# Developing Robust Arabic Morphological Transducer Using Finite State Technology

Mohammed A. Attia
mohammed.attia@postgrad.manchester.ac.uk
Ph.D. Student, School of Informatics,
The University of Manchester
To be submitted to the 8th Annual CLUK Research Colloquium 2005

## Abstract

Arabic morphology has always been a challenge for computational linguistics because of its richness and complexity. Arabic requires the verb to agree with the subject in person, number and gender, and requires the adjective to agree with the noun in number, gender, definiteness and case. The Arabic number system has a dual form. The base form in Arabic is the root. The root is a number of consonants (usually three) not pronounceable in themselves which undergoes a series of interdigitation with vowel marks, inflection and derivation to form hundreds of words, or stems. Moreover, Arabic employs clitics, which are grammatical morphemes, like prepositions and pronouns, that attach themselves to other words.

## Introduction

As Arabic is notorious for its morphological complexity (McCarthy 1985; Azmi 1988; Beesley 1998; Ratcliffe 1998; Ibrahim 2002), it has always been a challenge in computational morphology and a hard testing ground for morphological analysis technologies.

There are three strategies for the development of Arabic morphologies (Beesley and Karttunen 2003) depending on the level of analysis:

1. One level rules: analyzing Arabic at the stem level and using regular concatenation.
2. Two-level rules: analyzing Arabic words as composed of roots and patterns in addition to concatenations.
3. Three-level rules: analyzing Arabic words as composed of roots, templates and vocalization, besides concatenations.

I have developed my morphological analyzer using the one level rules approach considering stems as the base forms of Arabic words, and handling spelling variations through alteration rules. Actually using roots as the base forms of Arabic words is more efficient, especially in information retrieval systems. However, using the stem as base form is faster and easier to develop, and it will be more suitable for syntactic parsers that aim at translation.

In order to make the morphological transducer robust[1], I will develop a normalizer to handle diacritics, a guesser to handle unknown words, and rule relaxation layers to handle misspelled words.

## Arabic Morphology

It seems that Arabic traditional grammarians (Ibrahim 2002) have been persuaded by morphology to classify words into only three types: verbs, nouns and prepositions and particles. Adjectives take almost all the morphological forms of nouns. Adjectives, for example, can be definite, can be preceded by prepositions and are inflected for case, number and gender.

Arabic traditional grammarians (Ibrahim 2002) have also classified tense into present, past and imperative. This, as well, is influenced by the fact that verbs in Arabic are inflected for present, past and imperative. Moreover, both the past and the present have two forms: the active form and the passive form. To summarize, verbs are inflected to provide five forms: active past, passive past, active present, passive present and imperative. The base form of the verb is the past tense 3$^{rd}$ person singular. There are a number of factors that tell how the base form is inflected to give the other forms. Among these factors are the number of letters of the base form and its template. A template (Beesley and Karttunen 2003) is a kind of vocalization mould in which a verb fits. Diacritics are a major factor in template shaping. Although diacritics are not present in modern writing, we still need to worry about them as they trigger other phonological and orthographical processes like assimilation and deletion and the re-separation of doubled letters.

## Development Decisions

1. Using finite state technology. There are a number of advantages of this technology, among them are:
   - Handling concatenative and non-concatenative morphotactics (Beesley 1998).
   - Fast and efficient. It can handle very huge automata of lexicons with their inflections. Compiling large networks that include several millions of paths is only a matter of seconds in a finite state calculus. Moreover, these large networks can be easily unioned together to give even larger networks.
   - Unicode support which enables developers to accommodate native scripts.
   - Multi-platform support. Xerox finite state tools work well under Windows, Linux, UMIX and Mac OS, which means that the morphological transducer developed using finite state tools can serve applications under any of these platforms.
2. Separating the task of the developer and the linguist. As adding new terms to the lexicon in a morphological transducer is a never ending process, the lexicographer's job should be made as clear and easy as possible.
3. Making no account of diacritics. So this tool is not suitable for systems intended for speech applications. It is developed as a component in an Arabic-to-English MT

[1] A robust system is one that tries to give a useful output even if the input is varied or even incorrect.

system. After surveying a corpus of 1.5 million Arabic words, I found that only 347 words carry diacritic marks. One relatively common type of diacritics (*tanween* in the accusative case) is already handled by the system leaving only 54 instances of diacritic marks not accommodated by the system, which is statistically insignificant.

