To avoid redundancy, we avoid explaining these methods since they are illustrated in great detail in (Milne and Witten, 2008a), (Milne and Witten, 2008b) and (Cilibrasi and Vitanyi, 2007). However, this section briefly discusses the disambiguation methods.

All techniques used in experiments depend on the Arabic Wikipedia statistics such as in-links and out-links of an article, number of out-links of an article and others extracted from the link structure.

Two techniques have been examined to disambiguate between concepts and each has been evaluated. First technique is choosing *the most common concept* among all concepts. The commonness measure depends on number of articles refereeing to the article that representing the concept and is calculated by dividing the number of in-coming links to the page representing that concept divided by the sum of numbers in-coming links of all concepts being disambiguated.

The other disambiguation technique depends on so-called *semantic relatedness* between the candidate concept and the surrounding terms that possess only single concept (or so-called *context terms*).

We have examined three methods to compute the semantic relatedness; the first depends on the in-links counts, the second depends on out-links count and the third depends on the average between both the first and the second ways.

The idea is to choose the concept with highest average of semantic relatedness with context terms. Since the context terms are not the same in their representation of the context of a document, the semantic relatedness of the context terms are weighted. The weight expresses the importance of the context term to the document by averaging the semantic relatedness between the desired context term and all other context terms.

5. EXPERIMENTS

The experiments measure the effect of using index terms produced by our technique to improve retrieval effectiveness of the information retrieval system.

We examined the two methods of disambiguation techniques; the technique which depends on the most common concept and the technique that depends on semantic relatedness. In addition, we examined calculating semantic relatedness using the in-link method, out-link method and both in-link and out-link. Furthermore, to show that the performance of our technique improves as Wikipedia develops, we used different version of the Arabic Wikipedia project.

As we mentioned earlier, our technique could be used in existence of other text processing steps such as normalization and stemming. Four runs have been conducted to show the effect of other processing steps on our technique. These runs are Run1; is to only normalize the text and then applying our method to produce concept ids and use these ids only as index terms (we neglect other terms that hasn't been detected as phrases), Run2; is applying our technique after stemming the text and using only concept ids as term index, Run3; is to only normalize the text and then applying our method to generate concept ids, however, here we use the remaining terms (terms that has not identified as phrases) in addition to the ids as index terms, Run4; is same as Run3, however, the remaining terms are going to be stemmed.

The results of our techniques are compared with stemming techniques, since they outperform the other techniques for processing Arabic text (Disooqi and Arafa, 2009).

We have used TREC-2001 Arabic corpus for evaluation. TREC-2001 Arabic corpus, also called the AFP_ARB corpus, consists of 383,872 newspaper articles in Arabic from Agence France Presse. This fills up almost a gigabyte in UTF-8 encoding as distributed by the Linguistic Data Consortium. There were 25 and 50 topics used in 2001 and 2002 respectively with relevance judgments, available in Arabic, French, and English, with Title, Description, and Narrative fields. We used the Arabic titles and descriptions as queries of the 75 topics in the experiments.

For all the experiments, we used the Lemur language modeling toolkit[4], which was configured to use Okapi BM-25 term weighting with default parameters and with and without blind relevance feedback (the top 50 terms from the top 10 retrieved documents were used for blind relevance feedback). To observe the effect of alternate indexing terms, mean average precision, *MAP*, was used as the measure of retrieval effectiveness.

[4] http://www.lemurproject.org/

As a requirement for Arabic text to be indexed with Lemur toolkit, corpus and topics have been converted to CP1256 encoding. Then a normalization step was performed. The encoding conversion and normalization steps were conducted on both text collection and the topics where queries were extracted.

The experiments used three versions of Arabic Wikipedia database dump to show that the performance is improving as Wikipedia continuously grows. The first one has been completed on 2010-01-23 and contains about 119,000 articles and we call it (version1), the second completed on 2010-03-10 and contains about 122,662 articles and we call it (version2**)** and the third dump completed on 2010-05-31 and contains about 127,273 articles and we call it (version3).

In order to be able to compare the retrieval performance with the light stemmers mentioned in (Disooqi and Arafa, 2009), the same experiment parameters have been used for current work.

