2010

Corpus-based Arabic stemming using N-grams

Abdelaziz Zintouni
University of Wollongong in Dubai

Asma Damankesh
University of Wollongong in Dubai

Foroogh Barakati
University of Wollongong in Dubai

Maha Atari
University of Wollongong in Dubai

Mohamed Watfa
University of Wollongong in Dubai

See next page for additional authors

Corpus-based Arabic Stemming Using N-grams

Abdelaziz Zitouni, Asma Damankesh, Foroogh Barakati, Maha Atari, Mohamed Watfa, and Farhad Oroumchian

University of Wollongong in Dubai, POBox 20183, Dubai, UAE
az688@uow.edu.au, adamankesh@acm.org, shamim_66_87@yahoo.com, maoa704@uow.edu.au, {MohamedWatfa,FarhadOroumchian}@uowdubai.ac.ae

Abstract. In languages with high word inflation such as Arabic, stemming improves text retrieval performance by reducing words variants. We propose a change in the corpus-based stemming approach proposed by Xu and Croft for English and Spanish languages in order to stem Arabic words. We generate the conflation classes by clustering 3-gram representations of the words found in only 10% of the data in the first stage. In the second stage, these clusters are refined using different similarity measures and thresholds. We conducted retrieval experiments using row data, Light-10 stemmer and 8 different variations of the similarity measures and thresholds and compared the results. The experiments show that 3-gram stemming using the dice distance for clustering and the EM similarity measure for refinement performs better than using no stemming; but slightly worse than Light-10 stemmer. Our method potentially could outperform Light-10 stemmer if more text is sampled in the first stage.

Keywords: Arabic Stemmer, Information Retrieval, N-Gram, Corpus-based Stemmer

1 Introduction

The rapid growth of the internet has increased the number of documents available online. The latest statistics show that Arabic is the 7th most popular language used over the net by the end of the year 2009 [10]. Arabic Information retrieval faces many challenges due to the complex and rich nature of the Arabic language. Stemming is one of the techniques used to improve the Arabic information retrieval by reducing the words' variants into the base words like stems or roots.

Stemming improves the information retrieval by reducing the word mismatch between the query and the document. This will result in returning more relevant documents to the query. Stemming has a great effect on the retrieval when the language is highly inflected for example Arabic language [4]. There has been several attempts to solve the Arabic text stemming including constructing manual dictionary [2], affix removal which is also called light stemming [5, 4, 11, 13], morphological stemming [2, 6] and statistical stemming [3, 12].

In this paper, we have used a technique called Corpus-based Stemming that generates lists of words from the same root [19]. Then we have used these lists in Arabic information retrieval experiments and compared the results with a more complex and linguistic-based stemming approach known as Light-10 stemmer.

The remaining of this paper is structured as follows. In the next section, we describe the complexity of the Arabic language. Section 3 describes and compares similar stemming approaches to our proposed approach. Section 4 explains our approach to Arabic stemming for information retrieval. The results of experiments are described in section 5. Section 6 concludes this paper and suggests some potential improvements and future work.

2 The Arabic Language

The Arabic language is complex and has a rich grammar; it consists of 28 letters and a set of short vowels *(harakat)*, long vowels and nunation (تنوين , tanwin). Arabic text is written from right to left where some letters are "vocalized" and embrace diacritics. Interestingly enough the meaning of a word might change based on its diacritics (i.e. the word كَتَبَ [kataba: he wrote] is different from the word كُتُب [kotob: books] although they both written with the same three letters (k,t,b)).

Moreover, the Arabic language has a very complex morphology. Most of the words are created from a root of 3 letters. Other words have 4, 5 or 6 letters roots. Some of the words are constructed by attaching a prefix at the beginning or a suffix at the end of the root word. But, most of the adjectives, nouns and verbs are generated by infixing the root. The most challenging morphological problem in the Arabic language is that plural and singular forms of nouns are mostly irregular which makes it difficult to conflate them. Consequently, Arabic morphological analysis is a very complicated task and so far no single stemming technique has been able to resolve all the issues for all the cases.