4. Developing a guesser to prevent the system from failing to give an output in the case of unknown words.
5. Handling multi-term expressions as a component inside the tokenizer.
6. Generating valid surface forms. Apparently, if the system is used only for analysis, there is no much point in making a restriction on it to generate only valid forms. Yet, practically, it is very helpful during development, as a way of testing the rules, to generate only valid forms. Moreover, overgeneration increases the size of the network needlessly. This may become a performance issue when the system comes to large-scale real-world implementation.
7. Developing a normalizer or spelling relaxation rules, to handle spelling variations and common Arabic mistakes.

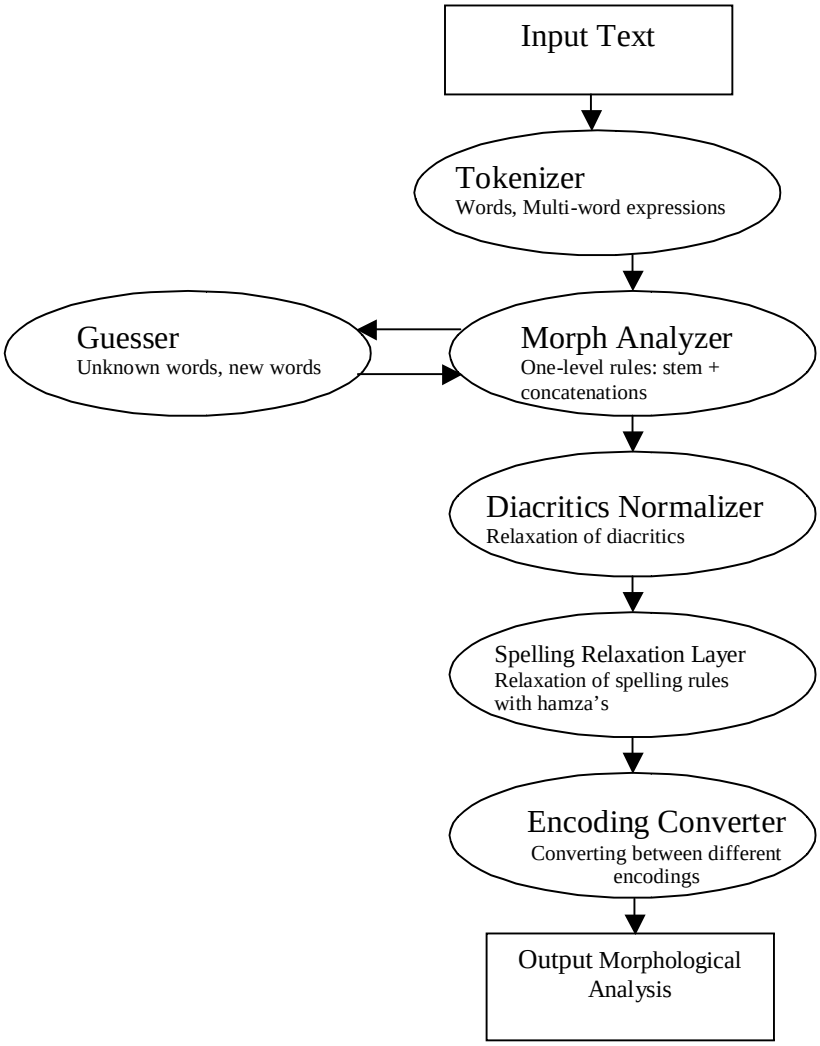


Figure 1. System Architecture

# System Description

The system handles concatenative morphotactics and spelling variations. Concatenative morphotactics are handled through continuation classes, and spelling variations are handled through alteration rules and compile replace. The core system has been developed in one month and is expected (when enough lexical items are incorporated) to provide as full and efficient coverage of MSA (Modern Standard Arabic) as any large-scale morphological analyzer. Two major development decisions led to this significant reduction in the development time. First I used stems as the base forms instead of roots (or radicals) interdigitated with patterns. Second, I avoided the nuisances of diacritics which are seldom used in MSA texts. Introducing diacritics is a heavy-weight burden for the both the developer and the lexicographer. The core system handles a test suite of 115 verbs, 10 nouns and adjective and the full range of the closed classes of prepositions, particles and modal verbs.