Six experiments have been conducted for each of the four runs with and without query expansion. Experiments Exp1, Exp2, Exp3, and Exp4 have used the same version, (version3), of Wikipedia database dumb (the latest version) to evaluate the disambiguation techniques. Experiment Exp1 used system that disambiguate between concepts using "*most common sense*" disambiguation technique; by choosing the most common concept of the candidate ones. Experiment Exp2 uses the disambiguation technique that depends on in-link concepts. Experiment Exp3 uses the disambiguation technique that depends on out-link concepts. Experiment Exp4 uses both in-links and out-links concepts for disambiguation. In the other hand, Experiment Exp5 and Exp6 use version1 and version2 respectively with the disambiguation technique that give highest performance through experiments Exp1, Exp2, Exp3 and Exp4.

One parameter has been used to adjust the disambiguation techniques for the system speed purpose. The parameter is used to reduce the number of candidate concepts to be disambiguated for an n-gram which reduces the overall computation time. The parameter calculates the percentage of appearance of a concept as out-link relevant to the sum of appearance of all the candidate concepts and neglecting concept under certain threshold. This threshold is set to 0.02 as in (Milne and Witten, 2008b).

6. RESULTS AND DISCUSSION

Table 2 and Table 3 show the six experiments for all the four runs with and without query expansion. By comparing the results of the four runs we found that Run4 slightly outperforms other runs. Run2 shows the worst result. One could justify the bad results of Run2 because for stemming step performed for text before using our approach which increased the number of candidate concept for each detected phrase in the document as well as decrease the number of context terms (terms with single concept that are used in disambiguation). As result the performance of the disambiguation technique declines. One could attribute the good result of Run4 to the lack of many representative terms from the controlled vocabulary extracted from Arabic Wikipedia.

Table 2: Mean Average Precisions for the different Experiments, with and without query expansion, using only concept ids as index terms. Run1 the documents only normalized before phrase detection and concept disambiguation. However, Run2 stems the documents before before phrase detection and concept disambiguation.

Experiment	Run1		Run2	
Query Expansion	without	with	without	with
Exp5 (v1+out-link)	0.3111	0.3312	0.2138	0.2301
Exp6 (v2+out-link)	0.3120	0.3550	0.2120	0.2350
Exp1 (v3+comm.)	0.3252	0.3561	0.2252	0.2961
Exp4 (v3 + both in-link and out-link)	0.3225	0.3721	0.2225	0.2621
Exp2 (v3 + in-link)	0.3290	0.3659	0.2290	0.2559
Exp3 (v3+out-link)	0.3301	0.3801	0.2301	0.2501

Table 3: Mean Average Precisions for the different experiments using as index terms the concept ids plus the remaining tokens that aren't detected as phrases. In Run3 the remaing tokens were normalized. However, in Run4 the remaining tokens were stemmed. Both runs were only normalized before phrase detection.

Experiment	Run3		Run4	
Query Expansion	without	with	without	with
Exp5 (v1+out-link)	0.3009	0.3387	0.319	0.3713
Exp6 (v2+out-link)	0.3120	0.3624	0.3213	0.3721
Exp1 (v3+comm.)	0.329	0.362	0.3219	0.3606
Exp4 (v3 + both in-link and out-link)	0.3245	0.3744	0.326	0.3798
Exp2 (v3 + in-link)	0.3288	0.3691	0.33	0.3731
Exp3 (v3+out-link)	0.3351	0.3752	0.3394	0.3813

As you can notice from experiments Exp5, Exp6 and Exp3 for all four runs, the mean average precisions are gradually increase. This indicates that as Wikipedia continually grows and develops the performance improves. Thus, even if the current retrieval system that uses the light stemming is outperform our system; there is likelihood that using Wikipedia could outperform retrieval using light stemming technique over time.

Experiments Exp1, Exp2, Exp3, and Exp4, use the same version of Wikipedia. The experiments evaluate the different disambiguation techniques. The result of all runs shows that disambiguation technique that use semantic relatedness that depends on out-links outperforms other techniques of disambiguation.

The following table, Table 4, shows the mean average precision for experiments conducted in (Disooqi and Arafa, 2009) for comparison's sake. In addition to experiment *raw* which was conducted on the Arabic News corpus without performing any normalization steps or stop words removal there are two other experiments, one conducted after normalization and stopwords removal process, called *normalized*, and the other after stemming the corpus using the light10 stemmer, called *Light10*.

