Induction of Root and Pattern Lexicon for Unsupervised Morphological Analysis of Arabic

Bilal Khaliq
Dept of Informatics, University of Sussex
Brighton BN1 9QJ, UK
bk54@sussex.ac.uk

John Carroll
Dept of Informatics, University of Sussex
Brighton BN1 9QJ, UK
J.A.Carroll@sussex.ac.uk

Abstract

We propose an unsupervised approach to learning non-concatenative morphology, which we apply to induce a lexicon of Arabic roots and pattern templates. The approach is based on the idea that roots and patterns may be revealed through mutually recursive scoring based on hypothesized pattern and root frequencies. After a further iterative refinement stage, morphological analysis with the induced lexicon achieves a root identification accuracy of over 94%. Our approach differs from previous work on unsupervised learning of Arabic morphology in that it is applicable to naturally-written, unvowelled text.

1 Introduction

Manual development of morphological analysis systems is expensive. It is impractical to develop morphological descriptions for more than a very small proportion of human languages. In recent years a number of approaches have been proposed that learn the morphology of a language from unannotated text. The Morpho Challenge and similar competitions have further motivated researchers to devise techniques for unsupervised learning of language.

Previous work in unsupervised morphology learning has mostly addressed concatenative morphology, in which surface word forms are sequentially separated or segmented into morpheme units. However, some languages (in particular Semitic languages) have another type of word formation in which morphemes combine in a non-concatenative manner, through the interdigitation of a root morpheme with an affix or pattern template. Unsupervised learning of non-concatenative morphology has received comparatively little attention.

In this paper we describe a conceptually simple yet effective unsupervised approach to learning non-concatenative morphology. We apply our approach to inducing an Arabic lexicon of trilateral roots and pattern templates. Lexicon acquisition is based on the idea that roots and affix patterns may be revealed by their converses, i.e. roots are identified from occurrences of patterns and conversely patterns are recognized from root frequencies. Subsequently, the lexicons are iteratively improved by refining the morpheme strengths computed in the previous step.

The paper is organized as follows. We survey previous related work (Section 2), and then give a brief introduction to Arabic root and pattern morphology (Section 3). We explain our basic technique for unsupervised lexicon induction in Section 4, followed by the refinement procedure (Section 5). Section 6 describes how the lexicon is used for morphological analysis. Finally, we present an evaluation (Section 7) and conclusions (Section 8).

2 Related Work

Beesley (1996) describes one of the first morphological analysis systems for Arabic, based on finite-state techniques with manually acquired lexicons and rules. This kind of approach, although potentially producing an efficient and accurate system, is expensive in time and linguistic expertise, and lacks robustness in terms of extendibility to word types not in the dictionary (Ahmed, 2000).

Darwish (2002) describes a semi-automatic technique that learns morphemes and induces rules for deriving stems using an existing dictionary of word and root pairs. It is an easy to build and fairly robust method of performing morphological analysis. Clark (2007) investigates semi-supervised learning using the

International Joint Conference on Natural Language Processing, pages 1012–1016,
Nagoya, Japan, 14-18 October 2013.

complex broken plural structure of Arabic as a test case. He employs memory-based algorithms, with the aim of gaining insights into human language acquisition.

Other researchers have applied statistical and information-theoretic approaches to unsupervised learning of morphology from raw (unannotated) text corpora. Goldsmith (2000, 2006) and Cruetz and Lagus (2005, 2007) use the Minimum Description Length (MDL) principle, considering input data to be 'compressed' into a morphologically analysed representation. An alternative perspective adopted by Schone and Jurafsky (2001) induces semantic relatedness between word pairs by Latent Semantic Indexing.

Most work on unsupervised learning of morphology has focused on concatenative morphology (Hammarström and Borin, 2011). Studies that have focussed on non-concatenative morphology include that of Rodriguez and Ćavar (2005), who learn roots from artificially generated text using a number of orthographic heuristics, and then apply constraint-based learning to improve the quality of the roots. Xanthos (2008) deciphers roots and patterns from phonetic transcriptions of Arabic text, using MDL to refine the root and pattern structures.

Our work differs from these previous approaches in that (1) we learn intercalated morphology, identifying the root and transfixes/ incomplete pattern for words, and (2) we start from 'natural' text without short vowels or diacritical markers.

3 Root and Pattern Morphology

Words in Arabic are formed through three morphological processes: (i) fusion of a root form and pattern template to derive a base word; (ii) affixation, including inflectional morphemes marking gender, plurality and/or tense, resulting in a stem; and (iii) possible attachment of a final layer of clitics. Our work addresses the first two of these processes.

