Developing a Robust Arabic Morphological Transducer/Tokenizer, and Integration with XLE

By
Mohammed A. Attia
Ph.D. Student,
School of Informatics,
The University of Manchester
Introduction

Available Arabic Morphological Analyzers:

- Xerox Finite State Arabic Morphological Analyzer
- Buckwalter Arabic Morphological Analyzer
Introduction

Arabic Morphological Peculiarities

- Large number of prefixes and suffixes to show person, number and gender with verbs, and number and gender with nouns
- Separated Dependencies
- Clitics
A New Arabic Transducer - Why?

- Specific domain – News
- Specific language – MSA
- Specific purpose – MT
- Compatibility – XLE
- Native script
- Maintenance and update
- Owning tools: customizability in form and content
Development Decision

- Using finite state technology with the Advantages:
  - Handling concatenative and non-concatenative morphotactics
  - Fast and efficient
  - Unicode support
  - Multi-platform support
Development Decision

- Using the stem as the base form, which makes the solution:
  - Easier and faster to develop
  - More suitable for translation
Development Decision

- Separating the task of the developer and the lexicographer
- Making no account of diacritics
- Generating valid surface forms
- Developing a guesser to prevent the system from failing
System Architecture

- Tokenizer
- Morphological Transducer
- Guesser
- Diacritics Normalizer
- Spelling Relaxation Layer
Morph Analyzer
One-level rules: stem + concatenations

Encoded Converter
Converting between different encodings

Diacritics Normalizer
Relaxation of diacritics

Spelling Relaxation Layer
Relaxation of spelling rules

Tokenizer
Words, Multi-word expressions

Guesser
Unknown words, new words

Input Text

Output Morphological Analysis
## Verb Morphotactics

### Possible Concatenations

<table>
<thead>
<tr>
<th>(Conjunctions or question Article)</th>
<th>(Complementizers)</th>
<th>Tense Prefixes</th>
<th>Verb Stem</th>
<th>Tense Suffixes</th>
<th>(Clitic Object Pronouns)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Conjunctions و “wa” (and) or ف “fa” (then)</td>
<td>ل “li” (to)</td>
<td>Present tense prefixes (5)</td>
<td>Stem</td>
<td>Present tense suffixes (10)</td>
<td>First person object pronoun (2)</td>
</tr>
<tr>
<td>Question word أ “a” (does or did)</td>
<td>س “sa” (will)</td>
<td>Past tense prefix (1)</td>
<td></td>
<td>Past tense suffixes (12)</td>
<td>second person object pronoun (5)</td>
</tr>
<tr>
<td></td>
<td>ل “la” (then)</td>
<td>Imperative prefix (2)</td>
<td></td>
<td>Imperative suffixes (5)</td>
<td>Third person object pronoun (5)</td>
</tr>
</tbody>
</table>
Verb Morphotactics

- Statistically these (unconstrained) concatenations can generate up to: 33,696 Forms
  \[3 \times 4 \times 8 \times 27 \times 13\]
- Flag Diacritics are used to handle separated dependencies (constrained concatenations)
- 2,552 well-formed forms for transitive verbs
Verb Morphotactics

Alternation Rules

- Over 60 replace rules to handle alternation rules with “weak letters”
  - Verbs with an initial glottal stop, long vowel or glide
  - 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.
  - Verbs with a final glottal stop, long vowel or glide.
  - Verbs that contain a doubled letter in the second, third, fourth, fifth or sixth position.
## Noun Morphotactics

### Possible Concatenations

<table>
<thead>
<tr>
<th>(Conjunction or question Article)</th>
<th>(Preposition)</th>
<th>(Definite Article)</th>
<th>Noun Stem</th>
<th>(Suffixes)</th>
<th>(Clitic Genitive Pronoun)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Conjunctions و “wa” (and) or ف “fa” (then)</td>
<td>Feminine Mark (1)</td>
<td>ال “al” (the)</td>
<td>Stem</td>
<td>Masc Dual (4)</td>
<td>First person pronoun (2)</td>
</tr>
<tr>
<td>Question word أ “a” (does or did)</td>
<td></td>
<td></td>
<td></td>
<td>Fem Dual (4)</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Masculine regular plural (4)</td>
<td>second person pronoun (5)</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Third person pronoun (5)</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Third person pronoun (5)</td>
<td></td>
</tr>
</tbody>
</table>
Noun Morphotactics

- Statistically these (unconstrained) concatenations can generate up to 6,240 forms
  \[4 \times 4 \times 2 \times 15 \times 13\]
- Constrained concatenations generate 519 valid forms
Noun Morphotactics