The result in Table 4 shows that our technique for text processing outperforms raw text for information retrieval. Run1, Run3 and Run4 outperform system that use normalized text for retrieval. However, using just normalized index terms is better than index terms produced in Run2.

The results show that light10 stemmer - which outperforms all other stemming technique according to (Disooqi and Arafa, 2009) - outperforms our technique for both expanded and unexpanded form of queries for all runs. However, they show that the information retrieval performance is improving as Wikipedia develops and grows.

Table 4: Mean Average Precisions for three experiments conducted (Disooqi and Arafa, 2009) for the sake of comparison. Experiment Exp3 of Run4 (best perofrmance) is added to the table for ease of comparing.

Experiment	Unexpanded	Expanded
raw	0.2056	0.2645
normalized	0.2478	0.3057
Run4, Exp3 (v3+out-link)	0.3394	0.3813
Light10	0.3490	0.3982

7. CONCLUSION

One way to solve the synonymy and polysemy problem is using the concept-based information retrieval. The use for concepts in retrieval led to significantly higher performance than ordinary normalization process. A good disambiguation technique is needed for concept disambiguation.

Although the stemming technique is still better, the continuous growth of Wikipedia improves the performance of concept-based information retrieval. Results show that the information retrieval performance is improving as Wikipedia develops and grows. Also, Wikipedia is a good source for concepts since the new concepts that appear on the scene are frequently added.

REFERENCES

Abu-Salem, H., Al-Omari, M., and Evens, M. Stemming methodologies over individual query words for Arabic information retrieval. *JASIS*, 50 (6), pp. 524-529, 1999.

Al-Ameed k. Hayder, Al-Ketbi O. Shaikha, Al-Kaabi A. Amna, Al-Shebli S. Khadija, Al-Shamsi F. Naila, Al-Nuaimi H. Noura, Al-Muhairi S. Shaikha, Arabic Light Stemmer: A new Enhanced Approach, *The second international conference on innovations technology (IIT'05)*, 2005.

Al-Kharashi, I. and Evens, M. W. Comparing words, stems, and roots as index terms in an Arabic information retrieval system. *JASIS*, 45 (8), pp. 548-560, 1994.

Aljlayl, M., & Frieder, O, On Arabic search: Improving the retrieval effectiveness via light stemming approach. *In Proceedings of the 11th ACM International Conference on Information and Knowledge Management*, Illinois Institute of Technology (pp. 340–347). New York: ACM Press.2002.

Bhogal J., Macfarlane A., Smith P., A review of ontology based query expansion, *Information Processing and Management* 43, 866–886, 2007.

Beesley, K. R. Arabic finite-state morphological analysis and generation. *In COLING-96: Proceedings of the 16th international conference on computational linguistics*, vol. 1, pp. 89-94, 1996.

Chen, A., and Gey, F. Building an Arabic stemmer for information retrieval. *In TREC 2002. Gaithersburg: NIST*, pp 631-639, 2002.

Cilibrasi, R.L. and Vitanyi, P.M.B. (2007) The Google Similarity Distance. *IEEE Transactions on Knowledge and Data Engineering* 19(3), 370-383.

Darwish, K., Doermann, D., Jones, R., Oard, D., and Rautiainen, M. *TREC-10 experiments at Maryland: CLIR and video*. In TREC 2001. Gaithersburg: NIST, 2001.

Darwish, K. Building a shallow morphological analyzer in one day. *ACL 2002 Workshop on Computational Approaches to Semitic languages*, July 11, 2002.

Darwish, K. and Oard, D.W. CLIR Experiments at Maryland for TREC-2002: Evidence combination for Arabic-English retrieval. *In TREC 2002. Gaithersburg: NIST*, pp 703-710, 2002.

Darwish K., Hassan H., and Emam O., Examining the Effect of Improved Context Sensitive Morphology on Arabic Information Retrieval. *Proceedings of the ACL Workshop on Computational Approaches to Semitic Languages*, pages 25–30, Ann Arbor, June 2005.

De Roeck, A. and Al-Fares, W. (2000) A Morphologically Sensitive Clustering Algorithm for Identifying Arabic Roots, *Proceedings of the 38 th Annual Meeting of the ACL*, Hong Kong. pp. 199–206

El-Disooqi M., Arafa W. and Darwish K. Stemming techniques of Arabic Language: Comparative Study from the Information Retrieval Perspective. *The Egyptian Computer Journal* , Vol. 36 No. 1, June 2009.