As an example of word formation in Arabic, the word *ktAby* is formed from the root *Ktb* and the pattern *--A-y*, where *y* is an inflectional marker and *A* is the derivational infix marker for nouns.

During analysis, we decompose each word w into a set of tuples encoding all k possible combinations of a root (of at least 3 letters) and associated pattern (Eq. 1)

$$d(w) \rightarrow \{\langle r^x, p^x \rangle\} \quad \text{(Eq. 1)}$$

where x ranges from 1 to k. For example, the decomposition of the word *yErf* is shown in Figure 1.

$$yErf \rightarrow \begin{Bmatrix} \langle y\ E\ r, & - - - f \rangle, \\ \langle y\ E\ f, & - - r - \rangle, \\ \langle y\ r\ f, & - E - - \rangle, \\ \langle E\ r\ f, & y - - - \rangle, \\ \langle y\ E\ r\ f, & - - - - \rangle \end{Bmatrix}$$

Figure 1. Decomposition of a word into all possible combinations of roots and patterns.

4 Building Lexicons Using Contrastive Scoring

Based on the idea that roots and patterns may be revealed by their converses, we score a pattern based on the frequency of occurrence of the roots, and score a root according to the number of occurrences of patterns. We score each morpheme and then rescore it weighted by previous scores. Our technique resembles the *hubs and authorities* algorithm originally devised for rating Web pages (Kleinberg, 1999), which assigns to each Web page two scores: its hub value and its authority. These two values are updated in a similar mutually recursive manner as we describe for roots and patterns.

4.1 Frequency-Based Scoring

The initial scoring function is simple: firstly, we aggregate over the number of occurrences of a root radical sequence in a word w_i, for words $i=1,2,\ldots N$ in the input dataset. The function for scoring each pattern in the target word, *t*, is given in equation (Eq. 2).

$$S(p_t^x) = \sum_{i=1}^{N} \left(1 \mid r_t^x = r_{w_i}^y \right) \quad \text{(Eq. 2)}$$

The function for scoring the root, r_t^x, in each target word, *t*, with pattern, p_t^x, is given in equation (Eq. 3).

$$S(r_t^x) = \sum_{i=1}^{N} \left(1 \mid p_t^x = p_{w_i}^y \right) \quad \text{(Eq. 3)}$$

We choose this as our baseline, to which we compare subsequent enhancements.

4.2 Scaling

Since pattern strength is computed based on root occurrence frequency and vice-versa, each pattern and root has a different score range due to the distinct distributions of patterns and roots. In order to make the scores comparable and contribute equally, we scale the scores in one lexicon with respect to the other.

We take the pattern lexicon as reference and scale each root, r_u (u=1,2,…R entries in root lexicon) by the ratio of the maximum pattern score to the maximum root score:

$$SS(r_u) = S(r_u)(\times \frac{\max(S(p))}{\max(S(r))} \quad \text{(Eq. 4)}$$

4.3 Iterative Rescoring

Having obtained initial scores for the root and pattern lexicons, they are improved through an iterative rescoring process. We rescore each morpheme lexicon in a similar manner to equations (Eq. 2) and (Eq. 3), but weighted with the normalized score for each morpheme of previous scores. This is an iterative process starting with the initial score, S_0, calculated using frequency counts (as in section 4.1). Let S_j be the new score based on previous scores, and scaled scores, S_{j-1} and SS_{j-1}, respectively, for iterations $j=0,1,2,\ldots n$,

$$S_j(r_t^x) = \sum_{i=1}^{N} \left(S_{j-1}(p_t^x)/\max(S_{j-1}) \;\middle|\; p_t^x = p_{w_i}^y \right) \quad \text{(Eq. 5)}$$

$$S_j(p_t^x) = \sum_{i=1}^{N} \left(SS_{j-1}(r_t^x)/\max(SS_{j-1}) \;\middle|\; r_t^x = r_{w_i}^y \right) \quad \text{(Eq. 6)}$$

Here we have normalized the score with respect to the maximum value for the reference pattern lexicon, thus keeping the magnitude of the rescored value in range.