These complexities in the Arabic language make it a highly inflated language where many similar words have variant morphological forms. This increases the likelihood of word mismatch in information retrieval systems. Therefore, stemming is a very important process in information retrieval where word conflation can be found and word matching between existing documents and queries can be improved to return more relevant documents. In the next section, we describe a number of stemming techniques focusing on the statistical stemming approach.

3 Related Research

Xu and Croft [19] have used a two stage approach in their pioneering work on corpus-based stemming. In the first stage, they experimented with both aggressive stemmers such as Porter and K-stem and also a trigram matching approach. They created equivalence classes that contained all the words with the same root. In the aggressive stemming method, they grouped all the words that generated

the same root with the stemmer in an equivalence class. In the trigram approach, they put all the words that started with the same three letters in the same equivalence class. In the second stage, they refined these equivalence classes by using a variation of the Expected Mutual Information Measure (EMIM) called EM to calculate the closeness of each pair of words in the same equivalence classes. The EM unlike $EMIM$ does not favor words with high frequency. For two terms a and b, the EM is calculated as below:

$$EM(a,b) = max(\frac{n_{ab} - En(a,b)}{n_a + n_b}, 0) \quad (1)$$

Where n_{ab} is the number of times a and b co-occur within a window in the corpus, n_a and n_b are number of times a and b appear in the corpus. $En(a,b)$ is the expected number of co-occurrences assuming a and b are statically independent and it is calculated as $k_{n_a n_b}$ where k is a constant calculated based upon the window size.

The Connected Component and Optimal Partition algorithms were used to cluster the words within the same equivalence classes into more refined groups of similar words based on their EM similarity values. Their experiments showed that using their approach, aggressive stemmers like Porter for English can be improved. They also showed that a crude method like trigram can be employed in the first stage of that process with little loss of performance. They have also applied their trigram approach to other languages such as Spanish [20]. The main assumption in this approach is that words that belong to the same equivalence class (i.e. have the same root) will co-occur in the same document or text window.

The N-gram is a language-independent approach in which each word is broken down into substrings of length N. This approach has been applied to information retrieval in many languages such as English [9], Turkish [8], Malay [16] and Farsi [1] with varying degrees of success. In [13], Larkey et al. have used the bigram and trigram string similarity approach for Arabic text retrieval. In their experiment, bigrams have performed better than the trigrams; however the N-gram approach did not perform well in general. Authors have traced back the problem to the peculiarities imposed by the Arabic infix structure that increases the N-gram mismatches. However, they did not use stemming in their approach. In [19], the N-gram is used with and without stemming and it was shown that stemming resulted in minor improvements in the search results.

Light-10 [13] is a stemming tool based on Arabic morphological analysis that uses a rule-based affix-removal technique for light stemming. The prefix and suffix of words are removed if certain conditions were satisfied. For example the letter 'و' can be removed from the beginning of the word if the remaining of the word has three or more characters. Light-10 was claimed to perform better than the other affix-removal approach proposed by Khoja and Garside [11], and Backwater Morphological Analyzer [3] . In Backwater Morphological Analyzer approach each word is segmented into a prefix, a stem and a suffix using three dictionaries and three compatibility tables. It then produces a list of all the possible analysis of each word. In [14], Larkey et al. have applied the approach proposed by Xu and Croft [19] on the Khoja stemmer [11] and Light-10

stemmer [13] and concluded that co-occurrence analysis has not improved the performance of the retrieval compared to the base stemmers. However, they reported that the corpus-based approach breaks down the equivalence classes into precise conflation groups with average size of five words. The reason for the low performance of their stemming strategy is claimed to be the complexity and the nature of the Arabic language. In the next section, our approach to the Arabic word conflation is explained in details.

4 N-Gram Conflation and Co-Occurrence Analysis for Language-Independent and Corpus-based Stemming

Our approach is based on the corpus-based approach developed by Xu and Croft [19] with some subtle differences. In their approach, they have used a trigram prefix matching for forming crude equivalence classes. That approach is not useful for Arabic because many nouns in Arabic have irregular plural forms.

Table 1. A few examples of conflation in Arabic words.