The system can integrate different parts and components as shown in Figure 1. These components are:

1. The tokenizer, which outputs each token to a single line and handles multi-term expressions.
2. The morphological analyzer.
3. The guesser, which serves a dual purpose (Beesley and Karttunen 2003), first as a guard against failing to give an analysis, and second as a way of adding new terms to the core lexicon.
4. The diacritics normalizer. This allows the analysis of diacritized texts, though the transducer will not make use of these diacritics to reduce the number of ambiguities, as it is designed for undiacritized texts. The aim is to prevent the system from failing to provide an analysis.
5. The spelling relaxation layer, which handles the following common spelling variations (Darwish 2002) or mistakes.
   - ى and ي could easily replace each other at the end of words
   - ى and ا could easily replace each other at the end of words
   - ئ, ء, أ, ؤ could replace each other
   - ا, أ or إ could mistakenly replace each other at the start of words
   - ه and ة could mistakenly replace each other at the end of words
   - Any letter can come accidentally with a diacritic mark
   - Kashida بـ, الـ, هـ, لـ
6. The encoding converter. This can be developed when needed. Yet UTF-8 encoding reduces the need for such a converter.

## Handling Arabic Morphotactics

Morphotactics is the study of how morphemes combine together to form words (Beesley 1998) These can be concatenative with morphemes either prefixed or suffixed to stems or non-concatenative, with stems themselves undergoing alterations to convey morphosyntactic information.

## Verbs

Possible concatenations and inflections in Arabic verbs are shown in Table 1. Elements in parentheses are optional. These entries are connected together and controlled though continuation classes.

Flag Diacritics are used to handle long distance morphotactic restrictions or what is termed separated dependencies for Arabic verbs. These can be summarized as follows:

- The yes-no-question article (أ "a" or *does*) cannot co-occur with imperatives or with the accusative case.
- The complementizer (ل "li" or *to*) cannot co-occur with the nominative case.
- Cliticized object pronouns do not occur either with passive or with intransitive verbs.
- Affixes indicating person and number in the present tense come in two parts one preceding and one following the verb and each prefix can co-occur only with certain suffixes.
- Present, past and imperative have each a range of prefixes or suffixes or both which must be precisely constrained.

<table>
<tr><th>(Conjunction or question Article)</th><th>(Complementizer)</th><th>Tense Prefix</th><th>Verb Stem</th><th>Tense Suffix</th><th>(Clitic Object Pronoun)</th></tr>
<tr><td>Conjunctions و "wa" (and) or ف "fa" (then)</td><td rowspan="3">ل "li" (to)</td><td>Present tense prefixes</td><td rowspan="3">Stem</td><td>Present tense suffix</td><td>First person object pronoun</td></tr>
<tr><td rowspan="2">Question word أ "a" (does or did)</td><td>Past tense prefix</td><td>Past tense suffix</td><td>second person object pronoun</td></tr>
<tr><td>Imperative prefix</td><td>Imperative suffix</td><td>Third person object pronoun</td></tr>
</table>

Table 1: Possible concatenations in Arabic verbs

The tool generates up to 1800 well-formed forms for transitive verbs. The verb tested was شكر "shakar" (to thank). This incredible amount of form variations is really a good indication of the richness and complexity of Arabic morphology. The spelling of the generated words is checked manually and using MS Words and almost all of them are sound forms.

## Nouns

Possible concatenations and inflections in Arabic nouns are shown in Table 2, which are controlled and connected through continuation classes.

| (Conjunction or question Article) | (Preposition) | (Definite Article) | Noun Stem | (Dual/Plural Suffix) | (Clitic Genitive Pronoun) |
|---|---|---|---|---|---|
| Conjunctions و "wa" (and) or ف "fa" (then) | ب "bi" (with), ك "ka" (as) or ل "li" (to) | ال "al" (the) | Stem | Dual | First person pronoun |
| Question word أ "a" (does or did) | | | | Masculine regular plural | second person pronoun |
| | | | | Feminine regular plural | Third person pronoun |

Table 2: Possible concatenations in Arabic nouns

Flag Diacritics are also used to handle separated dependencies for nouns. These can be summarized as follows:

- The definite article (ال "al" or *the*) cannot co-occur with a genitive pronoun.
- The definite article cannot co-occur with an indefinite noun marking (*nuun* with the dual and plural or *tanween* with the singular).
- The cliticized genitive pronoun cannot co-occur with an indefinite noun marking.
- Prepositions cannot co-occur with nominative or accusative case markings.