Noun Types according to gender and number

- 13 Types
- Valid inflections must be specified in the lexicon
<table>
<thead>
<tr>
<th>No.</th>
<th>Word</th>
<th>Masculine Singular</th>
<th>Feminine Singular</th>
<th>Masculine Dual</th>
<th>Feminine Dual</th>
<th>Regular Masculine Plural</th>
<th>Regular Feminine Plural</th>
<th>Broken Plural</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>jahil</td>
<td>jahilah</td>
<td>jahilan</td>
<td>jahilatan</td>
<td>jahilun</td>
<td>jahilat</td>
<td></td>
<td>juhala’</td>
</tr>
<tr>
<td>2</td>
<td>mu’allim</td>
<td>mu’allimah</td>
<td>mu’alliman</td>
<td>mu’allimatan</td>
<td>mu’allimuun</td>
<td>mu’allimat</td>
<td>X</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>talib</td>
<td>talibah</td>
<td>taliban</td>
<td>talibatan</td>
<td>X</td>
<td>Talibat</td>
<td>tullab</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>ta’limi</td>
<td>Ta’limiah</td>
<td>ta’limian</td>
<td>ta’limiatan</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>5</td>
<td>imtihan</td>
<td>X</td>
<td>Imtihanan</td>
<td>X</td>
<td>X</td>
<td>Imtihanat</td>
<td>X</td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>kitab</td>
<td>X</td>
<td>kitaban</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>kutub</td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>shajarah</td>
<td>X</td>
<td>shajaratan</td>
<td>X</td>
<td>shajarat</td>
<td>shajar</td>
<td></td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>hamsah</td>
<td>X</td>
<td>hamsatan</td>
<td>X</td>
<td>hamasat</td>
<td>X</td>
<td></td>
<td></td>
</tr>
<tr>
<td>9</td>
<td>shams</td>
<td>X</td>
<td>shamsan</td>
<td>X</td>
<td></td>
<td>X</td>
<td>shumus</td>
<td></td>
</tr>
<tr>
<td>10</td>
<td>tanazul</td>
<td>X</td>
<td>X</td>
<td></td>
<td></td>
<td>tanazulat</td>
<td>X</td>
<td></td>
</tr>
<tr>
<td>11</td>
<td>khuruj</td>
<td>X</td>
<td>X</td>
<td></td>
<td></td>
<td>X</td>
<td>X</td>
<td></td>
</tr>
<tr>
<td>12</td>
<td>Mohammed</td>
<td>X</td>
<td>X</td>
<td></td>
<td></td>
<td>X</td>
<td>X</td>
<td></td>
</tr>
<tr>
<td>13</td>
<td>Zainab</td>
<td>X</td>
<td>X</td>
<td></td>
<td></td>
<td>X</td>
<td>X</td>
<td></td>
</tr>
</tbody>
</table>
# Noun types according to number and gender

<table>
<thead>
<tr>
<th></th>
<th>Masculine Singular</th>
<th>Feminine Singular</th>
<th>Masculine Dual</th>
<th>Feminine Dual</th>
<th>Regular Masculine Plural</th>
<th>Regular Feminine Plural</th>
<th>Broken Plural</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>2</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>3</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>No</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>4</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
<td>No</td>
</tr>
<tr>
<td>5</td>
<td>Yes</td>
<td>No</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>6</td>
<td>Yes</td>
<td>No</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>Yes</td>
</tr>
<tr>
<td>7</td>
<td>No</td>
<td>Yes</td>
<td>No</td>
<td>Yes</td>
<td>No</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>8</td>
<td>No</td>
<td>Yes</td>
<td>No</td>
<td>Yes</td>
<td>No</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>9</td>
<td>No</td>
<td>Yes</td>
<td>No</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
<td>Yes</td>
</tr>
<tr>
<td>10</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>11</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
</tr>
<tr>
<td>12</td>
<td>Yes (Prop)</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
</tr>
<tr>
<td>13</td>
<td>No</td>
<td>Yes (Prop)</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
</tr>
</tbody>
</table>
Broken plurals are not handled in a rule-based approach. The problem with broken plural:

- 30 singular noun templates served by 39 broken plural templates
- Broken plural forms are fossilized
- They are to be entered by hand
Function Words Morphotactics

- Conjunctions
- Pronouns
- Prepositions
- Modal Verbs
- Question Words

- Demonstratives
- Relatives
- Particles
  - Confirmation
  - Negation
  - Exception
  - Complementization
  - Future
  - Condition
Function Words Morphotactics

Function words take either:

- No prefix or suffix
  - Independent conjunctions
- Conjunction prefixes and no suffix
  - Independent Pronouns
- Conjunction prefixes and a pronoun prefix
  - Modals
- Conjunction and preposition prefixes and no suffix
  - Demonstrative pronouns
Analysis

- Ambiguities
  - Active vs. Passive vs. Imperative
    - كرم
      - Karrama (Active)
      - Kurrima (Passive)
      - Karrim (Imperative)
  - 2nd Person Masc vs. 3rd Person Fem
    - تشكر
      - tashkur (2nd Person Masc)
      - tashkur (3rd Person Fem)
  - 1st Person sg vs. 3rd Person fem
    - شكرا
      - shakartu (1st Person sg)
      - Shakarat (3rd person fem)
Analysis

