Examining the Effect of Improved Context Sensitive Morphology on Arabic Information Retrieval

Kareem Darwish
Dept. of Information Engineering & Technology
German University in Cairo
5th District, New Cairo, Cairo, Egypt
and
IBM Technology Development Center
P.O. Box 166, El-Ahram, Giza, Egypt
kareem@darwish.org

Hany Hassan and Ossama Emam
IBM Technology Development Center
P.O. Box 166
El-Ahram, Giza, Egypt
{hanyh,emam}@eg.ibm.com

Abstract

This paper explores the effect of improved morphological analysis, particularly context sensitive morphology, on monolingual Arabic Information Retrieval (IR). It also compares the effect of context sensitive morphology to non-context sensitive morphology. The results show that better coverage and improved correctness have a dramatic effect on IR effectiveness and that context sensitive morphology further improves retrieval effectiveness, but the improvement is not statistically significant. Furthermore, the improvement obtained by the use of context sensitive morphology over the use of light stemming was not significantly significant.

1 Introduction

Due to the morphological complexity of the Arabic language, much research has focused on the effect of morphology on Arabic Information Retrieval (IR). The goal of morphology in IR is to conflate words of similar or related meanings. Several early studies suggested that indexing Arabic text using roots significantly increases retrieval effectiveness over the use of words or stems [1, 3, 11]. However, all the studies used small test collections of only hundreds of documents and the morphology in many of the studies was done manually.

Performing morphological analysis for Arabic IR using existing Arabic morphological analyzers, most of which use finite state transducers [4, 12, 13], is problematic for two reasons. First, they were designed to produce as many analyses as possible without indicating which analysis is most likely. This property of the analyzers complicates retrieval, because it introduces ambiguity in the indexing phase as well as the search phase of retrieval. Second, the use of finite state transducers inherently limits coverage, which the number of words that the analyzer can analyze, to the cases programmed into the transducers. Darwish attempted to solve this problem by developing a statistical morphological analyzer for Arabic called Sebawai that attempts to rank possible analyses to pick the most likely one [7]. He concluded that even with ranked analysis, morphological analysis did not yield statistically significant improvement over words in IR. A later study by Aljlayl et al. on a large Arabic collection of 383,872 documents suggested that lightly stemmed words, where only common prefixes and suffixes are stripped from them, were perhaps better index term for Arabic [2]. Similar studies by Darwish [8] and Larkey [14] also suggested that light stemming is indeed superior to morphological analysis in the context of IR.

Proceedings of the ACL Workshop on Computational Approaches to Semitic Languages, pages 25–30,
Ann Arbor, June 2005.

However, the shortcomings of morphology might be attributed to issues of coverage and correctness. Concerning coverage, analyzers typically fail to analyze Arabized or transliterated words, which may have prefixes and suffixes attached to them and are typically valuable in IR. As for correctness, the presence (or absence) of a prefix or suffix may significantly alter the analysis of a word. For example, for the word "Alksyr" is unambiguously analyzed to the root "ksr" and stem "ksyr." However, removing the prefix "Al" introduces an additional analysis, namely to the root "syr" and the stem "syr." Perhaps such ambiguity can be reduced by using the context in which the word is mentioned. For example, for the word "ksyr" in the sentence "sAr ksyr" (and he walked like), the letter "k" is likely to be a prefix.

The problem of coverage is practically eliminated by light stemming. However, light stemming yields greater consistency without regard to correctness. Although consistency is more important for IR applications than linguistic correctness, perhaps improved correctness would naturally yield great consistency. Lee et al. [15] adopted a trigram language model (LM) trained on a portion of the manually segmented LDC Arabic Treebank in developing an Arabic morphology system, which attempts to improve the coverage and linguistic correctness over existing statistical analyzers such as Sebawai [15]. The analyzer of Lee et al. will be henceforth referred to as the IBM-LM analyzer. IBM-LM's analyzer combined the trigram LM (to analyze a word within its context in the sentence) with a prefix-suffix filter (to eliminate illegal prefix suffix combinations, hence improving correctness) and unsupervised stem acquisition (to improve coverage). Lee et al. report a 2.9% error rate in analysis compared to 7.3% error reported by Darwish for Sebawai [7]. This paper evaluates the IBM-LM analyzer in the context of a monolingual Arabic IR application to determine if in-context morphology leads to improved retrieval effectiveness compared to out-of-context analysis. To determine the effect of improved analysis, particularly the use of in-context morphology, the analyzer is used to produce analyses of words in isolation (with no context) and in-context. Since IBM-LM only produces stems, Sebawai was used to produce the roots corresponding to the stems produced by IBM-LM. Both are compared to Sebawai and light stemming.