The tool generates up to 519 valid forms for regular nouns that accept the feminine mark and regular plural marks. The noun tested was معلم "mu'allim" (teacher).

```
LEXICON Nouns

+masc^ss^معلم^se^                              DualFemFemplMascpl;

+masc^ss^طالب^se^                              DualFemFempl;
طالب+masc+irregplural:^ss^طلاب^se^             CaseEnds;

+masc^ss^كتاب^se^                              Dual;
كتاب+masc+irregplural:^ss^كتب^se^              CaseEnds;

+fem^ss^كراسة^se^                              DualFempl;

+fem^ss^شمس^se^                                Dual;
شمس+fem+irregplural:^ss^شموس^se^               CaseEnds;
```

Figure 2. Noun Stem Entry

Besides continuation classes that can be deduced from Table 2, There are a number of continuation classes with regard to the type of noun in question, as shown in Figure 2. These additional continuation classes are based on the following facts:

1. All nouns can take the dual morpheme.
2. Some masculine nouns take the feminine, regular plural feminine and regular plural masculine morphemes. This is represented by entry 1 in Table 3.
3. Some masculine nouns take the feminine and regular plural feminine morphemes. However they take a broken plural masculine form. This form must be entered separately by the lexicographer. This is represented by entry 2 in Table 3.
4. Some masculine nouns do not accept the feminine morpheme and have a broken plural form. This is represented by entry 3 in Table 3. This is usually the case with inanimate masculine nouns such as كتاب "kitab" (book) and masculine nouns that have separate lexical entry for the feminine such as the masculine noun عجل "'ijl" (bull) whose feminine form is بقرة "baqarah" (cow).
5. Some feminine nouns take the regular feminine plural morpheme. This is represented by entry 4 in Table 3. This is usually with grammatical or natural feminine.
6. Some feminine nouns have a broken plural form. This is represented by entry 5 in Table 3.

| | Stem | Feminine Singular | Masculine Dual | Feminine Dual | Regular Masculine Plural | Regular Feminine Plural | Broken Plural |
|---|---|---|---|---|---|---|---|
| 1 | معلم mu'allim (teacher) | معلمة mu'allimah | معلمان mu'alliman | معلمتان mu'allimatan | معلمون mu'allimuun | معلمات mu'allimat | X |
| 2 | طالب talib (student) | طالبة talibah | طالبان taliban | طالبتان talibatan | X | طالبات Talibat | طلاب tullab |
| 3 | كتاب kitab (book) | X | كتابان kitaban | X | X | X | كتب kutub |
| 4 | كراسة kurrasah (notebook) | X | X | كراستان kurrasatan | X | كراسات Kurrasat | X |
| 5 | شمس shams (sun) | X | X | شمسان shamsan | X | X | شموس shumuus |

Table 3. Distribution of possible feminine and plural morphemes

The problem with nouns is mainly with broken plurals (Ratcliffe 1998; Ibrahim 2002). "Broken plural" is the traditional grammarians' term for describing the process of non-concatenative plural. The term was chosen to indicate that the base form of the nouns is broken either by removing one or more letters, adding one or more letters, changing vocalization or a combination of these. Arabic singular nouns have 30 templates served by 39 broken plural templates. A single template of the singular noun can have up to seven broken plural templates. The differing plural templates were historically meant to indicate some meaning differences, such as whether the number of the plural is below or

above ten, and whether the noun describes a profession or an attribute, and whether the attribute is static or transient. These subtle meaning differences are no longer recognized even by well-educated modern speakers.

The system relies only on the lexicographer to tell whether a particular noun is to have a regular or broken plural form and, if it is to take a broken plural form, which template it is to take. Trying to rely on the system to guess the broken plural form will make the transducer overgenerate excessively and needlessly. Typing in the broken plural form will be a burden but not a big trouble for a lexicographer as shown in Figure 2.

# Alteration Rules

Alterations or variations are the discrepancies between underlying strings and their surface realization which is phonological or orthographical or both (Beesley 1998).

Diacritics, when they are used, serve in Arabic to indicate short vowels. Long vowels, glides and the glottal stop are all represented by alphabetic letters. As could be expected phonologically, these sounds are the subject of a great deal of phonological (and consequently orthographical) alterations like assimilation and deletion. Most of the trouble a morphological analyzer faces is related to handling these issues. In my system I have written more than 60 replace rules composed on the bottom of the verbs lexicon to handle alteration rules that map stem forms into all other forms.