- Ambiguities
  - Different Entries
    - أقال
      - aqala (+QuestionParticle [qala])
      - aqala
  - Different POS
    - شكرا
      - shakara (verb)
      - shukr (noun)
Analysis

- المعلم: +3pers+noun+masc
- طالب: +3pers+noun+masc
- امتحن: +imp[امتحن]+2pers+masc+sg
- امتحن: +past+active[امتحن]+3pers+sg+masc
- امتحن: +past+active[امتحن]+3pers+pl+fem
- امتحن: +past+pass[امتحن]+3pers+sg+masc
- امتحن: +past+pass[امتحن]+3pers+pl+fem
- شكر: +past+active[شكر]+3pers+sg+masc
- شكر: +past+pass[شكر]+3pers+sg+masc
- فهم: +conj+pron+3pers+pl+masc
- فهم: +conj+obj3+them
- علم: +past+active[علم]+3pers+sg+masc
- علم: +past+pass[علم]+masc+sg
- انهزم: +imp[انهزم]+2pers+masc+sg
- انهزم: +past+active[انهزم]+3pers+sg+masc
- انهزم: +past+pass[انهزم]+masc+sg
- استعان: +past+active[استعان]+3pers+sg+masc
Generation

- Generating valid forms
- Eliminating ill-formed forms
- Accommodating spelling variation and common spelling errors in analysis but not in generation
Whereas the morphological transducer provides analysis, the tokenizer is responsible for identifying:

- Word boundaries
- Multi-word expressions
- Punctuation
- Abbreviations
- Clitics
Tokenization and Analysis:
First Approach – 2 in 1

Why they are inseparable in dealing with Arabic clitics (prepositions, pronouns, conjunctions, etc.)

- Clitics can be concatenated one after the other.
- Clitics undergo assimilation with words.
- Without complete morphological knowledge, you cannot tell whether some initial or final letters are part of the word or only clitics.
Tokenization and Analysis: First Approach – 2 in 1

Implementation

- Tokenizer is responsible for deciding word boundaries, clitic boundaries as well as analysis
- Morphological analyzer: accepts the output of the tokenizer as is

In fact the core morphological analyzer is part of the tokenizer
Tokenization and Analysis: First Approach – 2 in 1

Implementation – Tokenizer output: +morph feature @token boundary

- وللرجل (and to the man)
  - رجل + conj@ + prep@ + ل + al + defArt@ + noun + masc@

- ولمعلمهم (and to their teacher)
  - معلم +هم + noun + masc@ + genpron@

- وشكر (and he thanked/is thanked)
  - شكر + 3pers + sg + masc@ + conj@ + verb + past + active
  - شكر + 3pers + sg + masc@ + conj@ + verb + past + pass

- وليرشكورهم (and to thank them)
  - ل + comp@ + verb + pres + active + 3pers + شكر + masc + sg + genpron + objpron
Tokenization and Analysis: Second Approach – Clitics Guesser

Step 1: Developing a guesser for Arabic words with all possible clitics, and accommodating possible assimilations. This guesser is then used by the tokenizer to mark clitic boundaries. There will be no analyses, but there will be increased tokenization ambiguities.

(والرجل)

(الرجل والرجل)

(و للرجل والرجل)

(ووالرجل)
Step 2: Developing a lexc transducer for clitics only, treating them as separate words. Then a morphological transducer is created by applying rules to remove all paths that contain any clitics from the core morphology. The output is then unioned with the clitics transducer.
Tokenization and Analysis: Second Approach – Clitics Guesser

Advantages:
1. Keeping the core morphology intact
2. Following the usual rule of separating the tokenizer and the analyzer.
3. Trees display more nicely in XLE.

Disadvantages:
1. The system has to deal with tokenization ambiguities. For a simple sentence of 3 words, I get 8 different tokenization solutions.
2. I have to write stricter sublexical rules.
3. Treating clitics as free morphemes will create ambiguities with some originally free morphemes. Sometimes there will be an ambiguity also regarding whether this clitic belongs to the previous or the following word.
Integration

Integration with XLE: 4 Steps

- Adding a morphology section in the grammar file and referring to it in the grammar configuration section
- Setting the character encoding UTF-8 in the configuration section and in the test file
- Writing sublexical rules
- Writing sublexical entries
Integration

Problems with Arabic in XLE:

- Arabic fonts do not display correctly in trees and charts.
- When printing postscript for any chart, Arabic fonts disappear.
- You cannot write Arabic on the shell under Mac OS, and when you do under Linux the encoding is not interpreted correctly.
Conclusion

“Linguistic development is an endless round of observation, theorizing, formalizing and testing; and the goal, for a lexical transducer, is to create a system that correctly analyzes and generates a language that looks as much like the real natural language as possible.”

Beesley and Karttunen, Finite State Morphology. P. 287
Conclusion

- FST is fast, efficient and reliable.
- Development time can be reduced significantly for Arabic if we take the stem as the base form and ignore diacritics.