
Proceedings Paper:
Alosaimy, AMS and Atwell, ES (2015) A review of morphosyntactic analysers and tag-sets for Arabic corpus linguistics. In: Corpus Linguistics 2015. Corpus Linguistics 2015, 21-24 Jul 2015, Lancaster, UK. , pp. 16-19.

A review of morphosyntactic analysers and tag-sets for Arabic corpus linguistics

Abdulrahman AlOsaimy
University of Leeds
scama@leeds.ac.uk

Eric Atwell
University of Leeds
e.s.atwell@leeds.ac.uk

1 Introduction and credits

Geoffrey Leech applied his expertise in English grammar to development of Part-of-Speech tagsets and taggers for English corpora, including LOB and BNC tagsets and tagged corpora. He also developed EAGLES standards for morphosyntactic tag-sets and taggers for European languages. We have extended this line of research to Arabic: we present a review of morphosyntactic analysers and tag-sets for Arabic corpus linguistics.

The field of Arabic NLP has received a lot of contributions in the last decades. Many analysers handle its morphological-rich problem in Modern Standard Arabic text, and at least there are six freely available morphological analyzers at the time of writing this paper. However, the choice between these tools is challenging. In this extended abstract, we will discuss the outputs of these different tools. We show the challenge of comparing between them.

The goal of this abstract is not to evaluate these tools but to show the differences. We aim also to ease the building of an infrastructure that can evaluate every tool based on common criteria and produce a universal pos-tagging.

2 Presentation of morphological analysers

- **BAMA:** A widely-known Perl-based freely available Arabic morphological analyser by Tim Buckwalter. The analyser used in this research is version 1.3. Later versions needs an LDC licence and therefore not considered in this comparison.

Outputs: POS tag, gloss, voweled word and stem. The tagset of Buckwalter is about 70 basic subtags, and they can be combined to form more complex tag such as: IV_PASS which means imperfective passive verb. Those tags include features of verbs like person, voice, mood, aspect and its subject like gender and number. It also includes features of nominal like gender, number, case and state. BAMA provides a list of different analysis with no disambiguation of them.

- **Mada:** a freely available toolkit that tokenizes, pos-tags, lemmatize, stems a raw Arabic input. This toolkit, its successor MADAMIRA disambiguates the analyses by showing the probability of each analysis.

Outputs: POS tag, gloss, voweled word, stem and the word lemma. The output tagset can be one of four different POS tagsets: ALMORGEANA, CATiB, POS:PENN, Penn ATB, or Buckwalter. Features of verbs like person, voice, mood, aspect and its subject like gender and number are explicitly provided. Same for features of nominal like gender, number, case and state. MADA provides a list of different analysis each with a probability. The higher is the more likely one.

- **MadaAmira**: is the Java-Based successor of Mada that combines Mada and Amira tools. It adds some aspects from Amira tool.

Outputs: In addition to the output of Mada, the base phrase chunks and named entities can be provided.

- **AlKhalil**: **"**a morphosytactic parser" of MSA that is a combination of rule-based and table-lookup approach.

Outputs: AlKhalil is different as it all output is a table-like provided in Arabic sentence that describe the morphological analysis of each word. The table have POS-tags, prefix, suffix, pattern, stem, root and voweled word columns. Features of verbs like voice, transitivity and aspect are extractable. However the mood and person is not explicitly provided neither its subject if it a suffix. Nominal features are also extractable. In addition AlKhalil provides the nature of the noun, word root, and verb form.

- **Elixir:** is a morphology analysers and generator that reuse and extends the functional morphology library for Haskell.

Outputs: Elixir uses a custom output format including gloss, voweled word, root, stem, pattern, and a 10-letters word that describes the POS tag and all words features as Mada.

- **AraComLex**: is an open-source finite-state morphological processing toolkit.

Outputs: AraComLex provides the main POS tags categories: prep, conj, noun, verb, rel, adj … etc. For nominals, it provides its classification class (13 classes), number, gender, case, and whether it is human or not. For verbs, it provides number, gender, person, aspect, mood, voice, transitivity and whether allows passive or imperative.

- **ATKS**: is web-based service of NLP components targeting Arabic language that includes "full-fledged" morphological

analyser (Sarf) and part-of-speech (POS) tagger.

Outputs: Like Buckwalter tagset, ATKS provides complex tags that encompass nominal and verb features. All features are extractable from pos-tags. Sarf provides a list of features like: stem, root, pattern, discretized token, isNunatable and probability of each analysis.

- **Stanford NLP tools:** open-source software in Java that has a segmenter, pos-tagger and parser of Arabic text.

Outputs: The output of Stanford parser and pos-tagger is Bies tagset which is used for Arabic Penn Treebank. This tagset is linguistically coarse (Habash 2010) and therefore many features are missing. The features that are extractable are aspect (unless it is passive as perfect and imperfect verbs share the same tag), number (singular or plural only) and voice.

- **Xerox:** web-based morphological analyser and generator built using Xerox Finite-State Technology.

Outputs: The output of Xerox analyser includes POS tag, English gloss, root, verb form and verb pattern. Features of verbs like person, voice, mood, aspect and its subject like gender and number are provided. Same for features of nominal like gender, number, case and state.

- **QAC**: the Quranic Arabic Corpus is a linguistic resource that includes segmentation and pos-tagging the Quran text. We used this resource as the gold standard for evaluating other tools as it has been verified by experts in Arabic language.

3 Work

We built an infrastructure for parsing all results from the tools mentioned above. For every analysis of a word, we parsed the tags associated with it and extracted the features (if possible) of the nominals and verbs (Fig 1).