Traditional grammarians used to classify verbs regarding the number of letters of the base form into three-, four-, five- and six-letter verbs. Furthermore, regarding whether or not the verb includes a long vowel, a glide or a glottal stop, it can be broadly classified into five types:

1. Verbs with an initial glottal stop, long vowel or glide
2. Verbs with a medial glottal stop, long vowel or glide. With verbs more than three letters long, their position inside the word can have effective difference.
3. Verbs with a final glottal stop, long vowel or glide.
4. Verbs that contain a doubled letter in the second, third, fourth, fifth or sixth position. A verb consisting of two letters and one of them is doubled is traditionally and morphologically classified as a three-letter verb.
5. Sound verbs, or verbs that contain neither of the above.

The start and end of the stems are marked in the lower-side part of the network, as shown by Figure 3, so that alteration rules can be applied correctly to words. The markings and the information to be entered by the lexicographer are:

1. Start and end of verb stem. The multi-character symbol “^ss^” stands for stem start and “^se^” for stem end.
2. Which letter is doubled in the linear order as in the entries from 4 to 8 in Figure 3. The mark “^dbl2^dbl”, for example means that the second letter is doubled.
3. If there is a long vowel that undergoes assimilation, the assimilated form needs to be explicitly stated. This is represented by the entries from 10 to 13 in Figure 3. In traditional terms the origin of ا “a” in قال “qal” (said) is و “w”.

These markings are considered an intermediate language which is removed in the final stage, so that only surface strings are left on the bottom and analysis strings (or lexical strings) are left on the top of the network (Beesley 1996).

```
1  LEXICON Verbs
2   ^ss^شكر^se^                          Transitive;
3   ^ss^فـرح^se^                         Intransitive;
4   ^ss^رد^se^^dbl2^dbl                  Transitive;
5   ^ss^أمـر^se^^dbl2^dbl                Transitive;
6   ^ss^أضر^se^^dbl3^dbl                 Intransitive;
7   ^ss^امـتد^se^^dbl4^dbl               Intransitive;
8   ^ss^تمخض^se^^dbl3^dbl                Intransitive;
9   ^ss^استقـر^se^^dbl5^dbl              Intransitive;
10  ^ss^بـاع^se^^origي^orig              Transitive;
11  ^ss^قـال^se^^origو^orig              Intransitive;
12  ^ss^غزا^se^^origو^orig               Transitive;
13  ^ss^رمـى^se^^origي^orig              Transitive;
```

Figure 3. Verb Stem Entry

# Conclusion

Finite state technology can be efficiently used to make an Arabic morphological transducer. Development time can be remarkably reduced when we use the stem as the base form and when we ignore diacritics which are seldom used in Modern Standard Arabic.

# References


Azmi, M. (1988). Arabic Morphology: A Study in the System of Conjugation. Hyderabad, Hasan Publishers.

Beesley, K. R. (1996). Arabic Finite-State Morphological Analysis and Generation. Proceedings of the 16th conference on Computational linguistics, Copenhagen, Denmark, Association for Computational Linguistics.

Beesley, K. R. (1998). Arabic Morphology Using Only Finite-State Operations. Proceedings of the Workshop on Computational Approaches to Semitic languages, Montreal, Quebec.

Beesley, K. R. and L. Karttunen (2003). Finite State Morphology. Stanford, Calif., Csli.

Darwish, K. (2002). Building a Shallow Morphological Analyzer in One Day. Proceedings of the workshop on Computational Approaches to Semitic Languages in the 40th Annual Meeting of the Association for Computational Linguistics (ACL-02), Philadelphia, PA, USA.

Ibrahim, K. (2002). Al-Murshid fi Qawa'id Al-Nahw wa Al-Sarf [The Guide in Syntax and Morphology Rules]. Amman, Jordan, Al-Ahliyyah for Publishing and Distribution.

McCarthy, J. J. (1985). Formal Problems in Semitic Phonology and Morphology. New York ; London, Garland.

Ratcliffe, R. R. (1998). The Broken Plural Problem in Arabic and Comparative Semitic : Allomorphy and Analogy in Non-concatenative Morphology. Amsterdam ; Philadelphia, J. Benjamins.