The paper will be organized as follows: Section 2 surveys related work; Section 3 describes the IR experimental setup for testing the IBM-LM analyzer; Section 4 presents experimental results; and Section 5 concludes the paper.

2 Related Work

Most early studies of character-coded Arabic text retrieval relied on relatively small test collections [1, 3, 9, 11]. The early studies suggested that roots, followed by stems, were the best index terms for Arabic text. More recent studies are based on a single large collection (from TREC-2001/2002) [9, 10]. The studies examined indexing using words, word clusters [14], terms obtained through morphological analysis (e.g., stems and roots [9]), light stemming [2, 8, 14], and character n-grams of various lengths [9, 16]. The effects of normalizing alternative characters, removal of diacritics and stop-word removal have also been explored [6, 19]. These studies suggest that perhaps light stemming and character n-grams are the better index terms.

Concerning morphology, some attempts were made to use statistics in conjunction with rule-based morphology to pick the most likely analysis for a particular word or context. In most of these approaches an Arabic word is assumed to be of the form prefix-stem-suffix and the stem part may or may not be derived from a linguistic root. Since Arabic morphology is ambiguous, possible segmentations (i.e. possible prefix-stem-suffix tuples) are generated and ranked based on the probability of occurrence of prefixes, suffixes, stems, and stem template. Such systems that use this methodology include RDI's MORPHO3 [5] and Sebawai [7]. The number of manually crafted rules differs from system to system. Further MORPHO3 uses a word trigram model to improve in-context morphology, but uses an extensive set of manually crafted rules. The IBM-LM analyzer uses a trigram language model with a minimal set of manually crafted rules [15]. Like other statistical morphology systems, the IBM-LM analyzer assumes that a word is constructed as prefix-stem-suffix. Given a word, the analyzer generates all possible segmentations by identifying all matching prefixes and suffixes from a table of

prefixes and suffixes. Then given the possible segmentations, the trigram language model score is computed and the most likely segmentation is chosen. The analyzer was trained on a manually segmented Arabic corpus from LDC.

3 Experimental Design

IR experiments were done on the LDC LDC2001T55 collection, which was used in the Text REtrieval Conference (TREC) 2002 cross-language track. For brevity, the collection is referred to as the TREC collection. The collection contains 383,872 articles from the Agence France Press (AFP) Arabic newswire. Fifty topics were developed cooperatively by the LDC and the National Institute of Standards and Technology (NIST), and relevance judgments were developed at the LDC by manually judging a pool of documents obtained from combining the top 100 documents from all the runs submitted by the participating teams to TREC's cross-language track in 2002. The number of known relevant documents ranges from 10 to 523, with an average of 118 relevant documents per topic [17]. This is presently the best available large Arabic information retrieval test collection. The TREC topic descriptions include a title field that briefly names the topic, a description field that usually consists of a single sentence description, and a narrative field that is intended to contain any information that would be needed by a human judge to accurately assess the relevance of a document [10]. Queries were formed from the TREC topics by combining the title and description fields. This is intended to model the sort of statement that a searcher might initially make when asking an intermediary, such as a librarian, for help with a search.

Experiments were performed for the queries with the following index terms:

- w: words.
- ls: lightly stemmed words, obtained using Al-Stem [17][1].
- SEB-s: stems obtained using Sebawai.
- SEB-r: roots obtained using Sebawai.
- cIBM-LMS: stems obtained using the IBM-LM analyzer in context. Basically, the entire TREC collection was processed by the analyzer and the prefixes and suffixes in the segmented output were removed.
- cIBM-SEB-r: roots obtained by analyzing the in-context stems produced by IBM-LM using Sebawai.
- IBM-LMS: stems obtained using the IBM-LM analyzer without any contextual information. Basically, all the unique words in the collection were analyzed one by one and the prefixes and suffixes in the segmented output were removed.
- IBM-SEB-r: roots obtained by analyzing the out-of-context stems produced by IBM-LM using Sebawai.