We plan to benchmark every tool by comparing its results to the Quranic Arabic Corpus. For every feature that the QAC provides, we will find the accuracy, precision, and recall of each tool. However, benchmarking needs to first map all part of speech tags to one universal tag set. Another problem is that some tools provide different *unordered* analyses. We plan to find the best analysis that matches the QAC and report the results of that analysis.

Feature	Possible Values	Applied to
Gender	Male/Female	Nomonials & Subj. of verb
Number	Sing./Dual/Plural	Nomonials & Subj. of verb
Case	nominative, accusative, genitive	Nominals
state	Definite or Not	Nominals
Person	First, Second, Third	Verbs
voice	active, passive	Verbs
aspect	perfective, imperative, imperfective	Verbs
mood	indicative, subjunctive, jussive, energetic	imperfective verbs

Table 1 8 inflectional features in Arabic

4 Challenges:

Problem 1: The diverse in the format of the output: Every tool has its own format of output. Alkhalil return a table-like CSV file. Mada and MadaAmira return a text of *feature:value* pairs. However, some tools have more complex output like BAMA that needs to build a custom parser designed specifically for that tool. Therefore, for each tool, we need to translate the custom outputs to an open standard format: JSON. As a consequence, the infrastructure needs to be updated every time one of the tools changes its output scheme.

Problem 2: The availability of some tools: While many researchers published papers about their morphology tools, many of these are either not available or require a licence. For example, although Mada toolkit is freely available, it requires a lexicon tables that are only available with membership of LDC. In addition, some web services such as Xerox are limited to some quotas.

Problem 3: Different segmentation of words: For a valid comparison, words need to be similarly segmented. However, some tools cannot accept segmented text and instead it segments the input text as a preprocessing step.

Problem 4: Extracting features from POS tags: Although some tools do not explicitly present some important features such as gender, number and person, these features can be extracted from the POS tag of that word. However, such handling needs very careful understanding of the POS tags and could produce some errors by such manipulation. Every tool has its own set of tagsets. Tagsets sizes vary wildly. Buckwalter tagset for example can hypothetically reach over 330,000 tags (Habash 2010), while Stanford tagger used Bies tagset that has around 20+ tags. Those tagsets needs to be mapped to one universal tagset in order to be able to compare between them. Mapping will result in many features unknown, or have multiple possible values. In addition, the values of some features do not cover all possible values; number feature in Stanford can be only singular or plural, but in Arabic it could be

dual.

Problem 5: Different possible configurations: Mada has different configurations of preprocessing the input text. Different configurations lead to different tokenization, and therefore different analyzing. We chose the default settings, and we will leave comparing different configurations for future work.

Problem 6: Expectancy of input: While some tools expect unvoweled text data (AraComLex), some accept fully or partially voweled such as AlKhalil. ATKS used these short vowels to filter the best analyses if it fits or the diacritics will be ignored. Mada expects the input text to be text-only one sentence per line with no tags or meta data. AraComLex expects every word to be in a single line. Stanford parser expects tokenized words except the definitive AL.

Problem 7: Different Transliteration Schemes: Different tools encode the results in either ASCII or UTF-8. Some use a one-to-one transliteration scheme like Buckwalter transliteration. However, B.W. transliteration received several extensions, and determining which extension can be difficult when tool has a lack or poor user manual. Other tools like Elixir uses ArabTex encoding whose mapping can be two-to-one or has some alternatives.

References

Aliwy, Ahmed Hussein. *Arabic Morphosyntactic Raw Text Part of Speech Tagging System*. Diss. Repozytorium Uniwersytetu Warszawskiego, 2013.

Habash, Nizar Y. "Introduction to Arabic natural language processing."*Synthesis Lectures on Human Language Technologies* 3.1 (2010): 1-187.

Smrž, Otakar. *Functional Arabic Morphology. Formal System and Implementation*. Diss. Ph. D. thesis, Charles University in Prague, Prague, Czech Republic, 2007.

Boudlal, Abderrahim, et al. "Alkhalil Morpho SYS1: A Morphosyntactic Analysis System for Arabic Texts." *International Arab Conference on Information Technology*. 2010.

Dukes, Kais, and Nizar Habash. "Morphological Annotation of Quranic Arabic."*LREC*. 2010.

Atwell, E. S. "Development of tag sets for part-of-speech tagging." (2008): 501-526.

Jaafar, Younes, and Karim Bouzoubaa. "Benchmark of Arabic morphological analyzers challenges and solutions." *Intelligent Systems: Theories and Applications (SITA-14), 2014 9th International Conference on*. IEEE, 2014.

Pasha, Arfath, et al. "Madamira: A fast, comprehensive tool for morphological analysis and disambiguation of arabic." *In Proceedings of the 9th International Conference on Language Resources and Evaluation, Reykjavik, Iceland*. 2014.

Habash, Nizar, Owen Rambow, and Ryan Roth. "Mada+ tokan: A toolkit for arabic tokenization, diacritization, morphological disambiguation, pos tagging, stemming and lemmatization." *Proceedings of the 2nd International Conference on Arabic Language Resources and Tools (MEDAR), Cairo, Egypt*. 2009.

Green, Spence, and Christopher D. Manning. "Better Arabic parsing: Baselines, evaluations, and analysis." *Proceedings of the 23rd International Conference on Computational Linguistics*. Association for Computational Linguistics, 2010.

Attia, Mohammed, et al. "A lexical database for modern standard Arabic interoperable with a finite state morphological transducer." *Systems and Frameworks for Computational Morphology*. Springer Berlin Heidelberg, 2011. 98-118.

Buckwalter, Tim. "Buckwalter {Arabic} Morphological Analyzer Version 1.0." (2002).