Word	Letters	Pronunciation	Meaning	New word	PoS	Letters
كتاب	KTAB	k e t A b	book	كتب	plural	KTB
قوم	QVM	q o m	nation	اقوام	plural	AQVAM
رسول	RSVL	r a s u l	prophet	رسل	plural	RSL

As it is depicted in Table 1 the plural forms of the nouns have different trigrams than the original words. That is why, we have used an N-gram approach instead of relying on only 3-letter prefixes in forming equivalence classes.

An N-gram is a string of consecutive N characters. Generally an N-gram approach involves representing a word with a vector of strings of length N formed from the consecutive letters of the word. The N-gram approach has mixed performances in information retrieval. In some languages like English, it results in a poor performance however in languages like Farsi, it has an acceptable performance [1]. As mentioned earlier, most Arabic words are made up of roots with three letters which led us to use trigrams for word segmentation. The general process undertaken is as follow:

1. The corpus is normalized by removing all the stopwords, numbers and diacritics, and then the set of unique words in the corpus is generated.
2. Words are passed to the N-gram algorithm and a set of overlapping trigram substrings is generated for each unique word. For example the 3-grams for the word كتاب are: كتا – تاب
3. A distance matrix is constructed and the Dice Distance is measured for each pair of words and recorded in this matrix. For two words a and b, the Dice measure (S_{ab}) is

$$S_{ab} = \frac{2C_{ab}}{C_a + C_b} \tag{2}$$

where S_{ab} is the similarity between a and b, C_{ab} is the number of trigrams shared by a and b, and C_a and C_b are number of trigrams in each word a and b.

4. The words have been clustered into large equivalence classes based on their pair wise Dice similarity measure and the Complete Linkage clustering Algorithm (CLA). We have assumed that if the similarity of the two words is less than a threshold then those words are not similar. This decision is made to increase the similarity of the words that are assigned to the same equivalence class. In total, we have experimented with three different thresholds ($t = 0.5$, $t = 0.6$ and $t = 0.7$).
5. The EM measure described in the previous section is used to calculate the significance of the co-occurrence of each pair of words in the same equivalence class. In this experiment, the window size for calculating the co-occurrence is set to two paragraphs (approximately 50-100 words). The k value for this size of window is 2.75×10^{-6} as reported in [19].
6. The Optimal Partition Algorithm (OPA) is used for clustering within the equivalence classes with the EM score. In order to measure the drawbacks of keeping a and b in the same class, Xu and Croft have proposed using a constant $\delta = 0.0075$ where the net benefit of keeping a and b is $EM(a, b) - \delta$. In this way, OPA improves the *recall* measure while preserving the *precision* measure. The OPA algorithm refines each cluster by partitioning and keeping only very related words in the same partition.
7. We have also experimented with combining the EM(a,b) and the Dice similarity measures. So, in some experiments we calculated a new matrix using the mean of $Dice$ and EM measures.

$$SEM = \frac{S_{ab} + EM(a, b)}{2} \tag{3}$$

For those experiments two new sets of equivalence classes are constructed using $t = 0.5$ and $t = 0.6$ thresholds.

Table 2 illustrates the conflations generated for the word معلومات using Dice distance, EM and SEM average with $t = 0.5$.

Table 2. Conflations for the word معلومات based on different measures.

N-grams: معل, طو , لوم, وما, مات		
DiceDistance	*EM*	*SEM*
ومعلومات, معلوماتنا, معلومات, للمعلومات, لمعلومات, المعلوماتى, معلوما, معلومه, المعلوماتيه, بمعلومات, لمعلوماتها, بالمعلومات, معلوماتك, معلوماته, معلوماتى, معلوماتها, والمعلومات, المعلومات	ومعلومات, معلوماتنا, معلومات, للمعلومات, المعلوماتى, المعلوماتيه, بمعلومات, لمعلوماتها, بالمعلومات, معلوماتك, معلوماتها, والمعلومات, المعلومات	ومعلومات, معلوماتنا, معلومات, للمعلومات, لمعلومات, المعلوماتى, معلوما, معلومه, المعلوماتيه, بمعلومات, لمعلوماتها, بالمعلومات, معلوماتك, معلوماته, معلوماتى, معلوماتها, والمعلومات, المعلومات