All retrieval experiments were performed using the Lemur language modeling toolkit, which was configured to use Okapi BM-25 term weighting with default parameters and with and without blind relevance feedback (the top 20 terms from the top 5 retrieved documents were used for blind relevance feedback). To observe the effect of alternate indexing terms mean uninterpolated average precision was used as the measure of retrieval effectiveness. To determine if the difference between results was statistically significant, a Wilcoxon signed-rank test, which is a nonparametric significance test for correlated samples, was used with p values less than 0.05 to claim significance.

4 Results and Discussion

Figure 1 shows a summary of the results for different index terms. Tables 1 and 2 show statistical significance between different index terms using the p value of the Wilcoxon test. When comparing index terms obtained using IBM-LM and Sebawai, the results clearly show that using better morphological analysis produces better retrieval effectiveness. The dramatic difference in retrieval effectiveness between Sebawai and IBM-LM highlight the effect of errors in morphology that lead to inconsistency in analysis. When using contextual information in analysis (compared to analyzing words in isolation – out of context) resulted in only a 3% increase in mean average precision when using stems (IBM-LMS), which is a small difference compared to the

[1] A slightly modified version of Leah Larkey's Light-10 light stemmer [8] was also tried, but the stemmer produced very similar results to Al-Stem.

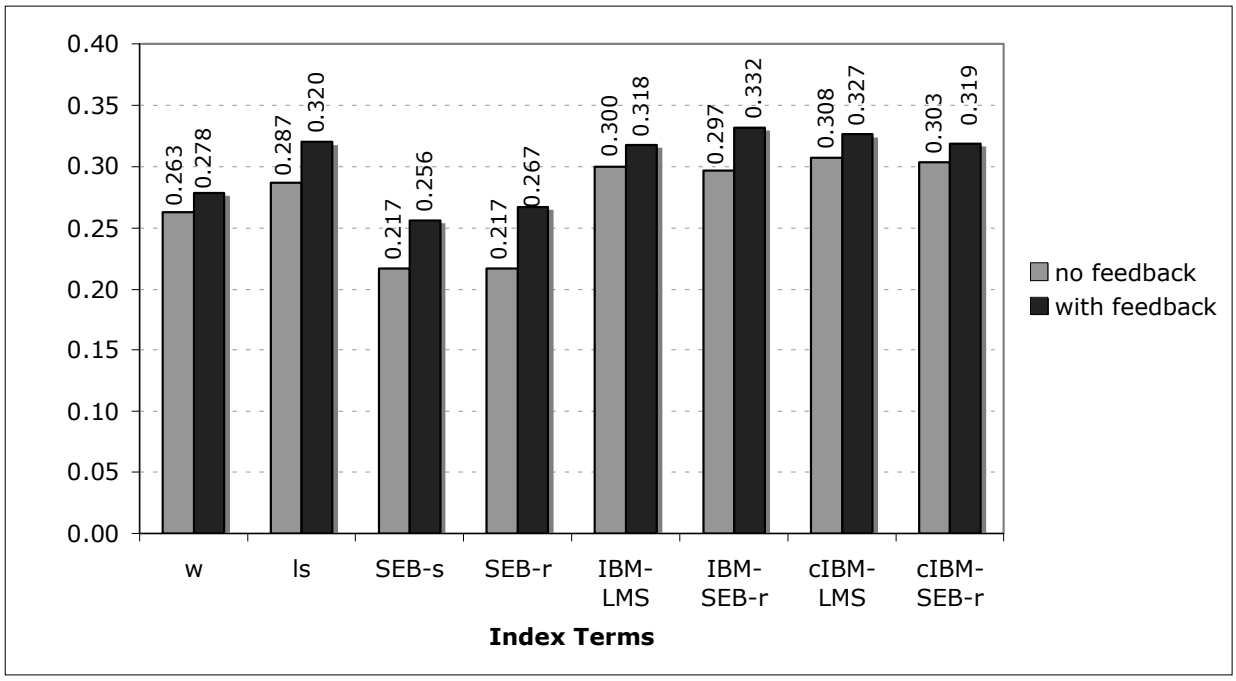

Figure 1. Comparing index term with and without blind relevance feedback using mean average precision