5 Refinement

The refinement phase considers the overall strength of occurrence of each morpheme in the vocabulary. Thus, if a certain root morpheme is a true morpheme then all the pattern morphemes it occurs with would have higher scores since they also would be true morphemes. In such a case, this phase would increase the overall average strength for the root. The scores obtained from the frequency-based method (Section 4) are frequency counts or weighted frequency counts. The scoring and rescoring in this refinement step differs in that it evaluates each root by averaging over scores of the corresponding patterns it occurs with in the dataset. We again iteratively refine based on the previous scores for $k=0,1,2,\ldots m$ iterations,

$$S_j(r_t^x) = \frac{1}{f_r} \sum_{i=1}^{N} \left(S_{k-1}(p_{w_i}) \middle| r_t^x = r_{w_i} \right) \quad \text{(Eq. 7)}$$

where f_r is the number of words with root r, from a total of N vocabulary words. Similarly, for the pattern rescoring with the best so far pattern, p_w^b,

$$S_j(p_w^x) = \frac{1}{f_p} \sum_{i=1}^{N} \left(S_{k-1}(r_{w_i}) \middle| p_w^x = p_{w_i} \right) \quad \text{(Eq. 8)}$$

Here we sum over the score of counterpart morphemes based on the match of the target morpheme in a vocabulary word, unlike the rescoring step, where we match the corresponding roots.

6 Morphological Analysis

A word, w_i, is analysed into its potential root and pattern template by considering every possible combination of trilateral root and corresponding pattern pairs, $\langle r^x, p^x \rangle$, as defined in equation (Eq. 1). Each analysis is scored with the sum of the scores for the root, r^x, and pattern, p^x, in the root lexicon and pattern lexicon, respectively. While combining scores we again apply scaling as in equation (Eq. 4) in order to guarantee equal contributions from each morpheme. The analysis, x, with the highest score, as calculated in equation (Eq. 9), is selected and is output.

$$\max_{x=1..n} \left(S(r_w^x) + SS(p_w^x) \right) \quad \text{(Eq. 9)}$$

Since we are considering text without diacritics, due to the absence of short vowels, we only expect words to contain single letter infixes. Hence we also experiment with an alternative configuration of the word decomposition, $\langle r^z, p^z \rangle$ in which only those tuples with single character infixes in patterns are considered for analysis, and all other tuples are dropped. We refer to this configuration as 'IF1' in the following evaluation.

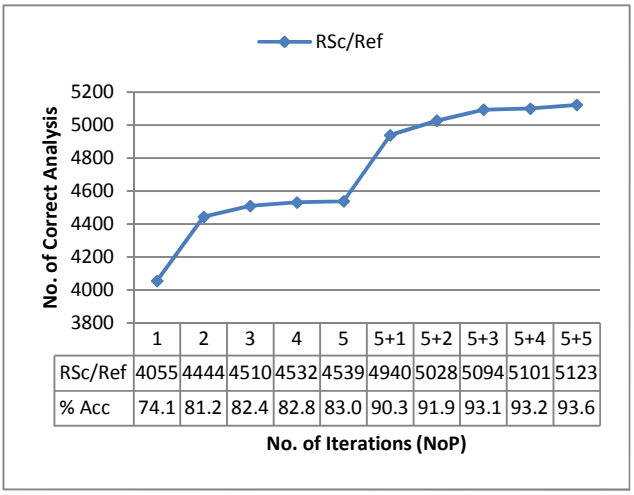
7 Evaluation

The evaluation dataset comes from the Quranic Arabic Corpus (QAC),[1] which contains approximately 7370 undiacritized, stemmed token types. Although for evaluation purposes we use the stemmed vocabulary provided by QAC, such stemmed words could be obtained using existing techniques for unsupervised concatenative morphology learning (e.g. Poon *et al.*, 2009).

More than 7192 words (95% of the total vocabulary) are tagged with their root forms since the Quran consists mostly of words of derivable forms, with very few proper nouns. Sometimes alterations in root radicals take place, for example, in hollow roots, when moving from a root containing a long vowel to the surface word, the long vowel might change its form to another type or get dropped. Such words with hollow roots or reduplicated radicals, whose characters do not match every radical of the root, were excluded from the evaluation as they are beyond the scope of the learning algorithm. After these exclusions, 5468 word and root evaluation pairs remain.

7.1 Root Identification

We evaluate morphological analysis through correct identification of the root. Accuracy is measured in terms of the percentage of roots that are correctly identified.



	1	2	3	4	5	5+1	5+2	5+3	5+4	5+5
RSc/Ref	4055	4444	4510	4532	4539	4940	5028	5094	5101	5123
% Acc	74.1	81.2	82.4	82.8	83.0	90.3	91.9	93.1	93.2	93.6

Figure 2. Results for iterative scoring and refinement.