5 Experiments

We have used a portion of INFILE 2009 Arabic text collection for running our experiments. This collection contains 100,000 Arabic newswires from Agence France Presse (AFP) for the years 2004, 2005 and 2006. There are also 50 queries (30 general queries about sport, international affair, politics, etc and 20 scientific and technology related queries). All the documents and queries are in xml format consisting of headline, keyword and description tags. This corpus is used because of the diversity in the documents and queries. In these experiments, due to limited computational power and memory issues, only 10% of the corpus is used for generating the equivalence classes but the information retrieval experiments are conducted on the whole collection. We used python for processing the XML files and working with matrix using Numpy and Scipy plug-ins. The Java Lucene is used as the default search engine for all runs. The TREC Eval tool [18] is used for evaluating the search results and calculating the *recall* and *precision*.

Table 3 reports the characteristics of the eight different sets of the conflation classes that have been generated for these experiments. Although it has been stated in [13] that large number of Arabic words in any corpus is unique, a large number of these words can be conflated with at least one other word using the Dice distance or the average of the *Dice* and *EM* measures. However, when using the *EM* measure, more precise classes are generated which might sometimes lead to having only one word in most of the clusters. The first three runs in Table

Table 3. Description of the eight different equivalence classes.

Experiment (trigram is used for all)	t	#of Words	#of Clusters with more than 1 word	#of words in the largest cluster	Average #of words in clusters
*Dice*0.5	0.5	59,251	12,069	59 (1 *cluster*)	2.68
*Dice*0.6	0.6	69,265	14,440	51 (1 *cluster*)	2.44
*Dice*0.7	0.7	69,265	15,002	43 (1 *cluster*)	1.79
*EM*0.5	0.5	59,251	3,259	23 (2 *clusters*)	1.13
*EM*0.6	0.6	59,251	3,116	23 (1 *cluster*)	1.19
*EM*0.7	0.7	59,251	2,781	17 (1 *cluster*)	1.44
*SEM*0.5	0.5	59,251	12,069	59 (1 *cluster*)	4.9
*SEM*0.6	0.6	55,413	14,439	51 (1 *cluster*)	3.8

3 (*Dice*0.5, *Dice*0.6 and *Dice*0.7) are single stage runs. In these runs, the Dice distance and Complete Linkage clustering algorithm with different thresholds (0.5, 0.6 and 0.7) were used to create the equivalence classes which were later used in stemming the entire corpus. All of the queries were used for retrieval and the precision, recall and precision at document cut-off measures were calculated for each run.

The other runs are two stage runs as described in *Section*4. The next three runs (*EM*0.5, *EM*0.6 and *EM*0.7) applied the Optimal Partition Clustering

algorithm and EM measure on the equivalence classes generated from the first stage with different thresholds. The last two runs ($SEM0.5$, $SEM0.6$) applied the average $Dice$ and EM measures along with the OPA clustering algorithm. In order to get a better understanding of the performance of the proposed meth-

Table 4. The precision of 8 experiments at different document cutoffs.

Cut-off	$Dice0.5$	$Dice0.6$	$Dice0.7$	$EM0.5$	$EM0.6$	$EM0.7$	$SEM0.5$	$SEM0.7$
5	0.32	0.312	0.328	0.328	0.336	0.316	0.32	0.316
10	0.282	0.286	0.294	0.29	0.294	0.298	0.282	0.288
15	0.26	0.258	0.264	0.28	0.265	0.266	0.26	0.261
20	0.242	0.244	0.245	0.264	0.257	0.243	0.242	0.246
30	0.213	0.222	0.218	0.246	0.232	0.226	0.213	0.222
100	0.118	0.119	0.120	0.123	0.120	0.118	0.118	0.119
200	0.069	0.070	0.071	0.071	0.070	0.067	0.069	0.070
500	0.033	0.033	0.032	0.032	0.031	0.031	0.033	0.033
1000	0.017	0.017	0.017	0.017	0.017	0.017	0.017	0.017

ods, we created two extra runs. One run used the Light-10 stemmer [14] for stemming the Arabic words and another run applied no stemming at all. Table 5 shows a comparison of these two runs with $EM0.6$ run which is the best run in Table 4. By analyzing these results one can conclude that using any sort of stemming technique improves the precision by at least 50%. It can also be inferred that the precisions obtained using EM based clustering are very close to those obtained using the Light-10 stemmer. This implies that although the numbers of conflation classes with more than one word are not many, they still had a positive impact. These results also show that mere statistical analysis produced results comparable to stemming with linguistic knowledge. Figure 1 depicts the

Table 5. Comparison of EM0.6 run with Light-10 stemming and no stemming runs.