effect of blind relevance feedback (about 6% increase) and produced mixed results when using roots (IBM-SEB-r). Nonetheless, the improvement for stems was almost statistically significant with *p* values of 0.063 and 0.054 for the cases with and without blind relevance feedback. Also considering that improvement in retrieval effectiveness resulted from changing the analysis for only 0.12% of the words in the collection (from analyzing them out of context to analyzing them in context)[2] and that the authors of IBM-LM report about 2.9% error rate in morphology, perhaps further improvement in morphology may lead to further improvement in retrieval effectiveness. However, further improvements in morphology and retrieval effectiveness are likely to be difficult. One of difficulties associated with developing better morphology is the disagreement on what constitutes "better" morphology. For example, should "mktb" and "ktb" be conflated? "mktb" translates to office, while ktb translates to books. Both words share the common root "ktb," but they are not interchangeable in meaning or usage. One would expect that increasing conflation would improve recall at the expense of precision and decreasing conflation would have the exact opposite effect. It is known that IR is more tolerant of over-conflation than under-conflation [18]. This fact is apparent in the results when comparing roots and stems. Even though roots result in greater conflation than stems, the results for stems and roots are almost the same. Another property of IR is that IR is sensitive to consistency of analysis. In the case of light stemming, stemming often mistakenly removes prefixes and suffixes leading to over conflation, for which IR is tolerant, but the mistakes are done in a consistent manner. It is noteworthy that sense disambiguation has been reported to decrease retrieval effectiveness [18]. However, since improving the correctness of morphological analysis using contextual information is akin to sense disambiguation, the fact that retrieval results improved, though slightly, using context sensitive morphology is a significant result.

In comparing the IBM-LM analyzer (in context or out of context) to light stemming (using Al-Stem), although the difference in retrieval effectiveness is small and not statistically significant, using the IBM-LM analyzer, unlike using Al-Stem, leads to

[2] Approximately 7% of unique tokens had two or more different analysis in the collection when doing in-context morphology. In tokens with more than one analysis, one of the analyses was typically used more than 98% of the time.

ls	SEB-s	SEB-r	IBM-LMS	IBM-SEB-r	cIBM-LMS	cIBM-SEB-r	
0.055	0.475	0.671	0.038	0.027	0.019	0.049	w
	0.004	0.023	0.560	0.359	0.946	0.505	ls
		0.633	0.005	0.001	0.001	0.012	SEB-s
			0.039	0.007	0.020	0.064	SEB-r
				0.0968	0.063	0.758	IBM-LMS
					0.396	0.090	IBM-SEB-r
						0.001	cIBM-LMS

Table 1. Wilcoxon *p* values (shaded=significant), with blind relevance feedback.

ls	SEB-s	SEB-r	IBM-LMS	IBM-SEB-r	cIBM-LMS	cIBM-SEB-r	
0.261	0.035	0.065	0.047	0.135	0.011	0.016	w
	0.000	0.000	0.968	0.757	0.515	0.728	ls
		0.269	0.000	0.000	0.000	0.000	SEB-s
			0.000	0.000	0.000	0.000	SEB-r
				0.732	0.054	0.584	IBM-LMS
					0.284	0.512	IBM-SEB-r
						0.005	cIBM-LMS

Table 2. Wilcoxon *p* values (shaded=significant), without blind relevanc e feedback

statistically significant improvement over using words. Therefore there is some advantage, though only a small one, to using statistical analysis over using light stemming. The major drawback to morphological analysis (specially in-context analysis) is that it requires considerably more computing time than light stemming[3].

5 Conclusion

The paper investigated the effect of improved morphological analysis, especially context sensitive morphology, in Arabic IR applications compared to other statistical morphological analyzers and light stemming. The results show that improving morphology has a dramatic effect on IR effectiveness and that context sensitive morphology slightly improved Arabic IR over non-context sensitive morphology, increasing IR effectiveness by approximately 3%. The improvement is almost statistically significant. Developing better morphology could lead to greater retrieval effectiveness, but improving analyzers is likely to be difficult and would require careful determination of the proper level of conflation. In overcoming some of the difficulties associated with obtaining "better" morphology (or more fundamentally the proper level of word conflation), adaptive morphology done on a per query term basis or user feedback might prove valuable. Also, the scores that were used to rank the possible analyses in a statistical morphological analyzer may prove useful in further improving retrieval. Other IR techniques, such as improved blind relevance feedback or combination of evidence approaches, can also improve monolingual Arabic retrieval.