Using the initial frequency-based scoring function (Section 4.1), we obtain a baseline (BL) accuracy of 74.1%. Figure 2 shows the results of the rescoring (RSc) and refinement (Ref) phases, with n=5 and m=5 iterations respectively (NoP n+m). There is a sudden improvement in accuracy after the first rescoring phase, and gradual improvement thereafter until the fifth. The refinement phase shows a similar trend, with a sudden improvement in accuracy at NoP 5+1. Here too the improvement is more gradual after each further iteration.

Configuration	Total Correct	Percentage Correct (%)
Baseline (BL)	4055	74.2
RSc_NoP1	4444	81.2
RSc_NoP5	4539	83.0
RSc_Ref_NoP5+1	4940	90.3
RSc_Ref_NoP5+5	5123	93.6
RSc_Ref_IF1	5159	**94.3**

Table 1. Results at key stages.

Table 1 shows the number of correct results at key stages. Rescoring and refinement each improve accuracy by 7 percentage points on their first iteration. This shows the advantage of using weighted morpheme scores. The subsequent iterations give total improvements of approximately 3 points. The IF1 configuration yields a further improvement of 0.75 points, indicating that some irrelevant analyses have been filtered out. With all the enhancements, the overall accuracy of 94.3% is an improvement of more than 20 percentage points over the baseline.

8 Conclusions and Future Directions

We have presented a novel, unsupervised approach to learning non-concatenative morphology. The approach learns trilateral roots and pattern templates, based on the idea that each may be revealed by their converses, using a mutually recursive scoring method. A subsequent refinement phase further increases accuracy.

The approach could be extended to roots beyond trilateral by adapting the scoring function to accommodate for morpheme length. In the future, we intend to apply the method to learning other kinds of morphological structures.

Acknowledgments
We thank the referees for valuable comments, and in particular pointing out the correspondence to the hubs and authorities algorithm.

[1] http://corpus.quran.com/

References

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

Kenneth Beesley. 1996. Arabic finite-state morphological analysis and generation. In *Proceedings of the 16th International Conference on Computational Linguistics (COLING*), Copenhagen, Denmark, 89-94.

Alexander Clark. 2007. Supervised and unsupervised learning of Arabic morphology. *Arabic Computational Morphology*, volume 38 of *Text, Speech and Language Technology*, pages 181-200.

Mathias Creutz and Krista Lagus. 2005. Inducing the morphological lexicon of a natural language from unannotated text. In *Proceedings of the International and Interdisciplinary Conference on Adaptive Knowledge Representation and Reasoning* (AKRR '05), 106-113.

Mathias Creutz and Krista Lagus. 2007. Unsupervised models for morpheme segmentation and morphology learning. *ACM Transactions on Speech and Language Processing*, 4(1-3):1-33.

Kareem Darwish. 2002. Building a shallow morphological analyzer in one day. In *Proceedings of the ACL-02 Workshop on Computational Approaches to Semitic Languages*, Philadelphia, PA, 1-8.

Guy De Pauw and Peter Wagacha. 2007. Bootstrapping morphological analysis of Gikuyu using unsupervised maximum entropy learning. In *Proceedings of the Eighth Annual Conference of the International Speech Communication Association*. Antwerp, Belgium.

John Goldsmith. 2000. Linguistica: An automatic morphological analyser. In *Proceedings of the 36th Meeting of the Chicago Linguistic Society*. 125-139.

John Goldsmith. 2006. An algorithm for the unsupervised learning of morphology. *Natural Language Engineering*, 12(4):353-371.

Harold Hammarström and Lars Borin. 2011. Unsupervised learning of morphology. *Computational Linguistics*, 37(2):309-350.

Jon Kleinberg. 1999. Authoritative sources in a hyperlinked environment. *Journal of the ACM*, 46(5):604-632.

Hoifung Poon, Colin Cherry and Kristina Toutanova. 2009. Unsupervised morphological segmentation with log-linear models. In *Proceedings of the Conference of the North American Chapter of the ACL*, Boulder, CO, 209-217.

Paul Rodrigues and Damir Ćavar. 2005. Learning Arabic morphology using information theory. In *Proceedings of the Chicago Linguistics Society. Vol 41*. Chicago: University of Chicago. 49-58.

Patrick Schone and Daniel Jurafsky. 2001. Knowledge-free induction of inflectional morphologies. In *Proceedings of the Conference of the North American Chapter of the ACL*, Pittsburgh, PA, 183-191.

Aris Xanthos. 2008. *Apprentissage Automatique de la Morphologie: Le Cas des Structures Racine-Schème 'The Automatic Learning of Morphology: The Case of Root-and-Pattern Structures'*. Berne, Switzerland: Peter Lang.