Cut-off	no stem	Light-10	EM0.6
5	0.061	0.34	0.336
10	0.063	0.336	0.294
15	0.057	0.302	0.265
20	0.055	0.279	0.257
30	0.053	0.254	0.232
100	0.034	0.143	0.120
200	0.022	0.083	0.070
500	0.011	0.037	0.031
1000	0.00	0.020	0.017

precision recall graph for the top3 runs along with Light-10 and no-stemming

runs. As shown in Tables 4 and 5 and Figure 1, the Light-10 stemmer which uses linguistic knowledge is the best run. Most runs are similar to each other. However, the results from $EM0.6$ run are very close to those of the Light-10 stemmer. It is safe to say that the EM measure in the second stage is necessary for eliminating erroneous conflations. It is also clear that using the Dice measure alone produces clusters that have disimilar conflated words. Since only 10% of

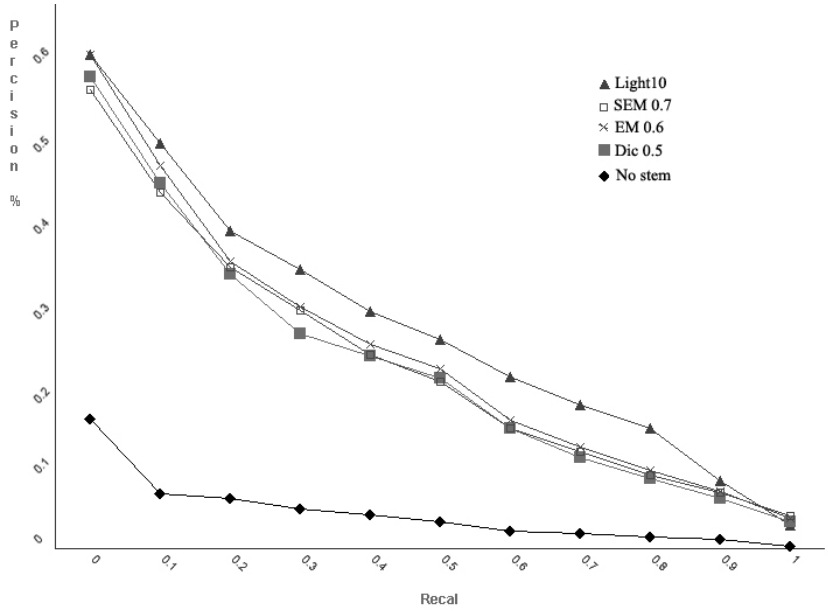

Fig. 1. Precision-Recall graph comparing our top 3 experiments with Light-10 and no-stemming.

the documents are used in generating the equivalence classes, it seems reasonable to believe that the result of $EM0.6$ will reach or exceed Light-10 if a higher percentage of the text is used.

6 Conclusion and Future Work

In this paper, we have successfully modified the corpus-based stemming proposed by XU and Croft to be used for Arabic text. We generated many different variations of our approach and compared them to the Light-10 stemming which used linguistic knowledge and no stemming approaches. Our comparison was based on precision, recall and precision at document cut-off values of retrieving 50 standard queries on a large text collection. The experiments show that using stemming without any linguistic knowledge can perform less than but comparable to well known approaches based on the morphological analysis. In our

approach, we used one stage and two stage models and our findings indicate that the second stage co-occurrence analysis is necessary to improve the conflation classes and weed out incorrect groupings of the first stage. It was also noticed that using trigram reduces the chance of word conflation and results in the construction of many single word clusters. Therefore, it is possible that bigrams will perform better by conflating more words and reducing the number of clusters with only one word. As part of the future work, we will use bigrams and hexagrams on the same corpus in order to investigate the effects of the length of roots in the Arabic language. Another future goal of ours is to improve our best performer ($EM0.6$) method with some linguistic knowledge. In this new approach, we will use a few clues to even further refine the equivalence classes produced. This refinement will be in the form of post processing and will include removing some words from equivalence classes or combining some the classes into larger units. We also intend to use only minimum morphological analysis in this new approach. We can also look into the implications behind other distance and similarity measures and different threshold values.