Perhaps improved morphology is particularly beneficial for other IR applications such as cross-language IR, in which ascertaining proper translation of words is particularly important, and

[3] The processing of the TREC collection using the in-context IBM-LM required 16 hours on a 2.4 GHz Pentium 4 machine with 1 Gigabyte of RAM compared to 10 minutes to perform light stemming.

in-document search term highlighting for display to a user.

References

1. Abu-Salem, H., M. Al-Omari, and M. Evens. Stemming Methodologies Over Individual Query Words for Arabic Information Retrieval. JASIS, 1999. 50(6): p. 524-529.

2. Aljlayl, M., S. Beitzel, E. Jensen, A. Chowdhury, D. Holmes, M. Lee, D. Grossman, and O. Frieder. IIT at TREC-10. In TREC. 2001. Gaithersburg, MD.

3. Al-Kharashi, I. and M Evens. Comparing Words, Stems, and Roots as Index Terms in an Arabic Information Retrieval System. JASIS, 1994. 45(8): p. 548 - 560.

4. Antworth, E. PC-KIMMO: a two-level processor for morphological analysis. In Occasional Publications in Academic Computing. 1990. Dallas, TX: Summer Institute of Linguistics.

5. Ahmed, Mohamed Attia. A Large-Scale Computational Processor of the Arabic Morphology, and Applications. A Master's Thesis, Faculty of Engineering, Cairo University, Cairo, Egypt, 2000.

6. Chen, A. and F. Gey. Translation Term Weighting and Combining Translation Resources in Cross-Language Retrieval. In TREC, 2001. Gaithersburg, MD.

7. Darwish, K. Building a Shallow Morphological Analyzer in One Day. ACL Workshop on Computational Approaches to Semitic Languages. 2002.

8. Darwish, K. and D. Oard. CLIR Experiments at Maryland for TREC 2002: Evidence Combination for Arabic-English Retrieval. In TREC. 2002. Gaithersburg, MD.

9. Darwish, K. and D. Oard. Term Selection for Searching Printed Arabic. SIGIR, 2002. Tampere, Finland. p. 261 - 268.

10. Gey, F. and D. Oard. The TREC-2001 Cross-Language Information Retrieval Track: Searching Arabic Using English, French or Arabic Queries. TREC, 2001. Gaithersburg, MD. p. 16-23.

11. Hmeidi, I., G. Kanaan, and M. Evens. Design and Implementation of Automatic Indexing for Information Retrieval with Arabic Documents. JASIS, 1997. 48(10): p. 867 - 881.

12. Kiraz, G. Arabic Computation Morphology in the West. In The 6th International Conference and Exhibition on Multi-lingual Computing. 1998. Cambridge.

13. Koskenniemi, K., Two Level Morphology: A General Computational Model for Word-form Recognition and Production. 1983, Department of General Linguistics, University of Helsinki.

14. Larkey, L., L. Ballesteros, and M. Connell. Improving Stemming for Arabic Information Retrieval: Light Stemming and Co-occurrence Analysis. SIGIR 2002. p. 275-282, 2002.

15. Lee, Y., K. Papineni, S. Roukos, O. Emam, and H. Hassan. Language Model Based Arabic Word Segmentation. In the Proceedings of the 41st Annual Meeting of the Association for Computational Linguistics, July 2003, Sapporo, Japan. p. 399 - 406.

16. Mayfield, J., P. McNamee, C. Costello, C. Piatko, and A. Banerjee. JHU/APL at TREC 2001: Experiments in Filtering and in Arabic, Video, and Web Retrieval. In TREC 2001. Gaithersburg, MD. p. 322-329.

17. Oard, D. and F. Gey. The TREC 2002 Arabic/English CLIR Track. In TREC 2002. Gaithersburg, MD.

18. Sanderson, M. Word sense disambiguation and information retrieval. In Proceedings of the 17th ACM SIGIR Conference, p. 142-151, 1994

19. Xu, J., A. Fraser, and R. Weischedel. 2001 Cross-Lingual Retrieval at BBN. In TREC, 2001. Gaithersburg, MD. p. 68 - 75.