El-Beltagy S., Rafea A.. A FRAMEWORK FOR THE RAPID DEVELOPMENT OF LIST BASED DOMAIN SPECIFIC ARABIC STEMMERS, *Proceedings of the Second International Conference on Arabic Language Resources and Tools*, 2009.

Fu, G. et al. . Ontology-Based Spatial Query Expansion in Information Retrieval ODBASE: OTM Confederated International Conferences, 4 November 2005.

Gey, F. C. and Oard, D. W. The TREC-2001 cross-language information retrieval track: Searching Arabic using English, French, or Arabic queries. *In TREC 2001. Gaithersburg: NIST*, 2002.

Hmeidi, I., Kanaan, G. and M. Evens (1997) Design and Implementation of Automatic Indexing for Information Retrieval with Arabic Documents. *Journal of the American Society for Information Science*, 48/10, pp. 867-881.

Khoja, S. and Garside, R. Stemming Arabic text. *Computing Department, Lancaster University*, Lancaster, 1999.

Kadri, Y., and Nie, J. Y. (2006), Effective stemming for Arabic information retrieval". The challenge of Arabic for NLP/MT Conference, The British Computer Society. London, UK.

Larkey, Leah S., Ballesteros, Lisa, and Connell, Margaret. (2002) Improving Stemming for Arabic Information Retrieval: Light Stemming and Co-occurrence Analysis. *In Proceedings of the 25th Annual International Conference on Research and Development in Information Retrieval (SIGIR 2002)*, Tampere, Finland, August 11-15, 2002, pp. 275-282.

Larkey, S. L., Ballesteros, L., and Connell, E. M. (2005), Light stemming for Arabic information retrieval. *Arabic Computational Morphology: Knowledge-based and Empirical Methods*.

LDC, Linguistic Data Consortium. LDC2001T55, 2001. http://www.ldc.upenn.edu/Catalog/CatalogEntry.jsp?catalogId=LDC2001T55

LDC, Linguistic Data Consortium. Buckwalter Morphological Analyzer Version 1.0, LDC2002L49, 2002. http://www.ldc.upenn.edu/Catalog/CatalogEntry.jsp?catalogId=LDC2002L49

Milne, D. and Witten, I.H. (2008a) An effective, low-cost measure of semantic relatedness obtained from Wikipedia links. *In Proc. of the first AAAI Workshop on Wikipedia and Artificial Intelligence (WIKIAI'08)*, Chicago, I.L.

Milne, D. and Witten, I.H. (2008b) Learning to link with Wikipedia. *In Proc. of the ACM Conference on Information and Knowledge Management (CIKM'08)*, Napa Valley, California.

Nwesri A., S.M.M Tahaghoghi, Falk Scholer, Stemming Arabic Conjunctions and Prepositions, In Mariano Consens and Gonzalo Navarro (eds.), *Lecture Notes in Computer Science - Proceedings of the Twelfth International Symposium on String Processing and Information Retrieval (SPIRE'2005)*, Buenos Aires, Argentina, 3772:206-217, November 2-4,2005.

Nwesri A., S.M.M. Tahaghoghi and Falk Scholer, Arabic Text Processing for Indexing and Retrieval, *Proceedings of the International Colloquium on Arabic Language Processing*, Rabat, Moroc, 18-19 June, 2007.

Rodríguez, R., Farwell, D., Farreres, J., Bertran, M., Alkhalifa, M., Martí, M.A., Black, W., Elkateb, S., Kirk, J., Pease, A., Vossen, P., and Fellbaum, C., (2008). Arabic WordNet: Current State and Future Extensions. *Proceedings of The Fourth Global WordNet Conference*, Szeged, Hungary. January 22-25, 2008.

Taghva, K., Elkoury, R., and Coombs, J. Arabic Stemming without a root dictionary. 2005.

Black, W., Elkateb, S., Rodriguez, H, Alkhalifa, M., Vossen, P., Pease, A. and Fellbaum, C., (2006). Introducing the Arabic WordNet Project, *in Proceedings of the Third International WordNet Conference*, Sojka, Choi, Fellbaum and Vossen eds.