References

1. AleAhmad, A. , Hakimian, P. , Oroumchian, F. : N-Gram And Local Context Analysis For Persian Text Retrieval, In: International Symposium on Signal Processing and its Applications (ISSPA2007), Sharjah, pp. 12-15. United Arab Emirates (February 2007)
2. 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)
3. Buckwalter, T.Q.: Arabic lexicography, `http://www.qamus.org/`
4. Chen, A., and Gey, F.: Building an Arabic stemmer for information retrieval. Proceedings of TREC 2002. Gaithersburg: NIST, pp 631-639. (2002)
5. Darwish, K. and Oard, D.W.: CLIR Experiments at Maryland for TREC-2002: Evidence combination for Arabic-English retrieval. Proceedings of TREC 2002. Gaithersburg: NIST, pp. 703-710. (2002)
6. De Roeck, A. N. and Al-Fares, W.: A morphologically sensitive clustering algorithm for identifying Arabic roots. Proceedings of the 38th Annual Meeting on Association for Computational Linguistics, ACL2000, pp. 199-206. Hong Kong (2000)
7. Diab, M., Hacioglu, K., and Jurafsky, D.: Automatic tagging of Arabic text: From raw test to base phrase chunks. In Proceedings of the Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics: HLT-NAACL (2004)
8. Ekmekcioglu, F.C., Lynch, M.F., and Willett, P.: Stemming and N-gram matching for term conflation in Turkish texts. Information Research News, 7 (1), pp. 2-6. (1996)
9. Frakes, W.B.: Stemming algorithms. In Information retrieval: Data structures and algorithms, W. B. F. a. R. Baeza-Yates, Ed. Englewood Cliffs, NJ: Prentice Hall, chapter 8. (1992)
10. Internet Word Stats, `http://www.internetworldstats.com/stats7.htm` (2010)
11. Khoja, S. and Garside, R., Stemming Arabic text. `http://zeus.cs.pacificu.edu/shereen/research.htm`

12. Khreisat, L. Arabic Text Classification Using N-Gram Frequency Statistics A Comparative Study, Proceedings of the 2006 International Conference on Data Mining. pp. 78-82. Las Vegas, USA (2006)
13. Larkey, L.S. and Connell, M.E. Arabic information retrieval at UMass. In Proceedings of TREC 2001, Gaithersburg: NIST (2001)
14. Larkey, L.S., Ballesteros, L. and Connell, M.E. Improving stemming for Arabic information retrieval: Light Stemming and co-occurrence analysis. In Proceedings of SIGIR 2002, pp. 275-282. Tampere, Finland (2002)
15. Mustafa, H.S. and Al-Radaideh, Q. : Using N-Grams for Arabic Text Searching, Journal of the American Conference on Data Mining, Society for Information Science and Technology, 55(11), pp. 1002-1007. (2004)
16. Oard, D.W., Levow, G.A., and Cabezas, C.I.: CLEF experiments at Maryland: Statistical stemming and backoff translation. In Cross-language information retrieval and evaluation: Proceedings of the CLEF2000 workshop, C. Peters, Ed.: Springer Verlag, pp. 176-187. (2001)
17. Oroumchian, F. and Garamaleki, F.M.: An Evaluation of Retrieval Performance Using Farsi Text, Workshop On Knowledge Foraging for Dynamic Networking of Communities and Economies, First Eurasia Conference on Advances in Information and Communication Technology, Shiraz, Iran (2002)
18. Trec eval Tool. `http://trec.nist.gov/trec_eval/` (2010)
19. Xu, J. and Croft, W.B.: Corpus-based stemming using co-occurrence of word variants. ACM Transactions on Information Systems, 16 (1), pp. 61-81. (1998)
20. Xu, J. , Fraser, A. and Weischedel, R.: Empirical Studies in Strategies for Arabic Retrieval. SIGIR '02 Tampere Finland, pp. 269-274. (2002)